UNIVERSITÀ
DI TRENTO

Department of Information Engineering and Computer Science

Doctoral Thesis

-About Me and You-

iTelos
A cooperative methodology for diversity-aware Knowledge Graphs

---

Author:
Simone Bocca

Supervisors:
Prof. Fausto Giunchiglia

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

March 31, 2025

Contents

List of Figures

Abstract

The representation of information through digital data has changed many times over the years, as has the way we interpret and use the data. Today, the increased use of AI technologies has led to an impressive amount of online data that needs to express the information it contains in a very precise and concrete way, for two main reasons. (i) To concretely express the information diversity of the geographical, social or even personal context for which the data was created, and (ii) to reduce the cost of data reuse, which is essential to support the development of data-based applications. However, the data currently available is not suitable to fully satisfy such requirements. Most of the time, data reuse is based on data transformation to better fit the purpose to be fulfilled, but the original diversity of information that the data carries is lost. This leads to a distortion of the reality represented by the reused data. A concrete case is the well-known AI services, which collect data from different local contexts, but transform this data to fit a unique, usually Western and business-oriented data model. This data reuse problem is exacerbated by a lack of data representation models that can concretely represent local information diversity. And a lack of methodologies to support data reuse. We propose iTelos, namely a methodology for data reuse based on the production, integration and distribution of diversity-aware Knowledge Graphs (KGs). iTelos is based on three main processes. The first aims at representing a new type of data, based on the Entity Base data model, capable of concretely representing the information diversity of a context. The second is a KG completion process, for the integration of diversity-aware data. The third one is a data distribution process, which aims to enhance the reuse of diversity-aware data by supporting the collaboration between those who produce data and those who are interested in using it, within a data distribution network called LiveData. The iTelos methodology supports both expert users in the creation of diversity-aware data and non-expert users in the development of data-based applications by providing them with the necessary data and support, and ultimately by establishing collaboration with expert users. The methodology has been tested in the Knowledge Graph Engineering (KGE) Masters course at the University of Trento (Italy), where students used iTelos to develop KGs suitable for solving several real-world problems. The first version of the LiveData network has also been developed by the University of Trento (Italy) and the National University of Mongolia (Ulaanbaatar, Mongolia).

Keywords: Information Diversity, Data Representation, Knowledge Representation, Data Integration, Knowledge Graphs, Data Distribution, Data Intermediary.

Acknowledgements

This work would never have been possible without the presence of some people and places who have changed my life in some way over the last few years. That is why I want to thank them below.

Thanks to my supervisor Fausto, whose words made me understand the meaning of the phrase 'the union is more than the sum of the parts'. Thanks to this, over the past few years, I have better understood my abilities which has allowed me, and will allow me, to make better use of them. I would also like to thank Fausto for teaching me to live the PhD as a personal growth process rather than the attainment of a degree. Thanks to this, many interesting scientific discussions have turned into journeys within myself, him, and other people I have met.

I thank the Knowdive research group and all the members who have been a part of it during these years. Starting with Gabor, who accompanied me in my first days of work in Trento, and then in the world of European projects. My office mates, Alessio, Andrea, Leonardo, Marcelo, Mayukh, Danish, Ali and Mugi, have been both sources of new knowledge and sources of lightness during the fun times spent together. I would also like to thank Prof Amarsanaa Ganbold from the National University of Mongolia for his incredible support during the months I spent in Ulaanbaatar. Thanks to him, I was able to learn about the amazing Mongolian culture and people.

Special thanks to Matteo, who over these years in Trento has not only become an indelible friend but has also managed, with very simple words and gestures, to make me see some wonderful aspects of myself and the people around us.

My heartfelt thanks to Ada, whom I had the good fortune to meet on my journey. Thank you to her for making me question myself like no one before, and for giving new meaning to the verb 'to love'. Thanks to her, I have better understood the importance of seeing the point of view of others.

An important thank you goes to my family and friends far away, in Piemonte and other parts of the world. I am lucky to have people who, although far away, know how to make their presence felt.

Finally, a 'thank you' under my breath goes to the place I have had the privilege to call 'home' over these years. To the mountains, lakes and trails that surround it. Thank you for giving me the lightness and strength I needed.

The research work in this thesis has been supported by:

InteropEHRate project, co-funded by the European Union (EU) Horizon 2020 programme under grant number 826106.

JIDEP project, founded by Horizon Europe Research and Innovation program under grant agreement number 101058732.

"DELPhi - DiscovEring Life Patterns" project funded by the MIUR Progetti di Ricerca di Rilevante Interesse Nazionale (PRIN) 2017 – DD n. 1062 del 31.05.2019.

Preface

How many times have we asked ourselves “Who am I ?” or “What am I able to do?”, “How can I improve myself and the world around me ?”. During the past years, in parallel with, and thanks to, my PhD research, I tried to answer the above questions. This preface aims to give an idea of what I’ve learnt and to provide a key for reading the following chapters.

We always act with the constant of our own "point of view" over ourselves and the world, which in the thesis will be defined as “The Purpose”. Our daily activities, at work and in our free time are led by a Purpose that somehow encodes our way to perceive ourselves, the world in which we live, and how we interact with it. We use our Purpose to transform what we perceive into our knowledge. The clearer our Purpose is, the more we can focus on the right things, people, events, and information that interest us. Not only that, but a clear Purpose allows us to reason about what we perceive and thus transform such perception into knowledge. In other words, the more you know who you are, the more you will understand what and how to do what you need to do. The Purpose is different for everyone, not only regarding the ideas we have (for example, I want to travel around the world to know foreign cultures, while my best friend’s objective is to actively participate in his village community, to maintain and support it) but also considering the way we perceive ourselves and the world around us. For example, as a PhD student, I can see the university as my workplace, while a master’s student who is also working in a company can see the university as a place to improve the skills required for her job in the company. We are both students, but we perceive ourselves differently as well as different is our perception of the same real-world entity (the university, in this case). I have learnt that it is very important to have doubts about our perceptions of other people, things, or events, it allows us to increase our vision (and knowledge) by perceiving not only our own point of view but also the others. Understanding someone else’s Purpose allows us to get closer to them. Even when we are fully committed to someone else’s initiative or objective, the way we perceive, interpret and subsequently act on that, is our own prerogative. We are the knowledge built on our Purpose. We build this knowledge over time, thanks to the continuos perception of ourselves, as well as with the interaction with the rest of the world. My house, a supermarket, a mountain, a festival, a university course; but also my heart, my hand, my body, my thoughts. All of this is part of me. However, from the moment we are born, we begin to perceive and transform the information that other people or the environment give us. Therefore, we can say that our own knowledge is generated by the combination of perceiving and transforming the knowledge of someone (or something) else. At this point, the first question becomes stronger: "Who

am I?". Am I the product of my own knowledge, or just a mixture of bits and pieces of other people's knowledge? In other words, are we the result of the world around us, or a unique part of it? At any moment in our lives, we perceive new knowledge, and we elaborate and learn about that. Considering this, it is impossible to define a person as a portion or a static composition of knowledge, since the knowledge defining us is dynamic and in constant motion. For this reason, my answer to the above question is, that we are our own way to build our knowledge, through the definition of our Purpose. We are not defined by static knowledge created uniquely by ourselves or taken from someone (or something) else, we are instead our own unique way to process such knowledge. Our learning process is more important than what we learn, thus enabling us to see the benefit of both errors and good results. The key ingredient of such a learning process is the Purpose. Not only our Purpose but also the others around us. Only by defining ourselves, we can clearly perceive ourselves and the world around us. But, we will never be fully aware of us without considering the others, in a continuous process of constructing us as individuals and as a society. What really matters is not the destination but the journey about me and you.

1 Introduction

Contents

1.1 Motivation

The use of data, and the way we see it, has changed over time along with the environments and Purposes for which data is used. In more detail, we can identify three different moments in time, or 'eras', where the concept of data, and its use accordingly, has changed the most.

The Corporate Data era. This first period started in a moment around 1970s, which can be also identified by the term "Industry 3.0", until the end of the '90s. In this period, the data was only used by those organizations and companies having the power to manage it, with the objective of improving their internal work management or providing services to their customers. Some example of actors and applications involved in this period are mobile phone companies or big companies adopting corporate Data Base Management Systems (DBMS) solutions to digitize (and partially automate) their work. The usage of Databases (DB), in this period was limited to a restricted context defined by the company having the technology and the expertise to handle it. The company, in this case, defined the requirements to be satisfied by the DB, as well as how to model the information within the DB itself. The representation of the data to be stored and maintained was defined by the company's Purpose and was not intended to be changed for reuse or secondary Purposes (like sharing the data with other companies for example). In such an environment, the modeling of a DB, its maintenance, as well as the DB exploitation through a DBMS, were all activities defined over a single Purpose (the company one), thus a single "point of view" over the

representation of the data to be used.

The (Semantic) Web era. The second period in which we identified a significant change in the data representation and usage, starts with the beginning of the 21st century (2000) and lasts little more than a decade (until 2012). It has been characterised by more widespread use of the Internet and the Web, which have become more available in these years. In this period a more massive usage (and availability) of Browsers, allowed more users to handle the data, thus effectively expanding the context in which the data is considered, from a narrow set of companies to a broader set of actors. The data becomes a way to discover the world, as an information source, but also a way to represent the world around us (Wikipedia, e-commerce, web journals). The single "point of view" leading the data usage and representation, turned into a variety of interpretations of the world captured by different data representations. The advent of multiple data representations related to data exchange, data retrieval, and data exploitation led, as a natural consequence, to the need to understand the different interpretations of the real-world entities represented by the data. As a consequence of such a diversified environment, the contents available on the Web changed becoming more sensitive to the representation of the semantic of data. The Semantic Web is defined, in this period, by Tim Berners-Lee, as a Web of data that can be processed by machine, or as he wrote "The Semantic Web is a web of data, in some ways like a global database" [7]. To support such a new vision of the Web, Knowledge Representation (KR) [21] became the key research area, and knowledge models like ontologies, Knowledge Bases (KBs), as well as the newest Knowledge Graphs (KGs), started to be largely used to express the data semantic, thus actually providing a different and more informative way to define real-world entities (see section 2 for more details about KR and KGs). A huge amount of information, and data, that before was privately maintained in corporate environments, becomes in this period available and "browsable", by any user having access to the Web. As a consequence, the reuse of existing data starts to have a strong impact on the development of new applications and services. In this period, different types of applications were born to support people's daily lives (mobile applications like Google Maps and social networks). The web era has changed the way data is presented and improved the availability of data, extending it to individuals. However, the technology capable of handling this new type of data remains in the hands of a few actors, such as companies, universities, and research centers, which have the resources and expertise to deal with it.

The Artificial Intelligence (AI) era. The last period we consider starts from 2012 with the advent of neural network usage and deep learning approaches applied to machine learning and, more in general, for AI applications. The AI era is currently ongoing with an increasing number of AI applications and

new approaches in AI research and innovation. If the research on AI was born in the middle '90s, its boom into real-world applications and services (Google Assistant, Amazon Alexa) started from the exploitation of Big Data and deep learning (2012-2017), and increased its power, together with the interest for research and business Purposes (ChatGPT, Microsoft Copilot), with the advent of Large Language Models (LLM), around 2020. In these years, the context of data use undergoes further expansion, not only thanks to a major amount of data available for an even wider set of individuals but also thanks to the technology used for data production and transformation, that becomes available to individuals too, thanks to AI services. In other words, the context in which the data is considered has become even wider, reaching more people with less technical experience. AI needs data to be trained and to learn from, for this reason, the re-use of existing data becomes even more important for the development and the accuracy of AI applications, but also for their capacity to deal with the diversity of the world. The Free Model (Google, Facebook), is an excellent example of such a data reuse need. It disrupts the market with an *"if-you are-not-paying-for-the-product-you-are-the-product"* ethos. To achieve this business model, vendors radically migrated their primary function from *"providing excellent products"* to *"harvesting data from excellent free products"* [70]. However, it is important to consider how the power of creating and reusing data is, now, in the hands of individuals, each having their own interpretation of the world. As a consequence, AI applications working on such data, have to be able to understand such interpretations, based on local geographical and social knowledge. In other words, the key requirement for AI applications has become to understand the "point of view" of their users, without obscuring it with more overbearing (and usually western, business-oriented) cultural views.

1.2 Problem

The three different historical moments described above introduce, at their time, specific needs to be satisfied regarding the representation and the management of data. The evolution of such needs along the years led to an always more informative form of data, suitable to be exploited by the relative applications and services. In parallel the problems of creating, transforming, and handling the data have changed themselves. More in detail, the problem of representing the data by using a different data model respects the one adopted in the previous "data era". As well as, the problem of reusing and integrating those new types of data. In this section, we report the evolution of such problems, by highlighting the most important difficulties to be addressed in the past and, most important, in the current time data representation and management environment.

In the corporate data era, where the data "lived" in more restricted and Purpose-specific environments, the main problem was to create data to support internal (relative to companies and organizations) tasks. The Purpose of the data application was quite clear, and the representation of the information through the data was, as a consequence, unambiguous due to the restricted context considered. The cost to be paid was, and it is still paid nowadays for similar applications, over the data generation and maintenance. It is important to mention that, even if the data models and the technologies are nowadays more advanced with respect to the 90s, the evolution and maintenance of data is still today one of the higher cost to be paid for the development of Purpose-specific apps and services. Nevertheless, in the corporate data era, the data representation did not consider a secondary use (or a data reuse), the data had to fit the requirements given by a single context and to satisfy a single (main) Purpose. This means that there was no need to consider multiple interpretations of the same real-world entity, due to a possible multi-Purpose application.

In the (semantic) Web era, instead, the multi-Purpose data representation introduced the need of representing the semantics of data. People from different parts of the world can access and search the information contained in the data. As a result, a real-world entity is no longer represented based on a single set of requirements, as in the corporate use case, but it is represented differently by users who live in different geographical and social contexts, or by users who live in the same context but having simply a different "point of view" (Purpose) over the entity to be represented. In other words, the environment in which the data "lives" has become more heterogeneous, thus leading towards more heterogeneous data representations. More in detail, such data heterogeneity appears differently depending on the type of data.

- *Language data*: the Web era has driven the use of the Web on a global scale. This implied also the representation of data in different natural languages. The heterogeneity of language data refers to the different ways of representing a concept in different languages. Such kind of data heterogeneity brought not only the need to translate the data for multilingual Purposes but also to represent the semantics expressed by data in a given language, by using a different language. For example, the concept identified by the term "Malga" in Italian means: "a relatively small alpine hut dedicated to the sheltering of cattle and deposit for milk and cheese"; however, it cannot be properly represented by a single word in English. A different representation may be required for the same concept expressed in two different languages.

- *Knowledge data*: the semantics of data is not only expressed by the meaning of its concept into a specific language but also by the way we

represent its information (real-world entities and entity types) through its properties. For example, a person can be represented through the following set of properties [name, surname, date-of-birth], but also with a different one, like [Identifier, date-of-birth]. The first representation can be quite efficient for a specific Purpose, while the second one can be adopted for Purposes where people's privacy needs to be more protected. Both the two data schema represent the entity "Person" but in two different ways. More formally the heterogeneity of knowledge data refers to the different data schema used to represent the same real-world entity. Such kind of data heterogeneity is also defined by the Entity Matching problem [69, 44].

- *Data value*: even assuming to use of a single language to represent data, as well as to agree on a single representation for each type of entity, the values associated with the entity properties could be represented in multiple ways. For example, the "date-of-birth" property, used to represent the "Person" entity type, can be associated with values expressed as timestamps, or by adopting the format ISO-8601[1] which represents date and time through a standardized format. The data heterogeneity in the values of data refers to the different ways that can be used to represent a data value. While the knowledge data heterogeneity leads to multiple ways to represent an entity type, the data value heterogeneity introduces multiple possibilities to define the entities instantiating a single entity type. It is important to notice how the entity representation became stratified, having in particular a well-defined and separated knowledge layer (entity type) and data value layer (entity values).

- *Data source*: the last type of heterogeneity is not intrinsic to the data, but more related to how the data is provided and retrieved. The Web era introduced new sources from which the data can be collected. Nevertheless, the choice of the data source could implicitly define how the information carried by its data is represented, as well as the quality of such information. For example, collecting data provided by Wikipedia could be more reliable with respect to data collected from an unknown website. Moreover, different data sources could provide data contained in different dataset formats, like CSV, JSON, PDF, and many others. The dataset format can be more or less suitable for a specific Purpose (or data representation), and therefore more or less effort to consider for data exploitation.

The data heterogeneity, as defined above, introduced the need to handle the specific information carried by different types of data. In other words, data

[1] ISO-8601

is no longer just a value in a dataset, but also the meaning of the terms used to represent it, by using a specific language, as well as its relative set of data properties (data schema). The Semantic Web changed the way of thinking, and reasoning about real-world entities, by adopting ontologies and knowledge bases to express the data semantic. As a consequence of the increased data availability, data reuse became more important with the objective of reducing the cost of data production. However, data reuse implicitly requires the cost of integrating different data (data coming from different sources), a cost that is made more expensive by the heterogeneity of the data itself.

The Web era increased heavily the complexity of the data representation and data exploitation scenarios. In particular, the lack of methodologies and well-defined processes supporting the users in the transformation and integration of the data produced in the previous era into data suitable for the Semantic Web significantly hinders the data reuse activities and the development of data-based applications and services. Such a lack of methodologies for data reuse and data integration is still today a crucial issue to be addressed, especially for those applications requiring a high level of data interoperability. For example, we report the InteropEHRate [2] EU project [65, 13], where different EU countries having different interpretations of the healthcare systems, need to share data about patient's diseases, drugs, medical history and other. The differences in the healthcare data interpretations appear at multiple levels, involving the different natural languages, and the different medical systems used to encode names of drugs or diseases, but also with the differences in the dataset formats and different ways of representing the same information entities (i.e., different set of properties used to represent a patient health record). The development of such a project highlighted that information and data standards for the representation of healthcare information, like SNOMED[3] and LOINC[4], are not sufficient to guarantee the required data interoperability among the different healthcare systems involved. A common methodology, applied within different contexts (countries with different healthcare systems) to fit the local requirements, is necessary for representing and transforming the data into a more interoperable version, without losing the local information diversity.

In the Web era, the Web's content starts to be a large data source from which the data can be explored but also collected through scraping and downloading activities. If on the one hand, the large availability of data has been a benefit for the data reuse, on the other hand, the low quality and interoperability levels of the Web data require an additional pre-processing

[2] InteropEHRate
[3] SNOMED
[4] LOINC

cost to be paid, to be exploited. Even today, the majority of the data collected, or downloaded, from the Web must be adapted somehow before being used. Such kind of adaptation includes activities like data formatting, data cleaning to remove "data noise", as well as data transformation to fit the desired requirements. Open Data initiatives play a crucial role in partially covering such need for quality data distribution. However, these initiatives are not always (most of the time) effective in supporting the reuse of data. Data sharing is not just about making data downloadable, but also about ensuring data quality and interoperability, and making data findable and discoverable by the users who want it. As a consequence of such data distribution needs in the Web era two distinct roles appeared:

- *The data consumer*: people interested in collecting data and exploiting it for the development of Purpose-specific applications and services. Often, the data consumers do not have technical expertise, thus making it difficult for them to address the pre-processing activities required to handle the Web data, as well as to exploit the data itself for the implementation of the desired application.
- *The data producer*: people having data management expertise (i.e., computer scientists, data and knowledge engineers), thus having the ability to create new data, or reuse existing data by addressing the required pre-processing activities. Data producers have the ability to produce high-quality and interoperable data and then distribute it on the Web, but often they lack a (application-specific) Purpose to motivate them to do so.

Usually the data producer and consumer are not interpreted by the same person. This situation generates a gap between the two roles. Such a gap is the cause of the high costs paid by both the producer and the consumer, for Purpose-specific data distribution and high-quality data production, respectively. Different Open Data initiatives tried to cover the gap between data producers and consumers [5] [6]. However, the lack of a concrete process able to guide the two roles actors close to each other requirements, hindered the achievement of a concrete and efficient solution to this problem.

If the Web era introduced the need to transform, integrate, and distribute data, the AI era clearly states such need as a mandatory requirement. With the advent of AI services and applications development, the demand for data increased enormously. Therefore, if the Semantic Web ages required new data models to structure the information (i.e., Knowledge Bases and Knowledge Graphs), the AI era needs to exploit such data scaling up its

[5] https://www.esri.com/it-it/home
[6] https://osmfoundation.org/

size and automating the generation of such models. Big Data became the main ingredient for the development of Machine Learning (ML) systems, Deep Learning implementations, Large Language Models (LLMs) [99], AI services, and real-time application development for many Purposes. On the other hand, AI technology became available to a larger set of individuals thus actually increasing the need to express their own local geographical and social information diversity through the data. It clearly shows how the support for AI should not be only focused on Big Data, but also on information and data diversity. The current advanced technologies, applying the research on AI, allow nowadays individuals to easily access tools supporting their daily life. Such tools are based on the reuse, and interpretation, of existing data to support specific user interests, in other words, a user Purpose. Like in the Web era, the data representation problem remains a crucial issue to be addressed, to the extent that now the data has to represent the interpretation of each user's Purpose. This means that, ideally, an optimal AI must consider the diversity of each of its users, as well as express the data diversity coming from each of them. Unlike the corporate and the Web eras, the data representation for AI cannot remain focused only on a domain of interest, a context, or a specific application, it has to support every single person's "point of view", who has access to the new AI technology. The interpretation of the information, thus the representation of data, processed and provided by AI technologies starts to be a problem affecting the representation of the reality [55]. Some of the currently available technologies are centralized, in the sense that they interpret the inputs, coming from different cultural and social context users, in a unique way based on a single contexts-specific information model. Such a model is in most cases unable to represent the diversity of the users who implicitly use it because the model builder does not have sufficient knowledge of all the geographical and social contexts from which the training data were collected. This kind of approach, based on a unique centralized data interpretation and representation, at one hand, allows for a reduction of the data integration and data reuse costs, from a procedural point of view. On the other hand, it is a distortion of reality that directly implies the loss of local information diversity characterizing a geographical and social context, as well as a group of people, or even single individuals, living in such contexts. The risk, very dangerous, is to cancel the local diversity with the adoption of a unique centralized data representation model. Another important risk associated with data reuse and data generation for AI systems, particularly for generative AI models, is the further concentration of the economy [55]. This means that, as reported by Huang et al., "if certain capital-intensive models are privately owned with limited or no outside access (e.g. Google's PaLM, DeepMind's Gopher, OpenAI's DALLE-2) or significant control over applications built on top, this could contribute to economic concentration. Access may be limited particularly if the high costs of creating GFMs (Generative Foundation Models) don't

decrease soon.".

Such a scenario enhances even more the need for concrete methodologies and processes for the three problems identified in the previous years (Web era) that now become crucial to support the development of AI-based applications and services. More in detail there is the need for:

- a precise process of data representation supporting a transformation of data towards the creation of more stratified data (including multiple types of data, like language, knowledge and data values) and diversity-aware data able to express the local diversity for each, domain, context, application, and even single user. Such a process has to enhance the production of a new generation of data, where the heterogeneity is not considered a bug but instead a feature to be exploited to highlight local diversity.
- A process of data integration that, by exploiting the above data representation process, allows for the integration of a larger amount of data produced for different Purposes and with different levels of quality and interoperability.
- A process of data distribution supporting the reuse of data by covering the gap between the data producer and the data consumer roles. This process should exploit the diversity-aware data representation as well as be strictly linked to the data integration process, with the objective of supporting the data production and data exploitation requirements.

The need to address the three problems listed above started during the shift from the corporate era to the Web era, and became a consolidated requirement in the AI ages. In our research, we aim to provide a methodological solution covering the three needs defined above.

1.3 Solution

In the previous section, we identified the problems of transforming, integrating, and distributing data to enhance the local diversity of the data users, all of which fall under the more general problem of reducing the cost of data reuse. The research reported in this thesis aims to provide a solution to the above problem, by defining a methodology that includes the design of three main processes encapsulated with each other, for data representation, data integration, and data distribution respectively. We propose iTelos [37, 11, 13, 39, 38] namely a methodology for data reuse based on the production, integration, and distribution of diversity-aware Knowledge Graphs (KGs). iTelos is a structured methodology that can be defined by three different

processes encapsulated with each other progressively as described here below.

The data representation process. The first process included in the iTelos methodology aims at supporting data reuse by transforming existing data into diversity-aware data. The input data considered by this process is the data originating from the period we defined above as the corporate era (mostly single-layer, non-stratified data, like tabular data values set), which is however the type of data that continues to be most used for several Purposes. The data produced by this process, instead, is the diversity-aware data, namely a novel type of stratified data [45] that better represents the diversity of the local geographical, social and individual (human) context in which the data has been produced, or where it needs to be exploited. More in detail, the process aims at generating data based on the Entity Base (EB) data model. This data model aims to represent concretely the real-world entities shaped as interlinked nodes of a KG. To this end, the data model defines an entity through three different components or layers, namely the entity layers of language, knowledge, and data values, which concretely represent the diversity of the respective entity as expressed by the data heterogeneity that appears within three types of data - language data, knowledge data, and data values, respectively. Each entity layer is shaped as a graph, and composed together they define an EB object, in which an additional layer represents the Purpose for which it has been created, thus contextualizing the information diversity for each of the three layers mentioned above. The idea behind the EB data model is to provide a representation of real-world entities closer to the way humans have to think and reason about it, instead of providing an entity representation that fits the requirements of a DBMS or a software application. The data representation process aims at transforming the data heterogeneity, initially seen as an issue to be addressed, into local diversity, seen as an exploitable data feature. Therefore, the process enhances the reuse of data in support of Web-based and AI-based applications which require a more stratified type of data, able to capture the local diversity of a given context or individual.

The Knowledge Graph Completion process. The second process included in the iTelos methodology is a KG Completion (KGC) process. It represents the core of the iTelos methodology by defining the structure of the different steps, or phases, to be executed with the objective of reusing and integrating different types of data into one, or a set of KGs. The KGC process, provided by the iTelos methodology, uses the above described data representation process by adopting the EB data model as the final representation of the KG it creates. While the data representation process provides the data model for diversity-aware data, the iTelos KGC process encapsulates the EB data model into a more concrete process supported by a dedicated framework.

It guides the users along four well-defined phases where a (diversity-aware) KG is created starting from the user's Purpose, formalized into a set of user requirements to be satisfied by exploiting the final process output, and a set of already existing data. More in detail the four phases of the iTleos KGC process are focused on: (i) the formal definition of the user Purpose, as a knowledge model specifying the requirements to be satisfied by the final KG; (ii) the handling of the input data heterogeneity at language level, with the generation of the language entity layer; (iii) the handling of the input data heterogeneity at knowledge level, with the generation of the knowledge entity layer; (iv) the handling of the input data heterogeneity at data value level, with the generation of the data value entity layer. Specific activities and dedicated tools support the KGC process users in the execution of each phase. Moreover, the iTelos KGC process supports both the two types of roles that a user (or different users) might interpret as a data producer or a data consumer. When the KGC process is used as a data producer, the input data are collected from "external" data sources, where "external" means: data source not certified by the iTelos methodology, thus not compliant with the data quality and interoperability standards, as well as not distributing diversity-aware data. The KGC process output is then uploaded into the iTelos official data distribution source (or internal source), called *LiveData* [12], namely a worldwide diversity-aware data space. Therefore, the iTelos data producer uses the iTelos KGC process to get "external" data and to transform it into diversity-aware data that is then pushed into the "internal" LiveData space. When, instead, the KGC process is used as a data consumer, the input data are collected from the LiveData space, and integrated together to satisfy specific application requirements. The input data in this case is already shaped as diversity-aware KGs. For this reason, data integration is cheaper, in terms of pre-processing and integration, on the consumer side than on the producer side. It is important to mention that, also the output of the KGC process used by the iTelos data consumer can also be pushed into the LiveData space, as it is a new diversity-aware data resulting from the composition of the consumer process inputs.

The data intermediary process. The third process defined by the iTelos methodology is a process designed to support the generation and the reuse of data, by exploiting the process and tools provided by the iTelos KGC process. In particular, it is based on the role of the Data Intermediary (DI). The DI is defined in different ways in [76]. As it is better detailed in chapter 5 We have been inspired by such DI definitions, but by DI we mean an agent (a human actor that can be semi-automated by an agent-based system) that aims to provide a concrete solution to the problem of data reuse by bridging the gap between the roles of data producer and data consumer. The iTelos DI offers the methodology users the required support to achieve their objective and satisfy their own Purposes, both for the data producer

and data consumer side, by providing access to the iTelos KGC process for the production or integration of diversity-aware data, respectively. The iTelos DI is responsible for the management of the LiveData space, namely a network for the distribution and retrieval of diversity-aware data. When the DI supports a data producer user, it will allow the user to upload and eventually publish, the output diversity-aware data into the LiveData space. On the other hand, when the DI is supporting a data consumer user, it will allow the user to get the required data from the LiveData space, as well as to upload and publish the results at the end of the process. Moreover, the iTelos DI aims at improving the relationships between the data producer users and the data consumers, by helping the data consumers find the data producer who can generate the diversity-aware data required to support the consumer's Purpose. In other words, the DI aims at creating links between who has the expertise of producing and managing data, and who has the Purpose-specific idea for the development of applications and services. For this reason, the iTelos methodology includes the DI process, which defines how to navigate "external" and "internal'" (LiveData) data spaces to obtain the data required for a given Purpose. But also how to move the data (store, publish, and distribute) to enable global collaboration on data reuse, which can drastically reduce the cost of dealing with data for applications and services development.

The iTelos methodology is the clue linking together the information representation, the transformation and integration of the data carrying it, and the data distribution without which the reuse of data, necessary to reduce the overall cost of data-based applications, would not be sustainable. The whole methodology aims at reusing existing data for the generation and distribution of Purpose-specific diversity-aware KGs. It is important to notice how the cost reduction given by data reuse becomes impactful only if a distribution of quality and interoperable data is considered in the overall process. The iTelos methodology defines a reuse and share approach over interoperable diversity-aware data, which aims to create over time a worldwide network where diversity-aware data is produced and shared through the collaboration of the users participating in that, with their local interest. In other words, iTelos provides a methodology for the development of Purpose (user) specific objectives (applications and services), through participation in a cooperative community of data users. The three components of the iTelos methodology, described above, are detailed in chapters number 3, 4, and 5 of this thesis.

1.4 Research Publications

The following research publications and pre-prints have been considered in the ideas expressed in the thesis. The results of the following works have

been integrated into the research described in this thesis.

- Simone Bocca, Amarsanaa Ganbold, and Tsolmon Zundui. "Live-Data–A Worldwide Data Mesh for Stratified Data". In: arXiv preprint arXiv:2407.00036 (2024)
- Simone Bocca, Gábor Bella, and Fausto Giunchiglia. "Interoperating Vocabularies in Multilingual Domain Data". In: 2022 IEEE 24th Conference on Business Informatics (CBI). Vol. 2. IEEE. 2022, pp. 73–7.
- Simone Bocca, Mauro Dragoni, and Fausto Giunchiglia. "iTelos - Case studies in building domain specific knowledge graphs". In: ISIC-22 International Semantic Intelligence Conference. 2022.
- Simone Bocca, Alessio Zamboni, et al. "Building Interoperable Electronic Health Records as Purpose Driven Knowledge Graphs". In: Data Science and Artificial Intelligence for Digital Healthcare: Communications Technologies for Epidemic Models. Springer, 2024, pp. 237–254.
- Fausto Giunchiglia, Simone Bocca, et al. "iTelos–Purpose Driven Knowledge Graph Generation". In: arXiv preprint arXiv:2105.09418.
- Fausto Giunchiglia, Simone Bocca, et al. "iTelos-Building reusable knowledge graphs". In: arXiv preprint arXiv:2105.09418 (2021).
- Fausto Giunchiglia, Simone Bocca, et al. "Popularity driven data integration". In: Iberoamerican Knowledge Graphs and Semantic Web Conference. Springer. 2022, pp. 277–284.
- Fausto Giunchiglia, Alessio Zamboni, Mayukh Bagchi, and Simone Bocca. "Stratified data integration". In: arXiv preprint arXiv:2105.09432 (2021).

2

Background & Related Work

Contents

The research work of this thesis aims to define a methodology to enable the generation of diversity-aware interoperable data, shaped as Knowledge Graphs (KGs), as well as to reduce the cost of their reuse and integration. To this end, in this chapter, we have studied the background and related work in terms of the history and current state of data representation, data integration, and data distribution, which are the key aspects influencing the proposed methodology. In addition, the history and current state of data reuse were studied first, since data reuse is the basic principle of the iTelos methodology, which includes data representation, integration, and distribution.

2.1 Data reuse

The entire iTelos methodology is based on the reuse of data. Data reuse is the key principle underlying the three main iTelos processes. First, the data representation process transforms existing data into diversity-aware data by following the Entity Base (EB) data model. Second, the iTelos KGC process, aims at reusing as much as possible already existing (diversity-aware and non-diversity-aware) data, to produce, and integrate, diversity-aware KGs. Last but not least, the data intermediary process makes data reuse the fundamental objective through a high-quality distribution of diversity-aware data. For this reason, the design of the iTelos methodology required a study of the reuse of data, to better understand the benefits and the issues to be addressed.

Academic research was the first sector to consider the re-use of data, in the first half of the twentieth century, as reported by a recent work by Mannheimer, S. [73], which addresses the problem of data reuse for big social research. Researchers began reusing survey data in an effort to "save time, money, careers, degrees, research interest, vitality, and talent, self-images and myriads of data from untimely, unnecessary, and unfortunate loss [46]. As early as 1962, Glaser wrote about the reanalysis of observation notes, unstructured interviews, and documents [47], thus indicating a data (re)interpretation activity for a secondary reuse (a secondary Purpose). However, as already mentioned in section 1.1, in the period identified as the corporate era ('70s to '90s), data reuse was not strongly considered by other sectors involving big data users, like companies and governments. The reuse of data became a common practice and continued to grow through the 1990s and 2000s [73], thus becoming an activity more involved in the period identified as the Web era (see section 1.1). In this period different benefits have been identified from different points of view. The scientific benefits of data reuse (through data sharing) are: (i) building new knowledge, new hypotheses, new methodologies, comparative research, and critiquing or strengthening existing theories [24]; (ii) promoting interdisciplinary use of data; (iii) providing data for teaching purposes (educational data) [9]. Between the moral benefits, we can mention the transparency and accountability of data, reached by sharing and making such data publicly available. The reuse of data has also major economic benefits by avoiding duplication of effort and allowing the conservation of time and resources, therefore supporting a higher return on investment [73] about the development of information systems. As mentioned in section 1.1, the above benefits of data reuse became even more impactful in the period identified as the AI era (from 2012 onwards), where the reuse of data has been strongly considered for the training of machine learning and deep learning models, first, and for generative AI and LLM afterward [55]. The issues to be addressed for the reuse of data need to be considered as well. In particular, we report here below three main criticalities of the reuse of data.

The context of data. As reported in [73], the reuse of data for qualitative research, but also for other types of secondary purposes, involves the analysis and understanding of the context in which such data were generated, as well as the study of the subjects described by the data, or using the data. As Hinds et al. [53] writes, "context is a source of data, meaning, and understanding... Ignoring context, underusing it, or not recognizing one's own context-driven perspective will result in incomplete or missed meaning and a misunderstanding of human phenomena.". Again, Pasquetto et al. [82] state that "removing data from their original context necessarily involves information loss". The works cited above reported a need to keep the data to be reused and connected with the information about the context in which

such data has been produced. However, it is often the case that it is difficult to extract contextual information from the data published or distributed for secondary purposes, resulting in a loss of information (a loss of information diversity about the relative context) and reducing the reusability of such data.

The paradox of reuse [75]. This phenomenon is mentioned in relation to generative AI systems whose use can be a disincentive for humans to contribute to the open digital ecosystem [55]. Meaning by this, as described by Huang et al., that "people may stop producing text/code/images in favor of using generative models, or never visit the websites from which data is sourced (i.e., Reddit or Wikipedia) in favor of using generative AI systems as search engines, and thus lead to lower contributions to them". In such a case, the reuse of AI-generated data leads to a reduction in the reuse of data collected from a greater variety of data sources that better express their own local diversity, as opposed to the AI that generates data based on a central, unique interpretation.

Data reuse process. Data reuse is usually more complicated than expected. By considering also the criticalities mentioned above, it appears that the reuse of data is a composition of different activities that need to be organized and properly executed. Different studies have been conducted over the steps, or phases, of data reuse [97, 83]. The activities involved in the reuse of data are often more specifically defined by the objective (Purpose) of the reuse. The integration of data is a task strongly involved in the reuse of data. It is possible to consider data integration as a structured process enabling the reuse of data. In [22] is described how the back-end of Business Intelligence (BI) technologies is constituted by a data integration pipeline for populating data warehouses (or other types of BI data management architectures) by extracting data from distributed and usually heterogeneous operational sources; cleansing, integrating and transforming the data; loading it into the data warehouse. Extract-Transform-Load (ETL) techniques are heavily involved in such integration processes [58]. The implementation and maintenance of ETL pipelines is often a huge fraction of the effort to be considered in BI projects. With the advent of AI-based systems, the development of real-time applications and services has increased significantly. As a result, data reuse and integration processes need to be scaled up to address the increasing complexity of the entire pipeline. It clearly appears that the reuse of data cannot properly scale up without a methodology able to adapt itself to the Purpose of the reuse, by supporting the users with the automation required by the real-time applications, as well as by keeping the human (data domain expert) in the loop to avoid the loss of the data diversity. A methodology that does not seem to be available at present.

2.2 Data representation

The design of the iTelos methodology required a study of the representation of the data, being the data representation is the key to encoding the information to be reused and exploited by applications, services, and human users. The iTelos methodology proposes a novel data representation model based on the stratification of the information. In other words, the methodology aims at representing, at different layers, the information about real-world entities [45]. Nevertheless, the definition of multiple information layers, denoting different types of information (like linguistic information, ontological or structural information, and the information carried by the values of data), is not a novelty defined in this research work. The novelty introduced by the iTelos methodology is to consider those layers together by concretely defining a new data model to represent real-world entities. A lot of research has been conducted on language data definition, and knowledge representation, as well as about data formats and data standards. In the following subsection, we provide the current state and related work about the three types of data mentioned above, as well as the definition and usages of KG, as the data structure used by the iTelos methodology to represent the diversity-aware data.

2.2.1 Language data

Research on language data representation is broad and covers multiple purposes, like information representation, information translation, natural language processing (NLP), formal language definition, word sense disambiguation, and others. However, the objective of the iTelos methodology is to represent the language diversity of real-world entities as data that compose the language layer of an EB. To this end, we focused the study of language data representation over the definition of language data structures able to represent and disambiguate the meaning of the terms (or words) used to express the real-world entities, and their properties, represented within a generic dataset. A more precise description of such kind of language data representation has been provided by Bella et al. [6] defining the *Language of Data* (LoD), namely languages formed by short labels typically found within structured datasets. The disambiguation [93, 94] of the terms used within the structured dataset, is one of the key issues to be solved in the iTelos data representation process (see chapter 3), as well as in the iTelos KGC process (see chapter 4). As better described in [10], the labels that compose the LoD of generic datasets can be classified within two main categories of languages, or vocabularies, being focused in this case over the word representation only, without considering a grammar required to formally define a language. The first category contains the labels to be parsed as natural language vocabulary (i.e., Italian, English, Mongolian) labels [98]. The

second category contains the labels coming from domain vocabulary, namely Controlled Natural Languages (CNL) [90] for the relative domain of interest. Some examples of domain vocabularies can be found in the healthcare sector (and many other areas) where specific terminologies are adopted are used to identify and describe health measurements, observations, and documents, but also drugs and diseases (i.e., LOINC[1], SNOMED[2]).

The representation of the terms defining a LoD has been addressed by using different kinds of resources, such as large-scale lexical structures like WordNet [78]. This kind of lexical architecture is focused on the representation of concepts through their lexicalizations [79]. For example, in WordNet [3], the word "house" can be used to represent different concepts. The house as "*a dwelling that serves as living quarters for one or more families*" (WordNet concept identifier: house.n.01), or the house as "*aristocratic family line*" (WordNet concept identifier: house.n.06). On the other side, a concept can be represented by different words, sometimes considering different natural languages or domain vocabularies. For example, the WordNet concept, used above, identified by "house.n.01", can be represented by the English word "house", and by the Italian word "casa". In addition, such linguistic architectures are hierarchically structured to define more general (hypernym or supertype) or more specific (hyponym) concepts. For example, by using always the same example, the concept represented by the word "house" is a more specific concept than the concept represented by the word "building", and vice versa, the concept "building" is more general than the concept "house".

By considering this kind of linguistic architectures, it appears how the analysis, as well as the disambiguation, of a word extracted from a dataset it always requires an additional contextual check over the vocabularies semantic structures, to identify the right concept, the one to be associated to the dataset's word. This is because, as stated by Binding C. et al. in [8], "*The requirement is concept alignment not term alignment*". As reported again in [8], different criteria con be used for such a semantic check. The analysis of concept's scope notes, if present in the linguistic data structure, is used to understand if two concepts are similar, as well as the analysis of synonyms, ancestors and descendants, of the concept examined. Nevertheless, the existing solutions always need to pass through the concept's lexicalisation in order to verify its semantic. This is because, within the vocabularies semantic structures, the concepts are defined by their lexicalisations. The complexity of such solutions, increases with the number of concept's lexicali-

[1] https://loinc.org/
[2] https://www.snomed.org/
[3] WordNet Explorer

sations to be checked. In the worst case, all words associated with a concept have to be checked in order to correctly identify that concept. It is clear how the choice of the right semantic structure to represent the LoD of a dataset becomes crucial to support concept identification or word disambiguation processes (as in the case of the iTelos methodology).

A very efficient linguistic architecture has been defined to mitigate the above described complexity. A large-scale multilingual lexical resource, called Universal Knowledge Core (UKC) [36, 35]. The key idea in the UKC architecture is that the linguistic information is split into two concretely different sub-structures: (i) the conceptual one, containing the concept expressed by a word, which is not dependent on any domain vocabulary or natural language; and (ii) the linguistic one, containing the lexical representations of the concept, namely the words, thus based on domain vocabularies and natural languages. As separated sub-structures, within the UKC knowledge base, each concept is associated with one, or more, linguistic elements (words). Each concept is defined separately from its linguistic part through a numeric id, and an international English description called *gloss*. The gloss aims to provide the meaning of a concept, allowing UKC users to recognize a specific concept without considering all its possible lexicalizations. Being a multilingual resource, the UKC requires a reference natural language for each word, thus allowing the definition of several translations (as different words) for a single concept's word. Following this structure, within the UKC architecture, we can identify two different sets of elements related to each other. The first, called UKC Concept Core (UKC-CC), collects all the concepts present in the UKC. The second, called UKC Language Core (UKC-LC), includes all the possible words used to represent the elements in UKC-CC. Thanks to the separation between concepts and their representation, the contextual semantic check required concept identification and word disambiguation processes, is focused on the UKC-CC only, thus reducing the complexity of dealing with the multiple concepts lexicalizations. For this reason, the UKC has been selected as linguistic architecture to represent the language data produced by the iTelos methodology.

2.2.2 Knowledge data

Knowledge Representation (KR) and Reasoning (KRR, KR&R, or KR^2) [21] is a field studying how to represent the information about real-world entities in a form that a computer system can use to solve complex tasks. This research field started to have a strong impact with the advent of the Semantic Web [7] in the period described above as the Web era (see section 1.1). The KRR was then widely used to model the information that AI systems would use in the AI era mentioned above (see section 1.1). Unlike the research about language data representation, which focuses on the linguistic

part of information, used to assign meaning to real entities, knowledge (data) representation is focused on structuring or schematising classes or types of real-world entities, called Entity Types (ETypes). ETypes are defined by a set of properties to represent classes of entities that instantiate their respective ETypes. Such properties can be used to define characteristics of the single EType (data properties), or relationships with other ETypes (object properties). Ontologies are special kinds of information objects or computational artifacts, used to formally model the structure of a system, i.e., the relevant entities and relations that emerge from its observation, and which are useful to our purposes [49]. Therefore, ontologies are used to model ETypes, thus structuring the real-world entities as they are perceived by an ontology producer having a specific Purpose, or point of view over the entities she wants to represent. Based on the above definition, it appears that ontologies can be used to define different types of information. In particular, during the study of the knowledge data representation required by the iTelos methodology, we focused on three categories of ontologies, described here below.

- *Top-level (Upper) ontologies*: the ontologies included in this category are used to model foundational knowledge, namely very general ontological notions to be used for a large variety of Purposes. One of the most famous top-level ontologies is SUMO [80] intended as a foundation ontology for a variety of computer information processing systems. Some examples of the most famous top-level ontologies, about different domains of interest, are listed in the Wikipedia page describing upper ontologies [4]. Some work [33] has explored the connection and integration of top-level ontologies with large lexical databases such as WordNet or the UKC, the latter being a way of giving understandable names to ontological entities, thus aligning the lexical meaning provided by lexical database concepts with the semantic structures of top level ontologies. For this reason, we consider the UKC Language Ontology (LO) as top-level ontology to model the diversity-aware data produced by the iTelos methodology.

- *Domain-level ontologies*: the ontologies included in this category are used to model reference knowledge about specific domains of interest. These ontologies are more specific than the top-level ones, however, they are still applied for different Purposes in the the relative domain of interest. Examples of domain-level ontologies, which have been explored and used over the years during the research work for the definition of the iTelos methodology, are the HL7 [5] FHIR ontology [6], for

[4] Wikipedia - Upper Ontologies
[5] https://www.hl7.org/fhir/
[6] FHIR ontology

the healthcare domain, adopted within the InteropEHRate EU project [7], and the EMMO ontology [8], in the material domain, adopted in the JIDEP EU project [9]. Moreover, the EMMO ontology has been considered to model the Material Passport Ontology [91] adopted within the JIDEP project.

- *Purpose-specific ontologies*: unlike those described above, the ontologies included in this category model Purpose-specific knowledge. These ontologies are adopted to satisfy very specific requirements. Unlike top-level and domain-level ontologies, purpose-specific ontologies are very performant for the purpose for which they were modelled, but they are less reusable for other purposes. Even if not recognized formally as ontologies, an example of this kind of knowledge representation is the modelling of DB schema, usually very specific to the DB context and to the DBMS used. Since this particular type of ontology is strictly related to the Purpose for which they have to be used, important research work has been done on the definition of purpose within an ontological model. In [42] Giunchiglia et al. propose *Teleologies* as an ontological model to encode the purpose as knowledge representation. This work has been previously inspired by the research over context modelling [34], in which a context is defined as: "*subset of the complete state of an individual that is used for reasoning about a given goal*". And subsequently by the idea of modelling knowledge as a set of contexts [74, 43].

The related work about ontologies, and more in general about knowledge representation, was born from the need to include, as part of the iTelos methodology, an ontology engineering process able to consider the integration of different types of ontologies, to build the EB knowledge layer. To this end, existing ontology modelling methodologies have been studied [30, 29], to understand how to model the information diversity, intrinsic to a specific user's Purpose. But also how to model ontologies that can be reused in different contexts and for different purposes, thus reducing the overall cost of knowledge representation when modelling an EB. Although the contribution of KR work has been crucial to the iTelos methodology, we have not identified studies on the composition of knowledge, language and data value representation into a unique KGs engineering process.

2.2.3 Data values

The third type of data considered by the iTelos methodology is the datasets carrying data values. More specifically, as already described in section 1.2,

[7] InteropEHRate
[8] EMMO ontology
[9] https://www.jidep.eu/

these datasets contain the values to be associated with the ETypes properties in order to instantiate the corresponding Etype with concrete entity representations. This type of data is the input of the iTelos methodology since one of the objectives of the methodology is to reuse such data by turning that into diversity-aware KGs. The analysis of the background and related work about data value representation has been focused on methods and standards to make data interoperable and reusable. In particular, the FAIR principles [10] for the definition of data to be Findable, Accessible, Interoperable, and Reusable, have been strongly considered for the design of the iTelos methodology. FAIR principles make an important contribution to the development of high-quality resources. A report of the European Commission, about the Cost-Benefit analysis for FAIR research data [20], reported the estimated cost of not having FAIR data within important activities included in the research data lifecycle, such as data collection, analysis of data, data integration, data pre-processing, data publication and peer review. The estimated cost reported by the study is €10.2bn per year. Another very important data model to be considered to produce high-quality data value datasets is the 5 ⋆ (5 stars) Open Data model [11]. Defined by Tim Berners-Lee, the 5 ⋆ model is an open data deployment scheme that aims to improve the functionality of open data. Although such a scheme focuses on open data, it contains important data representation requirements that non-open data must also meet to improve the interoperability and reusability of the data itself. Between them we can mention the need to represent data by using standard datasets and data value formats, thus allowing more applications and services to exploit such data. Moreover, the usage of URIs denotes things (entities, and entity's values) within datasets. The use of identifiers allows specific elements within the data to be uniquely recognised and referenced from other datasets. We have been inspired by the use of identifiers to improve the quality of data and, in particular, we have applied URI identification to the other types of data considered by iTelos (language concepts and knowledge types). The further vision that this choice implies is the automation of the iTelos methodology, thanks to automatic recognition of the information elements identified by URIs at several levels (language, knowledge and data values). Other important contributions in the area of data representation have been made by projects that have defined a data representation standard in their own domain of interest, thus improving the reuse and interoperability of such data. A very important example of such a domain data representation standard is Open Street Map (OSM) [52], which defines a representation of geographic data that is currently adopted by a large number of applications and data-based services.

[10] FAIR principles
[11] 5-star open data

2.2.4 Knowledge Graphs

The key idea behind the iTelos data representation approach is to create a new type of data where the information diversity, expressed at different levels by the above mentioned data types, is combined into a single data structure. Since the period defined in section 1.1 as the Web era, language and knowledge data started to be represented by using graph structure, to better represent the relationships between the information elements (concepts, ETypes) the data carries. In addition, the graph-based representation of data is closer to the way humans need to think and reason about real-world entities than the data representation adopted by databases in the so-called corporate data era. As reported in [1], graph-based data models have some benefits like:

- "it allows for a more natural modelling of data. Graph structures are visible to the user and they allow a natural way of handling applications data, for example, hypertext or geographic data."
- "Graphs have the advantage of being able to keep all the information about an entity in a single node and showing related information by arcs connected to it."
- "Queries can refer directly to this graph structure. Associated with graphs are specific graph operations in the query language algebra, such as finding shortest paths, determining certain sub-graphs, and so forth. Explicit graphs and graph operations allow users to express a query at a high level of abstraction."

As a consequence of that, we studied the Knowledge Graphs (KGs) as data structures suitable to represent all the three types of data considered by the iTelos methodology, namely language, knowledge and data values. Since 2007, DBpedia [3] and Freebase [14] projects have been defined as graph-based knowledge repositories. However, the first case of KG officially recognized was the Google KG [12] built on DBpedia and Freebase among other sources. In describing Google's KG, Amit Singhal wrote "Introducing the Knowledge Graph: things, not strings", noting the shift from a data-based view to a human-based view. Various definitions of KG have been proposed over the years, without an official one being established. In [25], Ehrlinger et al. selected some of the representative definitions and demonstrated the lack of a common core. What has been done by Paulheim in [84], was to list the minimum set of characteristics that must be present to distinguish knowledge graphs from other knowledge collections. The definition proposed by Paulheim, considering such requirements, was the following:

[12] Google KG

"A knowledge graph (i) mainly describes real-world entities and their interrelations, organized in a graph, (ii) defines possible classes and relations of entities in a schema, (iii) allows for potentially interrelating arbitrary entities with each other and (iv) covers various topical domains."

In the above KG definition, as well as in the other definitions reported in [25], three elements always appear. The first is the presence of a schema that defines the structures of the entities (ETypes) considered by a KG. The second is the presence of concrete entities represented by their property values, which distinguishes a KG from an ontology where only the classes of entities are specified. The third element is the reference to the possible domains in which a KG can be used to represent entities. The last element is a more general reference to the always present Purpose in a data representation, thus including the definition of a KG. However, the concrete implementation of KGs lacks the representation of the domain, context or Purpose relative to the entities they aim to represent, thus limiting the definition of KG to its schema and entity layers, as formally summarised here below.

A Knowledge Graph KG, can be defined as follows:

$$KG = (E, D, R, A)$$

Where:

- E: is the set of ETypes.
- D: is the set of real-world Entities representation. The Entities are ETypes instantiation.
- R: is the set of properties used to denote the ETypes. The elements of R, can be properties related to a single EType, (data properties), or properties used to define relations among different ETypes (object properties).
- A: is the set of property's values denoting the attributes of the set of Entities. Each attribute, associated to one and only one EType property, instantiates the relative data/object property.

The adoption of KGs into different domains and for different Purposes is demonstrated by many examples. Frequently mentioned implementations are Wikidata, Yahoo's semantic search assistant tool Spark, Google's Knowledge Vault, Microsoft's Satori and Facebook's entity graph [77, 84, 95, 67].

2.3 Data integration

If the study of the data representation provided an overview of the types of data suitable to represent the information diversity into KGs, another key

aspect to be considered is how to integrate such different types of data. The study of existing Data Integration (DI) techniques has been instrumental in defining the iTelos KGC process as a process capable of addressing different types of integration requirements. This means considering integration between and across language, knowledge, data values and KG data. Thanks to the studies on Semantic Web, the Ontology-Based approach to DI (OBDI) is now considered for many applications. The core element of OBDI techniques is the representation of the information (or the data domain) of each data source involved. Ontologies have been largely used to represent such information, becoming one of the most used objects to share data models. OBDI approach aims to share the models of the data sources considered by the integration requirements, with the consequent objective of building a global data model (global ontology) able to provide a unified view of the data. The existing OBDI strategies can be collected within two different categories, called virtualization and materialization.

Virtualization strategies aim to provide a unique virtualization of the data source, without extraction and transformation of their data. That is why they are also known as Ontology Based Data Access (OBDA). Virtualization strategies build a global ontology on which the queries are posed, allowing software applications to exploit the integration result. The unified ontology's queries are then translated into queries on single sources data models, allowing effective data retrieval. In [96] Wache et al. define three different approaches to classify OBDA strategies. The *single-ontology* approach provides a single ontology to which all source data models are mapped. The single ontology is queried to access the different data sources. The second approach, called *multiple-ontology*, aims to build a local ontology on each data source, on which the queries are performed. The ontology's role is to facilitate the access to each data source. The third approach called *hybrid*, exploits the definition of a vocabulary shared among the local ontologies, defined as in the second approach, facilitating in this way the construction of queries over the multiple local ontologies. Later in [26] Ekaputra et al. improve the OBDA strategies classification by adding a fourth approach. Such an approach, called *Global-As-View (GAV) OBDI*, is based on the definition of both locals and global ontologies plus independent mappings defined between them, simplifying in this way the data transformation from the local ontologies into the global one exploited by the software applications to query the data. After the analysis of the four approaches, summarizing the current status of OBDA solutions, we recognized that the virtualization strategies are not suitable for the objective of this research, mainly for two key observations: (i) leaving out the single-ontology approach (having more high maintenance cost [26]), adopting OBDA strategies the user has to define multiple ontologies as well as the respective mappings between them (GAV OBDI). As a consequence there is a higher level of knowledge representation

expertise required to use such kind of DI strategies; (ii) the definition of the ontologies is strongly focused on the data sources considered (data-driven modelling), leading to a unified view of the information to integrate, more focused on the data sources then on the user's requirements. We consider this category of DI techniques more suitable to corporate situations, where already defined databases need to communicate and exchange data.

The second main category of DI techniques is called *Materialization strategies*. These techniques are based on Extraction-Transformation-Load (ETL) procedures used to extract the data to be integrated from the respective data sources. The subsequent objective is to create a single information structure able to maintain the heterogeneous data collected. Compared with virtualization strategies, materialization ones work directly on the data extracted from the sources, building a single global view (interpretation, with a subsequent representation) of the data, thus reducing the amount of work in building ontologies (and mappings which are not required anymore). Materialization strategies have to deal with the heterogeneity of the data extracted from different data sources. Such heterogeneity can be recognized on different levels within the datasets collected (as already mentioned in section 1.2), at language level (in case of cross-country data integration), at the conceptual level (when a real word entity is represented by different ETypes and properties in different datasets) and at single data values level (different data format and standards adopted by different datasets). In our previous work [45], we proposed a solution which aims to handle the different levels of heterogeneity through different Knowledge Graphs (KGs) which are then composed together to build a single object able to maintain and exploit heterogeneous data. More in general the usage of KGs increased a lot within the data integration community, thanks to their exploitability in many different domains of interest (see previous section 2.2.3), such as sustainable transportation [64], biomedical data [101] as well as disaster management [87], and many others. DI based on KG Completion (KGC) is a process that involves different sub-problems to be solved, which can be summarised within four classes: data collection, schema alignment/definition, entity mapping, and DI workflow. The current status of research in data integration, and its application in practice, is hindered by a lack of quality frameworks and tools able to address all the classes of problems listed above. As a consequence, researchers and practitioners (as well as data consumers interested in applications and services development) cannot improve their expertise in this field [4]. Very few solutions have been proposed to cover this gap, such as BigGorilla [17] an open-source DI platform developed in Python. BigGorilla provides an environment, divided into four main packages containing specific tools adopted to solve the above classified DI sub-problems. Another interesting and efficient solution is FarsBase [2], a KG built by the integration of different Persian language information sources, which involves information

extraction techniques, as well as entity linking features allowing the compositionality with other KGs. Other tools have been developed to solve specific problems. Among them, one of the most important is Karma [66], a data mapping tool allowing data scientists to map different datasets on a unique schema, as well as OpenRefine [57], an open source project providing tools for data cleaning and formatting.

After the analysis of the (few) existing solutions to the challenging KGC-based DI, we discovered that materialization DI strategies are more suitable for the objective of this research, but there are still some issues to be considered. Firstly, there is a considerable problem of accessibility to data integration platforms by users without technical skills [63]. Due to the need of data management and knowledge representation skills, it will be difficult for users without such expertise, to use a framework like BigGorilla. Secondly, as per our knowledge, there are no general-purpose methodologies to build KGs. More in detail, almost every existing frameworks and tools rely on a fixed domain of interest, having thus a fixed background knowledge adopted to build the KG's schema. In other words, there is always a fixed upper or domain-specific ontology, that is used on top to construct the KG's purpose-specific ontology for the integration requirements. Some approaches have been proposed considering as base knowledge a more general ontology, thus allowing suitability for more domains (like *Schema.org*[13]), by adding extensions to the standard schema to represent more domain-specific aspects to be considered for the integration objective [28]. Nevertheless, this approach produces KGs that are more specific and, thus less reusable in other domains, the higher the number of extensions added to the original standard schema (i.e. Schema.org). The lack of reusable resources is the cause of additional costs in building exploitable KGs, and in turn leads to the production of more specific and less reusable resources, enforcing a negative loop [38].

2.4 Data distribution

Data distribution is a fundamental requirement for data reuse. For this reason, the final topic addressed by the analysis of the background and related work required by this research work focuses on the key principles, actors and infrastructures for data distribution. Such analysis has been crucial to better understand how to share the diversity-aware data generated by the iTleos methodology, thus enhancing its reuse.

The distribution of data goes in parallel with the data reuse (see section 2.1). The need to reuse data directly involves the need to find, discover, and access the data to be exploited for different Purposes. While talking about

[13] https://schema.org/

data distribution an important distinction to be considered is between Open Data (OD) and non-OD. As reported in [54], "Open data is a 'philosophy' or 'strategy' that encourages mostly public organisations to release objective, factual, and non-person-specific data that is generated or collected through the delivery of public services, to anyone, with a possibility of further operation and integration, without any copyright restrictions". On the contrary, non-OD refers to data that is not publicly available, is usually produced by private services, and contains sensitive information that requires protection and security requirements that OD does not. Nevertheless, data distribution involves non-OD too, by considering privacy policies, appropriate data licenses, as well as dedicated distribution process ensuring the required data protection. To be considered is also the economic value of non-OD with respect to OD. Since, non-OD often carries sensitive information about citizens, companies, products, and other potentially private entities, it is actually considered a bargaining chip in the information market. It is well recognized that OD plays a primary role in research and innovation [68, 61, 72]. A particular case of OD is Open Government Data (OGD), namely data that is processed and published by government organisations with the aim of making it available for the development of the country through digital applications and data-based services provided for the benefit of citizens. More in detail, in [59] Janssen et al. report the benefits and the barriers of using OD and OGD. Between the benefits, they reported transparency over the information distributed, democratic accountability of data, more participation and self-empowerment of citizens (data users), new governmental services for citizens, improvement of citizen satisfaction, more visibility for the data provider, and stimulation of knowledge developments. From a more technical point of view, the benefits reported are: the ability to reuse data, thus not having to collect the same data again and counteracting unnecessary duplication and associated costs; easier access to data and discovery of data; creation of new data based on combining data (DI); external quality checks of data (validation); sustainability of data (no data loss); the ability to merge, integrate and mesh public and private data (non-OD). However, some of the barriers to using OD and OGD, as reported by Janssen et al. are: making public only non-value-adding data; no process for dealing with user input; lack of ability to discover the appropriate data; no access to the original data (only processed data); no explanation of the meaning of data; no information about the quality of the open data; even if data can be found, users might not be aware of its potential uses; data formats and data sets are too complex to handle and use easily; focus is on making use of single datasets, whereas the real value might come from combining various datasets; lack of knowledge to make use of or to make sense of data; no central portal or architecture. In terms of quality, non-OD is considered more reliable, as a consequence of a more precise data curation implemented by private organizations and users, having usually the resources and more

(economic) interest, in producing high-quality data. However, non-OD is less available or accessible with restricted and controlled access, thus supporting a smaller number of innovative applications and services. Considering both OD and non-OD, it becomes clear how difficult it is to establish the connections between the data itself and the use, or re-use, of it. As reported in [59]: "Open data on its own has little intrinsic value; the value is created by its use.". But if the data is not accessible or understandable to those who can use it, its value remains unexplored. Several barriers to the use of OD are caused by a lack of cooperation between data producers and users who need to develop applications and services using such data (data consumers).

One way that has been considered to address this issue and enhance the efficiency of data ecosystems, is based on the role of the so-called Data Intermediaries [89]. As described by Oliveira et al. in 2019 [88] "a data intermediary is a role that facilitates the use of data for other actors.". More details about data intermediaries have been provided by a report of the Centre for Data Ethics and Innovation (UK government), in 2021[14], where a more detailed definition of different types of data intermediaries is provided. To provide a more complete overview of the role of data intermediaries and the activities for which they are responsible, the definition of such types of data intermediaries is provided below.

- *Data trusts*: Provide fiduciary data stewardship on behalf of data subjects.
- *Data exchanges*: Operate as online data platforms where datasets can be advertised and accessed commercially or on a not-for-profit basis.
- *Personal Information Management (PIM) systems* [62]: Seek to give data subjects more control over their persona data. etc.). PIM activities are an effort to establish, use, and maintain a mapping between user's information and user's need.
- *Industrial data platforms*: Provide shared infrastructure to facilitate secure data sharing and analysis between companies.
- *Data custodians*: Enable privacy-protecting analysis or attribute checks of confidential data, for example, via the application of Privacy Enhancing Technologies (PETs).
- *Data cooperatives*: Enable shared data spaces controlled by data subjects.
- *Trusted third parties*: Assure those looking to access confidential datasets that the data is fit for purpose (e.g. in terms of quality or ethical standards).

[14] Unlocking the value of data: Exploring the role of data intermediaries

From the data intermediaries definitions reported above, it clearly appears how data intermediaries perform different activities, in different contexts. The data intermediary role can be played directly by the data user, if she has to handle personal information, like in the PIM systems case. However, by considering the above definition and according to recent work [92], the data intermediary is a third-party actors who provide specialized resources, like data protection, data exchange, data sharing/distribution and data storage. For this reason, the development of a business model, which is necessary for the sustainability of a data intermediary, is a matter of complex analysis, as reported in [60]. The complexity comes from the need to merge the interests of public and private sectors, when public data needs to be exploited by private data consumers, as well as on the opposite side when private data has to be integrated or used for the development of public data-based services. In a recent work [85], Pilshchikova et al. described how Non-Profit Organisations (NPO) playing the role of data intermediaries can reduce the user barriers and increase the reusability of OD (especially OGD). As reported in the paper, "NPOs help create spaces for the users to come together with potential data providers. For example, the Open Knowledge Belgium NPO organises a conference: "*It's a day where data publishers sit next to users, citizen developers and communities to network and to openly discuss the next steps in [...] Open Data.*" Creating spaces for such communication helps users share knowledge and access the data providers to discuss data needs in person, establishing more lasting connections.". Another important aspect of NPOs as data intermediaries, reported by Pilshchikova et al., is about the presence of the NPOs within the local context, thus being able to "navigate the country context and its legislation around OGD to know what is available, what is expected, and what can be demanded". In other words, NPOs that are not driven by personal economic interests not only improve the necessary connection and cooperation between data producers and data consumers in a specific context, but they are also more aware of the diversity of this local context. They are therefore more committed to enhancing this diversity in the data produced, used and reused in their respective contexts.

As defined by Oliveira et al. in [81], "the conceptual elements of the (data) ecosystem identified are resources, roles, actors, and relationships. Resources can be datasets, data-based software, and hardware, which may be exchanged individually or in combination, through relationship transactions. A role is a function of an actor in the ecosystem. Actors are autonomous entities such as businesses, public institutions, and individuals serving one or more specific roles. Relationships are the interactions among actors in the (data) ecosystem.". It becomes clear that data distribution, as part of this ecosystem, is not only about the data itself but also about the roles (data producers, data consumers, data intermediaries) and actors that deal with it, as well as the architectural systems required to support the data distribution. Actually,

there is no evidence of a base architectural model for systems dedicated to data distribution, which involves all the issues mentioned above. Currently, the most updated systems for data distribution are based on data catalogues. In [51] Guptill describes a data catalogue as "A collection of metadata records combined with data management and search tools.". This definition encompasses different architectural approaches, from the management and visualisation of internal (private) data indexes to more open and widely distributed architectures spread across the Internet. The functionalities of data catalogues range from simple listing of content to interaction with users seeking data, including other specific functionalities such as searching, ordering and downloading data, as well as collecting and processing users' comments and feedback on the data (to improve the quality of the data and the catalogue itself). Data catalogues are nowadays largely adopted for data distribution purposes. Very important example to be mentioned is the catalogue[15] of UK Data Archive[16] that is UK's largest collection of social, economic and population data for over 50 years. As indicated in [27], in February 2022, the European Commission published a document providing an overview of the state of play of the ten common European data spaces initially listed in the European strategy for data, together with several additional ones. The goal of that work was to "set out a vision for a single European data space, which it describes as "a genuine single market for data – open to data from across the world – where personal and non-personal data, including sensitive business data, are secure and businesses have easy access to high-quality industrial data, boosting growth and creating value" (European strategy for data [17] of February 2020). The development of data spaces and data catalogues is nowadays supported by data management systems, like CKAN[18] a powerful platform for cataloguing, storing and accessing datasets with a rich front-end, full API (for both data and catalogue), visualization tools. CKAN is the world's leading open-source data portal platform, adopted by several public and private organisations to manage and distribute their data.

Although the current background to data distribution is well studied and efficient solutions can be adopted, it seems that we still lack a precise and clear methodology for data distribution that can meet the needs of all actors involved. Even if platforms such as CKAN provide important solutions for data distribution, they require non-trivial technical expertise to be properly set up and used. Such expertise is usually lacking for non-specialist citizens or domain experts who might be interested in sharing their own data to support the improvement of applications and services based on their needs. It is important to note that the latter case is becoming more and more

[15] UK Data Archive catalogue
[16] UK Data Archive
[17] European strategy for data
[18] https://ckan.org/

common in the current era, defined above as the AI era. (see section 1.1).

3

The iTelos Data Representation Process

Contents

This chapter aims to describe the iTelos data representation process. This is the first of the three processes which compose the iTelos methodology. The main objective of such a process is to transform existing data, where data heterogeneity (as described in section 1.2) hinders its interpretation and reuse, into stratified diversity-aware data, namely a novel type of data where the different layers of data heterogeneity have been transformed into layers of information diversity. More in detail, the diversity-aware data concretely represents the diversity of the local geographical, social or individual (human) context in which (or by whom) the data has been produced, through four diversity layers, namely the language layer, the knowledge layer, the data value layer and a specific layer representing the Purpose for which the data has been created. In this chapter, we describe how the iTelos second main process of the iTelos methodology, namely the data representation process, represents the four diversity layers as four different Knowledge Graphs (KGs), which composed together provide a novel data model, namely the Entity Base (EB) data model, in which the stratified diversity is available to be (re)used.

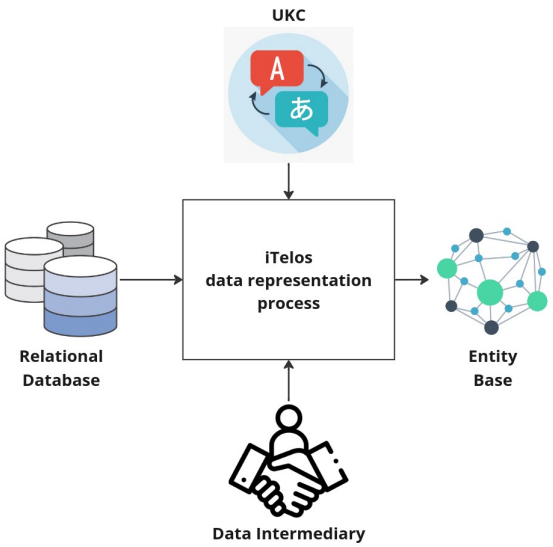


Figure 3.1: The iTelos data representation process applied to a relational database.

The key idea is that the novel type of data, represented by using the EB data model, can be more supportive for the applications and services working with AI technologies, being such data more rich of information able to capture the diversity of a context. Following such an idea, we decided to report in this chapter (section 3.1), the issues to be addressed to support such AI-based applications and services, by using relational data, and, as a consequence, the motivation to apply the iTelos data representation process to transform such a relational data into diversity-aware data represented by following the EB data model. However, the iTelos data representation process can be applied to transform different types of data models, therefore the considerations and the application of the same process principles can also be applied to non-relational data models. Figure 3.1 depicts how the iTelos data representation process is adopted as a key process to transform the data taken from a relational DB in input, into diversity-aware data following the EB data model. Moreover, the figure indicates the fundamental contribution given by the UKC (more detail about the UKC is provided in section 2.2.1) and the iTelos Data Intermediary (DI) through its data distribution network (detailed in chapter 5), for the generation of the output EB. The first section of this chapter (section 3.1) motivates the application of the iTelos data representation process over relational DBs, by highlighting the lack of diversity representation in the data handled by such kinds of DBs. The second section (section 3.2), instead, defines the EB data model as a solution for the above mentioned lack of diversity representation in relational DBs. Section 3.3 concludes this chapter with the definition of the represen-

tational process, showing how to concretely define the diversity of the entity extracted from a relational DB, into an EB.

3.1 Data heterogeneity in relational DBs

The objective of this section is to describe the motivation for applying the iTelos data representation process to relational data, to transform it into a concrete representation of the interested real-world entities, following the EB data model. As described in chapter 1, the need for such a transformation was born with the advent of what has been defined as the (Semantic) Web era, first, and grown up in the AI era (see section 1.1). Within these periods, the data are used, and re-used, in a wider range of applications and therefore have multiple representations depending on the context and Purpose for which they are adopted. Due to the increment of data availability and data-based applications, the reuse of data plays a crucial role in reducing the cost of applications and services development. Open data, data catalogues and other data distribution approaches are largely exploited to reduce such a cost, by reusing already existing data. However, most of the time, if not always, the reusable data is domain, context or Purpose-specific, thus bringing intrinsically the diversity of the geographical, social, or even individual, context in which (or by whom) such data has been created. The presence of such a data diversity raises the need to deal with data heterogeneity at different layers, as described in section 1.2. The data heterogeneity is the cause of a cost that cannot be ignored during data reuse. Such costs are expressed in terms of the time, tools and human effort required when the data diversity is not explicitly defined in the data, which then needs to be pre-processed by different actors (with respect to the data producer) who can implicitly replace the original data diversity with the one defined by their own point of view. In other words, the lack of a concrete representation of the data diversity hinders the possibility of interpreting and exploiting it. Here below, we examine how the phenomenon of data heterogeneity affects relational DBs at different levels, resulting in a lack of representation of data diversity, and thus a lack of information. More specifically we report how such a lack of diversity representation in relational DBs, can be considered over three diversity layers, namely at language layer, knowledge layer and data values layer. Then, in the following section, we define the Entity Base data model, as a solution to preserve and concretely represent such a diversity for each of the three layers mentioned above.

Language heterogeneity in relational DBs

The first layer affected by lack of a concrete diversity representation is the language layer. Often the construction of a DB, following the relational data model, starts with the definition of the requirements that such a DB

has to satisfy. This involves a domain expert (often the end user of the DB) working with someone with the technical skills to model and develop the DB structure. In such a case, the meanings of the terms used to name the relations (DB tables) and their attributes must be agreed between the two actors mentioned above. However, such terms, representing specific concepts adopted to define the information to be handled by the DB, are not concretely described in the DB itself. The terms are only used, without providing a concrete way to check their meaning to someone who did not participate in the DB development. The situation described above leads to a more complex reuse of the relational DB data, due to the difficulties of new data users in understanding the meaning of the terms used to define the DB information. A new user interested in reusing the relational DB data will always have her own interpretation of the term used in the DB. This interpretation may (as is usually the case) differ from the meaning of the terms intended by the domain experts and developers who created the relational DB. Such ambiguity, inherent in the terms used in the DB, represents data heterogeneity at the language layer, as described in section 1.2. However, it is important to note that the meaning of the DB's terms, provided by the DB's creators (domain experts and developers of the DB), represents their own "point of view" (their Purpose) about the information handled by the DB itself. In other words, the meanings of these terms define the language layer diversity specific to the data domain considered by the relational DB, as well as specific to the people who created the DB. Nevertheless, such language diversity is only implicitly defined, it remains in the minds of the DB creators, but it is not available or clearly understandable to new developers interested in reusing the relational DB's data.

Knowledge heterogeneity in relational DBs

We examined the relational DB's over a second layer to investigate the lack of a concrete diversity representation, namely the knowledge, or data schema, layer. This layer considers how the types of real-world entities are structured, in a heterogeneous way, within the relational DB's model [56]. The knowledge, or data schema layer, of a DB is the way it structures its information through a set of entities (or better Entity Types) with the relative properties, or attributes (as they are called in the Entity Relationship (ER) model specification). The modelling, and the subsequent development of a relational DB, are divided in two macro phases that we summarize as follows:

- Phase 1 - Logical model definition: a logical schema of the DB is modelled to shape the real-world entities to be considered by the DB. In the case of relational DBs, the relational model was proposed in 1970 by Edgar F. Codd [19] to define the logical schema of a DB. The relational model is still used nowadays, and is represented usually by using the Entity-Relationship (ER) formalism defined by Peter Chen

[18]. The DB's logical model represents through a graphical model (not concretely implemented) how the DB's knowledge is structured. The logical model definition is supervised by the expert of the data domain involved in the DB creation.

- Phase 2 - Physical model definition: the physical model of a DB, is the specification of how the logical model is concretely implemented. In the case of relational DBs, the physical model specifies how the relations (DB's tables) are defined, the data types, the ranges and the constraints of their attributes, and attribute values. Moreover, the physical model of relational DBs represents the status of the DB's knowledge representation after the normalization process. Normalization is the process of organizing data in a database. It includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency[1]. The DB's physical model is strictly related to the implementation of the DB itself, thus also dependent on the Data Base Management System (DBMS) adopted for the DB implementation. The physical model definition is executed by the DB's developer, due to her expertise about the DBMS to be adopted.

The two phases described above have different impacts on the representation of the DB's knowledge. First of all, it is important to notice that the logical and physical schema are independent of each other. In fact, while the logical schema is modelled to provide a structured representation of the DB's creator's Purpose, the physical one is an adaptation of the logical schema to fit the protection and consistency constraint for the DB implementation given by a specific DBMS. To better understand this main difference, it is possible to analyse how the logical model structure changes after the normalization process is applied to define the physical model. During such a process, new relations, and attributes, are added to the DB structure, thus actually changing the initial representation of the ETypes, as it was given by the domain expert. In other words, the relational DBs do not provide a concrete representation of the real-world entities, they only model logically such entities and then later change its logical representation to best fit the development constraints. The result is that the representation of the entity in relational DBs no longer fully incorporates the knowledge diversity provided by the domain expert, but is instead represented according to the DBMS constraints. The reuse of relational DB data is more complex because it is more difficult for new data users to properly understand the representation of entities by domain experts by only looking at the (physical) structure of the DB. Even assuming that the logical model of the DB is made available

[1] Microsoft - Normalization process

to the users interested in reusing the DB data, they will need to preprocess such data to properly represent it according to the logical schema.

Data value heterogeneity in relational Data Bases

The third layer of the relational DBs that we examined to investigate the lack of explicit diversity representation is the data value layer. This layer is focused on the heterogeneity of the data values that are associated with the entity types attributes modelled in the knowledge layer. Since the knowledge layer specifies the structure of each type of entity to be included in the DB, the data values are used instead to define each single entity, by assigning to the EType properties different values, for each entity considered. In relational DBs, such a representation is given in a tabular format, where each row of a table (relation) identifies part of the values characterizing an entity. This is only partially due to the fact that the logical schema representing the ETypes has been split over multiple tables (see section 3.1). However, to be compliant with the constraints introduced during the normalization process, the rows values of a table, in a relational DB, are not independent row by row. The row data values need to respect the consistency constraints given by each normal form, like the atomicity of each value (first normal form) and the dependency from the primary key only (second and third normal forms). This means that the values used in a row to represent an entity, need to be consistent with the values used to represent a different entity in another row in the same table. In other words, there are constraints which introduce a dependency between the DB's entities, but such constraints are imposed by the DBMS to properly access the DB data, not by the Purpose for which the data has been created. As a result, the choice of data values and the data types to be used for those values is also limited by those constraints. Static data types, like *String*, *Integer* and *DateTime* are less restricted by those constraints. However, more complex data types, which require a dedicated structure composed of its own set of properties (i.e., the date data type expressed as $[< dayValue >, < monthValue >, < yearValue >, < hoursValue >, < minutesValue >, < secondsValue >]$), must be properly structured to conform to the values of the entities from which they are used. The direct consequence is that more complex or dynamic (that needs to change in time) definitions of data types increase the complexity of the DB's structure, and limit the power of expression of the DB's data values. The scenario described above also hinders the concrete definition of the diversity defined by the domain expert (the user who has a clearer understanding of the information to be included in the DB) through the values of the data to be handled by the DB itself. In other words, the diversity of the domain expert, expressed by her Purpose, is not fully preserved in the relational DB due to the above-mentioned representational limitations, and consequently, it is also more difficult to be exploited as a feature by new users interested

in reusing such relational data.

3.2 The Entity Base data model

The lack of an explicit representation of the data diversity in relational DBs, as reported above, is a major obstacle to the reuse of relational data. In particular for applications based on AI technologies where the data diversity, which expresses the uniqueness of a context, becomes the key information during the AI learning processes. To cover the above described lack of data diversity representation, we propose a new data model, namely the Entity Base (EB) data model, based on a concrete and explicit representation of the data diversity at different levels (language, knowledge and data value). The EB data model is based on two key principles: (i) the concrete representation of the data diversity expressed by the data producer, through her Purpose; (ii) an explicit and Purpose-centric definition of "Entity", which in the relational DBs is only logically expressed. The first principle is fundamental to the production of diversity-aware data, where the richness of information about geographical, social and individual contexts can be exploited directly without re-interpretation. The second principle, instead is fundamental for representing such a diversity in a concrete and (re)usable form, namely an explicit representation of real-world entity.

The definition of the EB data model was born with the need to define a structured collection (or a "base") of entities (base of entities, or Entity Base) explicitly defined through their diversity at different layers. For this reason, before defining EB, it is necessary to define such a diversity-aware representation of the entity. A single real-world entity E, maintained into an EB, is defined by the EB data model as follows:

$$E :=< p(E), c(E), t(E), v(E) >$$

Where:

- $p(E)$ is the *Purpose* for which the entity E exist. Such a Purpose is expressed by a portion of a knowledge model (ontology, data schema, ER model) which aims at describing the context in which (or for which) the entity has been created.
- $c(E)$ is a *Concept Set* containing the concepts used to name the information carried by the entity E. Each concept in such a set is represented by three components: an identifier, a word, and a description (or gloss). Such concept representation comes from the one adopted by the UKC, as described in section 2.2.1.
- $t(E)$ is an *Entity Type (EType)* defining the structure (or data schema) of the entity E. It is defined as a set of data and object properties.

More detail about the EType definition is provided in section 2.2.2. The name of the EType and the names of its properties are the words of the concepts defined in $c(E)$.

- $v(E)$ is a *Value Set* containing the data values used to represent the entity E following the structure indicated by $t(E)$. Therefore, the values of this set instantiate the properties of $t(E)$.

The first component of the above entity definition is the entity's Purpose ($p(E)$). It models the interpretation of the real-world entity as it has been expressed by the person who created it. In other words, the entity's Purpose represents the point of view of the data producer over that specific entity. This component plays a crucial role in the reuse of the entity created following the EB data model. Thanks to the description of the entity's Purpose, a new user, different from the entity producer, will be able to understand the context for which the entity was created, and thus also the representation of its diversity, as concretely defined by the other entity components, will be more understandable.

The second component in the entity definition aims to concretely represent the language diversity layer of an entity. Such a component ($c(E)$) is a set of concepts containing all the concepts that are used to describe the EType as well as the properties which shape such EType. The concepts are formally defined by three elements. The first one is an identifier used to uniquely identify, in a language-agnostic way, a specific concept between one or more conceptual structures (like the UKC). In addition, the identifier is machine-readable and can be read immediately to automatically recognise one concept among others, without having to examine the linguistic components (word and gloss). The second element is a word, or a term, which is the language-dependent representation of the concept. This word is used to name the EType and the relative properties of the entity to be defined. The third element is a description of a concept, or a gloss, which allows human readers to disambiguate the meaning of the concept concerning other similar concepts. For example, for some Purposes and contexts, the concept described by the word "human" might be a synonym for the concept represented by the word "person", while in other contexts they might be different. The key idea is that the concepts in $c(E)$ concretely define the language diversity that is not explicitly expressed by the relations and attribute names in the relational DBs.

The third entity definition component is the EType which defines the structure (or the schema) of the entity. This component ($t(E)$) aims to concretely represent the knowledge diversity layer of an entity. The EType, as already mentioned in chapter 2, is defined by three elements. Also in this case the first one is an identifier which is used to uniquely identify the EType among

other ETypes into one or more knowledge bases (like other ontologies). As for the concepts defined in the first entity's component, the EType identifier is machine readable thus allowing for an automatic recognition of ETypes, without comparing the property sets of two or more ETypes. The second element of an EType is its name, a word that represents, in a meaningful and language-dependent way, a specific class of entities (like the EType named "car" defines, for a specific Purpose $p(E)$, the structure of all the entities representing a car). One of the concepts defined in $c(E)$ is used to give a name to each EType of $t(E)$. The third element of an EType is its property set, namely a set of properties which concretely define the structure (or schema) of an EType. It is important to note that the EType's property set is the most informative component of an EType, since an EType without properties is actually useless, so no entities can be represented by it. The property set of an EType is composed of data properties, describing the characteristics of the single ETypes, and object properties describing the relationships that the EType has with other ETypes in a specific context (the difference between data and object properties is more detailed in chapter 2). The ETypes in $t(E)$ concretely define the knowledge diversity of the entity, as it has been described by the data producer through the Purpose $p(E)$ (or the domain expert, having the highest level of expertise in the domain being considered). The key idea is that the logical representation of the entity in $p(E)$ is exactly the same as the relative EType in $t(E)$, thus eliminating the differences between the logical and physical representation of the entities, and preserving the knowledge diversity given by the data producer.

The fourth and last component in the above entity definition is a data value set ($v(E)$) which concretely defines the real-world entity. If the EType described above ($t(E)$) represents the schema of the entity E (or, more precisely, of a class of entities) through a set of properties, the values associated with those properties (instantiating the properties of the EType) are what concretely define each entity structured with that EType. The values set component aims to concretely represent the data values diversity layer of an entity. Unlike what happens in the relational DBs, the values representing each entity in the EB data model, are independent of each other, due to the fact that each entity is now defined independently from the others, not as a table row (relation) with values dependent on other rows. This allows the user responsible for the entity generation, to adopt the values that best represent the entity information, as well as to define more complex data types, to be eventually modelled at the knowledge layer into the EType definition. In the EB data model, an entity is a stand-alone object, with its own structure and values which are subject to constraints defined for the single entity only, by the Purpose $p(E)$. For this reason, the diversity, at the data values layer, is preserved as it is expressed by the data producer through her Purpose ($p(E)$).

As a consequence of the above entity definition, the EB data model defines an EB as a stratified collection of the elements defining the four entity diversity layers, for a set of entities interacting together into a context to satisfy a specific common Purpose. More formally, the EB data model defines an EB as follows:

$$EB :=< P, LT, ETG, EG >$$

Where:

- P is the *Purpose* expressed by the data producer as the leading expert about the domain and context in which the data has been created, as well as where the data is used. A more detailed definition of the EB Purpose is provided in the section below (section 3.3.1) describing how the EB Purpose is composed and used by the iTelos data representation process. More in detail, the Purpose P is the composition of the entity-specific Purposes $p(E)$ of all the entities to be considered for the construction of the EB.

- LT is the Purpose-specific *Language Teleontology*. The LT is an ontological model defining the hierarchical conceptual structure of the concepts defining the meaning of the information carried by the entities considered by the EB. The LT collects the entity-specific concept sets $c(E)$, for each EB's entity E, into a Purpose-specific language ontology following the UKC structure. More details about the composition and usage of the LT to build the EB, are provided in section 3.3.2.

- ETG is a Purpose-specific *Entity Type Graph* (ETG). It collects into the same ontological model the entity-specific $t(E)$ for all the entities considered by the EB. The ETG defines the schema of the EB by preserving the language, and knowledge layer diversity of the entities it models. More details about the composition and usage of the ETG to build the EB, are provided in section 3.3.3.

- EG is an *Entity Graph* (EG), namely a knowledge graph of entities. Such a KG is composed of the representation of all the EB's entities E described by their entity-specific values set $v(E)$. The EG is the instantiation of the ETG including all the entities considered by the EB. More details about the composition and usage of the EG to build the EB, are provided in section 3.3.4.

The EB data model defines the shape of the single entities and how they are composed together to satisfy a specific Purpose. The iTelos methodology adopts this data model to generate stratified diversity-aware data, by

reusing already existing data. To this end, the iTelos data representation process, described in the next section (section 3.3), defines how to represent the different components of an EB. Subsequently, the iTelos KGC process (described in chapter 4) reports all the activities required to concretely build an EB by following the representational stages indicated by the first methodology process.

3.3 The representation process

If the EB data model provides a new approach to concretely define a real-world entity and an EB, the iTelos data representation process, depicted in figure 3.2, is a representational process which defines how to concretely build each EB's components. It shows how an entity that in the input relational DB (left side of the figure) is only logically defined, is explicitly and formally defined into an output EB (right side of the figure). More in detail, The process inputs are two: (i) the logical schema of the relational DB, where the definition of the real-world entities is more detailed, even if such a definition is not concretely implemented in the relational DB; and (ii) the physical definition of the relational DB, represented by the tables (relations) containing the data. The two inputs are used to define the process output composed instead of four elements, namely the EB Purpose, the EB Language, the EB Knowledge and the EB Data values. The process follows the EB data model, defined in section 3.2, and it is composed of four main representational stages (represented by the vertical sections of the dashed table in figure 3.2) that aim at defining concretely the different diversity layers of each entity extracted from the input relational DB, to be then composed into the output EB. In other words, at each stage, the process shows how to define a single diversity layer for the entities implicitly defined in the input relational DB, with the aim of composing such layers representations into the EB, where the entities become explicitly defined. The four diversity layers, namely Purpose, Language, Knowledge and Data values, are concretely defined by the iTelos data representation process as layer-specific diversity-aware data which are used to build the EB. The definition of the language and knowledge layers of the EB, defined by the process described below, adopts the result of the work defined in [5]. The following section of this chapter details the generation of the process output, and how it is used to represent the four components of an EB. The following sections also describe the fundamental contribution of the UKC and the iTelos DI to the definition of the EB, shown at the top and bottom of figure 3.2, respectively. Their contribution to the iTelos representation process allows for the creation of more interoperable and reusable diversity-aware data.

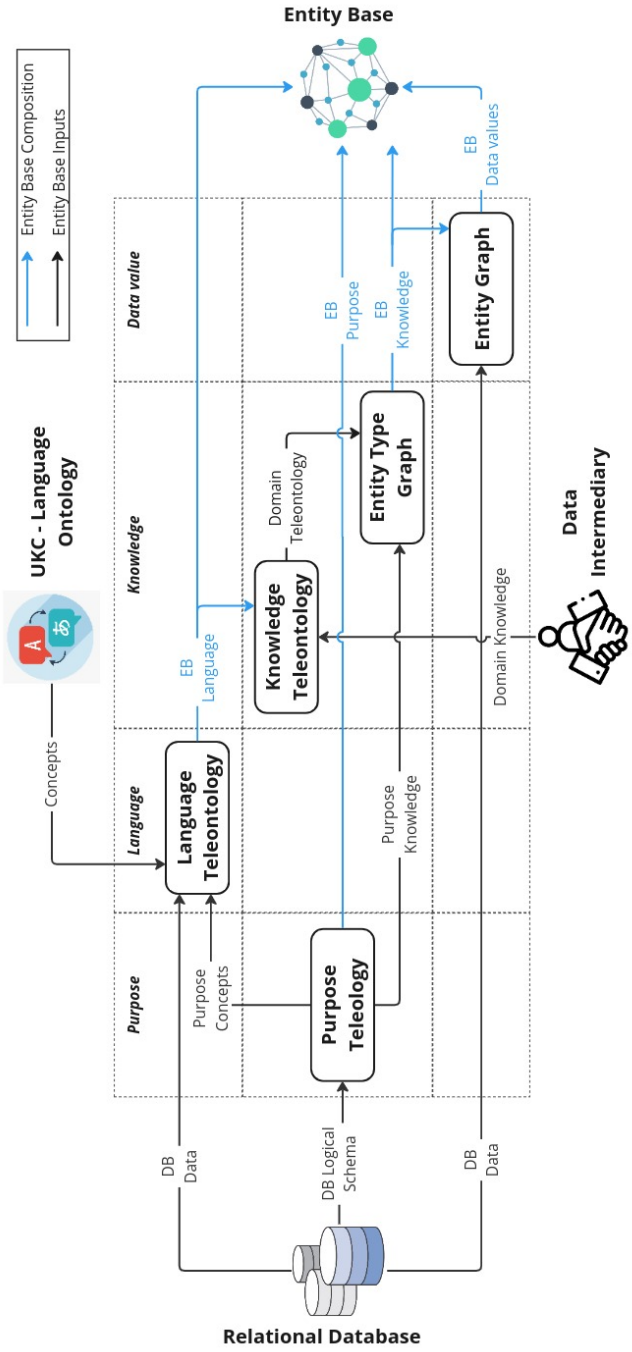


Figure 3.2: The iTelos data representation process.

3.3.1 EB Purpose representation

The Purpose is the first component of the EB, based on the EB data model. As already anticipated in chapter 1, the Purpose of data concretely expresses the diversity of the specific context in which the data has been created and used. Moreover, the Purpose expresses the diversity (the "point of view") of the person who created, or re-used, the data (the concept of Purpose can be applied to all the existing types of data, not only to the diversity-aware data generated and shared by the iTelos methodology). The Purpose intrinsically carries the information diversity which characterizes a person and a context, which can be considered as the person itself as an individual context, by highlighting the details that distinguish such a context from the others. For this reason, the Purpose is the first and most important component of the EB, being the definition of the diversity that will be used to create the diversity-aware data at the language, knowledge and data values layer, thus composing the other elements of the EB.

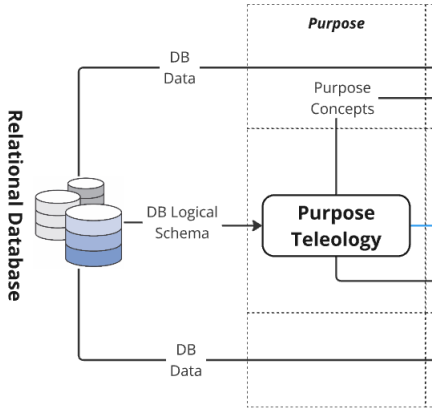


Figure 3.3: The Purpose representation phase of the iTelos data representation process.

As depicted in figure 3.3, extracted from the more complete figure 3.2, the EB Purpose is defined, in the first process representational phase, by taking in input the logical schema of the relational DB. Such an input schema, in fact, can be considered as an informal representation of the EB Purpose, shaped as an ER diagram (as it comes from the input DB design). However, the iTelos data representation process aims at providing a concrete and formal definition of each EB's components. For this reason, the input DB logical schema is formalized into a *Purpose Teleology*. To define the Purpose's Teleology, it is first necessary to define what teleology is. The word teleology builds on the Greek words *"telos"* (meaning: end, Purpose) and *"logia:* (meaning: a branch of learning). A teleology is defined as a flattened knowledge model of specific aspects of reality, more in detail, the

aspects considered by a specific Purpose. A teleology is always associated with a Purpose, and it aims at modelling the Purpose-specific EType and properties. Concretely speaking, teleology is a knowledge RDF file, defined in OWL language, modelling the ETypes, the data and object properties of a specific Purpose. By considering the above definition of teleology, the Purpose Teleology, depicted in figure 3.2, is a teleology that models all and only the ETypes and properties described informally by the ER model representing the input DB logical schema.

The EB Purpose representation defined by the iTelos data representation process, is described in this chapter to explain how a Purpose Teleology is represented, as part of the EB. However, the operational process for the construction of EB, included in the iTelos methodology and described in chapter 4, provides a more concrete definition of the activities to be executed for the formalization of the Purpose by considering also the cases in which an informal version of the Purpose is not given as DB logical schema.

3.3.2 EB language representation

The second representational phase of the iTelos data representation process aims at defining the EB language component. Such a component is defined above (see section 3.2) as a set of concepts, where each concept is used to define part of the information considered by the final EB. To be more precise, the concepts represented by the EB language component, define a Purpose-specific domain vocabulary, or what is described as a Controlled Natural Language (CNL) [90] in section 2.2.1. Such a domain vocabulary concretely defines the language diversity layer of the EB. However, since the iTelos methodology aims at producing formal and reusable resources, such Purpose-specific domain vocabulary has to be formalized into a *Language Teleontology* (LT). To define Language Teleontology, it is first necessary to define what teleontology is. The meaning of the word "teleontology" builds on the Greek words: *"telos"* (meaning: end, Purpose), *"onto"* (meaning: being) and *"logia"* (meaning: a branch of learning). Therefore, given what exists, teleontology refers to a formal, explicit hierarchical specification of objects, entities, functions and actions representing a shared conceptualization of a specific Purpose. A teleontology is written with a specific Purpose and there is no claim of generality beyond the Purpose for which it is modelled. More concretely, a teleontology is a knowledge model that uses four primary representational constructs: classes, object properties, data properties and data types, to define, as an RDF file using the OWL language, the ETypes for a specific Purpose, plus the ETypes defining more general aspects related to the Purpose (OWL super-classes). Unlike teleology, the teleontology is more hierarchically defined, by extending the description of the Purpose-specific ETypes with the description of more general ETypes

which categorize (classify at a more general level) the Purpose-specific ones. The Language Teleontology, depicted in figure 3.4, is a teleontology where the ETypes are represented by the UKC concepts related to a specific Purpose. In other words, the LT is the part of the UKC structure that contains the conceptual hierarchy that makes up all the concepts needed to define the Purpose's information. It is important to notice how, the LT is actually a representation of the EB language diversity where all the required concepts are formalized into a hierarchical structure which therefore provides a precise description of each concept's meaning, as well as how it is located into the Purpose-specific hierarchical structure.

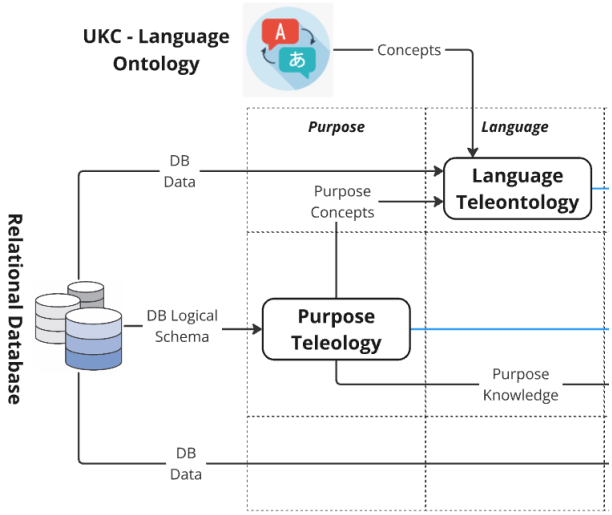


Figure 3.4: The Language representation phase of the iTelos data representation process.

As depicted in figure 3.4, extracted from the more complete figure 3.2, the iTelos data representation process defines an LT by collecting concepts from two inputs. The first input is the Purpose Teleology, modelled in the previous representational phase. The concepts extracted from the Purpose Teleology, are represented by the names of the ETypes and their properties. The second input, instead, is the data of relational DB. The concepts extracted from the second input are terms used to define specific values in the DB's tables (relations). For example, to define the gender of a person, some possible values are "Male", "Female" or "Other". Such values are all terms which represent specific concepts that need to be disambiguated into a diversity-aware language resource. The second representational phase models the LT executing two internal steps, described here below:

- *UKC alignment*: the first step aims at collecting all the concepts required to define the Purpose-specific domain vocabulary, concretely

represented by the LT, and to align such concepts with the UKC's LO. For this reason, in figure 3.4 the UKC's LO concepts are considered as process input. The UKC alignment is described by the first two steps of the process described in [10], summarized here below;

- *Step 1 - Terms collection.* The first step aims to identify and collect, from the Purpose Teleology and the DB's data, the terms representing all the concepts to be used for the modelling of the LT. The terms are collected within a term set (TS) from which they are extracted one by one in order to be processed.
- *Step 2 - Concept integration.* In the second step, for each term t extracted from TS, it is required to check if a concept c exists, in the UKC-CC (see chapter 2 for more detail regarding the UKC Concept Core), suitable for t. In this operation, the gloss associated with each concept is exploited to identify the right one. If such a concept is available, the term t will be added in the UKC-LC (see section 2.2.1 for more detail regarding the UKC Language Core), namely a representative word t for the concept c. If, instead, c is not available, a new concept c (with its gloss) is added in the UKC-CC, adding subsequently t(c) in UKC-LC, as in the previous case.

At the end of the concept integration step, all the concepts collected in the first step (terms collection) are properly aligned with the UKC structure (the UKC's LO) both in the UKC-CC and in the UKC-LC.

- *Language Teleontology definition*: the second step, for the representation of the LT, exploits the UKC's LO updated previously, to extract the hierarchical structure including all the concepts interested by the Purpose which, as mentioned in section 3.3.1, is leading the entire process. It is important to notice that, during the execution of this step, all the interested concepts have been already formalized and structured into the UKC's LO, thus becoming ETypes of the UKC's LO. Therefore, to concretely define the LT, the process considers one by one, each concept taken in input in the current representational phase (Purpose Teleology concepts and DB data concepts), and extracts from the UKC's LO, the portion of the hierarchical structure including all the parent and children ETypes of the specific EType representing the concept processed. Once all the concepts have been processed, the portion of the UKC's LO extracted is a hierarchical structure defining a formal version of the domain vocabulary specific to the Purpose considered.

The two steps defined by the iTelos data representation process, to represent the LT as an EB language component, are described in this chapter to ex-

plain how an LT is represented, as part of the EB. However, the operational process for the construction of EB, included in the iTelos methodology and described in chapter 4, provides a more concrete definition of the activities to be executed for the selection and formalization of the concepts used to generate the LT, as well as tools and instruments supporting the users interested in producing such kind of language resources.

3.3.3 EB knowledge representation

The third representational phase of the iTelos data representation process aims at defining the EB knowledge component. Such a component is defined above (see section 3.2) as a knowledge model called *Entity Type Graph* (ETG). As the name suggests, the ETG is a graph-based knowledge model describing the ETypes representing the classes of entities considered by the final EB, as well as the relationships linking such ETypes. To be more concrete, the ETG is again a Teleology (like the Purpose Teleology described in section 3.3.1), namely a flattened knowledge model related to a specific Purpose. Although the ETG and the Purpose Teleology refer to the same Purpose (within a single execution of the process) they differ in how they model the Purpose ETypes. In fact, while the Purpose Teleology is modelled to fit as much as possible the input Purpose, the ETG aims to be both Purpose-specific and highly interoperable, and as a consequence more reusable too. The key idea to satisfy the interoperability requirement is to reuse, whenever possible, ETypes and properties from already existing reference domain knowledge to model those Purpose-specific ETypes and properties that describe the same information. In this way, the resulting ETG will be focused on the Purpose to be supported, but representing a portion of its ETypes and properties in a standard way already adopted by largely used (in different contexts and for different Purposes) domain reference knowledge models.

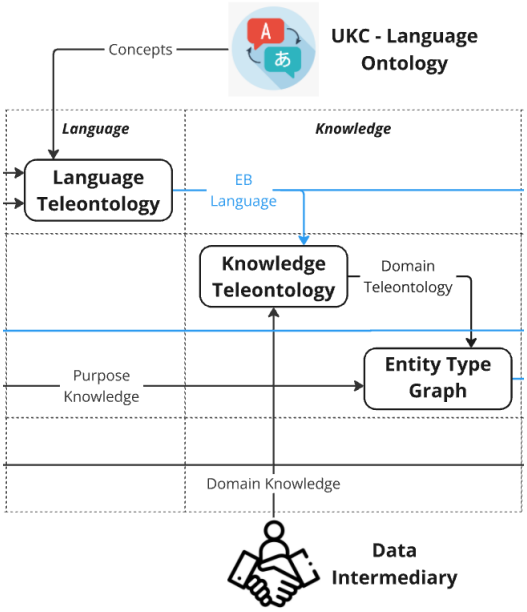


Figure 3.5: The Knowledge representation phase of the iTelos data representation process.

As depicted in figure 3.5, extracted by the more complete figure 3.2, such a feature of the iTelos data representation process, which leads to the definition of the EB's ETG component, is implemented in two steps, described here below.

- *Step 1 - Knowledge Teleontology definition.* The objective of the first steps is to identify and collect the reference domain knowledge which can be reused to model the information considered by the Purpose. To this end, two of the three representational phase inputs are used, namely the LT previously produced, and the domain knowledge provided by the iTelos DI. The LT is used to indicate which is the knowledge, considered by the Purpose, for which the reuse of already existing knowledge models needs to be maximised. This is possible thanks to the LT's ETypes and properties, which are clearly described by disambiguated concepts. On the other side, instead, the DI (described in chapter 5) acts as a collector of existing reference domain knowledge, for different domains, contexts and Purposes. Therefore, the LT drives the selection of already existing knowledge models, among those collected by the DI, by matching the ETypes meaning through their representations. Such a selection is concretely implemented by the ETypes Recognition (ER) algorithms presented by Giuchiglia and Fumagalli [41]. The matching between ETypes is supported by the fact that both the LT and the knowledge models provided by the DI are represented

as graph-based ontological models (RDF-OWL file). It is important to mention that, the reference knowledge models provided by the iTelos DI are knowledge resources which have been already handled by the iTelos methodology. This means that the ETypes within such knowledge models are also defined by using concepts already aligned with the UKC's LO. This is an important feature, since the selection of the ETypes to be reused, according to the LT's EType description, could be performed automatically by checking whether the ID of the concept used to define the EType to be reused is the same as the concept defining the EType specified by the LT. The DI intermediary provides also well-known standard domain knowledge models, like Schema.org[2] and FOAF[3], which have been processed by the iTelos methodology, to be aligned with the UKC's LO. Therefore, such models can be actually reused by the automatic EType selection, through the algorithm mentioned above.

Since the ETypes considered by the LT can be reused from different domain knowledge models, they need to be collected and composed into a single knowledge resource. For this reason, the iTelos data representation process defines, in this phase, the *Knowledge Teleontology* (KT). A KT is a teleontology, as described in section 3.3.2, composing reference domain knowledge collected from one or more domains of interest, as well as different contexts and Purposes within a single domain. Two observations have to be made over the KT. First, a KT is always compliant with the hierarchy of the UKC by construction. This first observation ensures that the meaning of the ETypes and their properties, as they are specified in the LT coming from the EB language definition phase, remains aligned with the initial Purpose. The second observation is that the KT aims to reuse as much as possible from the knowledge provided by the iTelos DI, thus representing the ETypes and the relative properties following such reference domain knowledge. If the first observation guarantees the modelling of a Purpose-specific KT, the second one aims to increase its reusability and interoperability, thanks to the adoption of standard domain knowledge. It is important to notice the main difference between the LT and KT. While the LT is modelled to express the language diversity of the Purpose, thus representing ETypes and properties associated with specific UKC concepts, the KT aims to enrich the meaning of such ETypes at the knowledge level, by representing their structure through data and object properties. In other words, even if LT and KT are both teleontologies, they are focused on two different information layers, one on the language layer and the other on the knowledge layer, respectively.

[2] https://schema.org/
[3] http://xmlns.com/foaf/spec/

- *Step 2 - ETG definition.* The second step aims at producing the knowledge model that best fit the Purpose leading the entire process, namely the ETG. To argue the need for this process step, it is important to note that the KT produced in the previous step is modelled by reusing knowledge from "external" knowledge models, where "external" means not defined for the specific Purpose guiding the current process execution. In other words, this means that the diversity at the knowledge layer indicated by the Purpose is not fully represented in the KT. This happens because the objective during the construction of the KT is to enhance interoperability. However, the KT, being a teleontology, includes the representation of additional general knowledge (superclasses) not strictly required to represent the information indicated by the Purpose. This introduces complexity, which must be paid with more effort when the KT is used to support applications and services (i.e. more complex queries). For these reasons, the iTelos data representation process defines in this phase the ETG as a teleology, obtained by composing the Purpose Teleology, received in input in the current phase (indicated by the arrow labelled with "Purpose Knowledge" in figure 3.5), and the KT previously defined. Such a composition aims to replace the ETypes and properties representation of the Purpose Teleology with a more interoperable version given by the KT, if and only if such a replacement does not change the interpretation of the ETypes given by the Purpose Teleology. In other words, in this step, the goal is to increase the interoperability and reusability of Purpose Teleology. Moreover, since this step is driven by the Purpose Teleology (the subject ETypes are all and only the ETypes of the Purpose Teleology), the resulting knowledge model, the ETG, will be composed of all and only the ETypes considered by the Purpose-specific teleology. This makes it possible to prune away unnecessary knowledge objects present in the KT and to reduce the complexity of the ETG produced with respect to the KT. It is important to note how the generation of the ETG is aided by the fact that the KT has previously been aligned with the LT (see previous step), which in turn has previously been aligned with the Purpose Teleology. Therefore, the meaning defined by the UKC concepts, and the conceptual structure maintained by the ETypes defined using such concepts, allow for a reduction in ambiguity during the composition of the ETypes of the ETG.

To summarise the execution of the two representational steps described above, the key idea is to model an interoperable diversity-aware knowledge model, namely the ETG. Where the ETG has to be firstly Purpose-specific, being the Purpose the reference for the diversity to be expressed at different levels, and secondly interoperable and reusable thanks to the reuse of standard domain knowledge.

3.3.4 EB data value representation

The fourth and last representational phase of the iTelos data representation process aims at defining the EB data values component. Such a component is defined above (see section 3.2) as a knowledge graph of entities, called *Entity Graph* (EG). Such a KG composes together all the entities extracted from the input relational DB and concretely represented in the final EB. The EG is a KG (see chapter 2 for a more precise definition of KG) whose schema follows the ETG, modelled in the previous phase, where the nodes are the instantiation of the ETG's ETypes and properties. The data values extracted from the relational DB are taken in input to be associated with the properties of each EType, thus representing concretely each entity expressed by the uniqueness of those data values. The EG concretely defines the entities by following the structure of the ETG that has been modelled starting from the logical schema of the relational DB. This means that, unlike a relational DB where the logical and physical schema are not aligned, in an EB the alignment of the EG and the ETG guarantees that the entities are modelled as they were logically interpreted at the beginning of the EB creation process. It is important to notice how the EG expresses the diversity of the entities at the data values layer, by generating a KG's node for each entity considered.

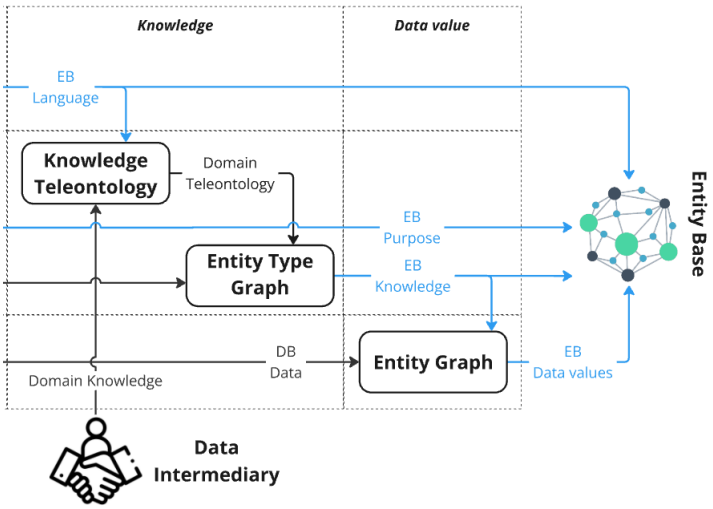


Figure 3.6: The Data values representation phase of the iTelos data representation process.

Unlike the relational DB, the representation of entities in the EG is independent entity by entity. Moreover, the independent representation of each entity as a KG's node, allows for the definition of more complex data types (i.e., DateTime is defined as the composition of "Year", "Month", "Day", "Hours" and "Minutes") without violating constraint given by the physical

structure, as it is the case in relational DB to align the entity representations into the same relation (table) following the normal forms. Figure 3.6 shows how the EG is represented by merging the ETG, modelled in the previous phase, and the DB data coming directly from the input relational DB, into a single object, the EG. This operation is a mapping of the DB data values over the ETG's ETypes and properties [50].

The four phases of the iTelos data representation process, produce the four components of the EB (blue arrows in figure 3.2). As a consequence, the entities that are implicitly defined in the input relational DB, become explicitly represented in the EB, as defined by the EB data model (see section 3.2). However, the iTelos data representation process, as the name suggests, aims to describe how the EB is represented, by defining the shape of its components. Instead, the following chapter (chapter 4) presents an operational KG Completion (KGC) process, which shows how the iTelos methodology concretely supports a user in the production of an EB by following the four phases of the data representation process according to the EB data model.

4

The iTelos KGC Process

Contents

This chapter aims to describe the iTelos KGC process. The process represents the core of the iTelos methodology being the main instrument used by users to create the data they require for the development of digital applications and services. The principal goal of the iTelos KGC process is to reduce the cost of creating KGs by reusing as much as possible already existing data. In this chapter, we describe how the iTelos KGC process adopts the EB data model defined by the iTelos data representation process, and how such data model is applied to provide a Purpose-driven KGC process based on data diversity. Moreover, this chapter describes how the iTelos KGC process aims at enabling a cooperative circular data economy where the data users can benefit from a reduced cost of data reuse thanks to the data produced, and shared, by other users. The rest of the chapter is organized as follows. Section 4.1 provides more details about iTelos basic principles and objectives. From section 4.2 to section 4.5, the chapter reports the description of the four iTelos KGC process phases. Section 4.6 describes the evaluation activities included at the end of each process phase. Section 4.7 concludes the chapters with a description of the metadata that the iTelos KGC process uses to share the diversity-aware data it produces after each execution.

4.1 Process overview

In this section, we provide a general overview of the process, by describing its high-level structure, the roles of the users involved in the process, and the principles at the base of such a process design.

4.1.1 Process structure

The iTelos KGC process is a phase based structured process, which takes at the beginning two inputs:

- the informal definition of the user Purpose. Given by the process user, this first input refers to a statement expressed by the user in a natural language. It represents the objective the user aims to satisfy through the creation of a KG. However, the user (who can be a person without technical expertise and without previous experiences in data and knowledge engineering) expresses such an initial Purpose statement with her own perception of the world (with her own vocabulary and mental structures) therefore it can result in an ambiguous definition. For this reason, it is defined as an "informal" user Purpose.
- A list of data sources providing information to be considered for the satisfaction of the user Purpose. The second input can be a set of links or a list of data source names, which will be considered for two main objectives. The first one is to better understand the informal user Purpose, thus supporting its following formalization. The second is to concretely collect the data to be handled by the whole process and to be integrated into the final KG.

The process output, instead, is a KG, or more in detail an EB as defined by the EB data model, which satisfies the requirements which have been extracted from a formalized version of the initial informal user's Purpose. Moreover, the final KG is built by integrating the data collected from the data sources indicated as input.

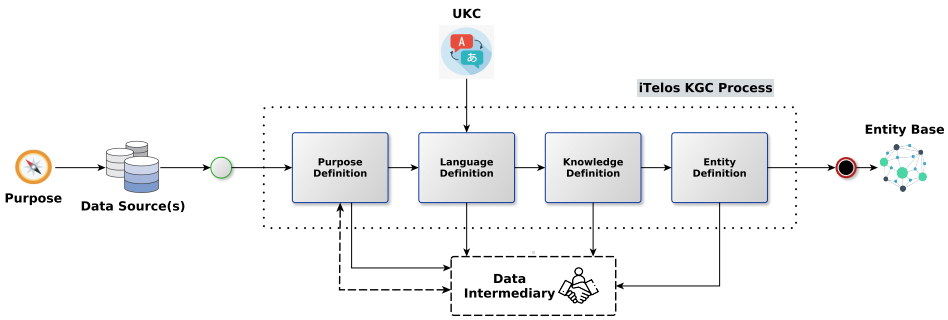


Figure 4.1: iTelos KGC process - top level structure.

Figure 4.1 depicts the iTelos KGC process structure, together with the inputs and output above described. In particular, the figure depicts the four phases of the process, namely Purpose Definition (PD), Language Definition (LD), Knowledge Definition (KD) and Entity Definition (ED). Each phase has its own internal structure, requires phase-based inputs, and produces specific outputs which have to be moved forward to be considered as inputs of the subsequent phase. In the figure, are also depicted the UKC (on top) which provides the Language Ontology required by the iTelos data representation process to build the Language Teleontology (see the previous chapter), and the Data Intermediary (on bottom) collecting the intermediate output of each phase to be then reused as diversity-aware data. In the next chapter, the DI will be better detailed. The different phases of the iTelos KGC process are described in detail in the following sections, nevertheless, we report here below a top-level description of the main objective of each phase.

- *Purpose Definition*: this phase has a twofold objective. The first one is to define formally the informal Purpose given as input by the user. By formally we mean a clear definition of the requirements to be satisfied by the final KG. The second objective of this phase is the collection of the data required to build the final KG. Such data is collected from the data sources indicated as input, as well as from the DI which is actually a source of diversity-aware data (having a reduced cost of integration and reuse). In this first phase, the iTelos KGC process addresses the issue of data heterogeneity at the data source layer.

- *Language Definition*: this phase receives as input what has been produced by the previous one, namely the formalized version of the user Purpose and the data collected. Additionally, the iTelos KGC process, in this phase, has access to the UKC. The objective of this phase is to build the Language Teleonotlogy (LT) as indicated by the iTelos data representation process. In the Language Definition phase, the iTelos KGC process addresses the issue of data heterogeneity at the language layer.

- *Knowledge Definition*: the third phase of the iTelos KGC process takes as input the formal user Purpose, the LT produced by the previous phase, as well as the set of reference domain ontologies and data schema collected with the objective of building the final KG. The objective of this phase is to build the Knowledge Teleontology (KT) first, and then the Purpose-specific Teleology representing the Entity Type Graph (ETG) of the final KG. In the Knowledge Definition phase, the iTelos KGC process addresses the issue of data heterogeneity at the knowledge layer.

- *Entity Definition*: the last phase of the iTelos KGC process takes as input the formal user Purpose, required, as in all previous phases, to

drive the process and evaluate the satisfaction of the user requirements. Moreover, this phase takes as input the data collected by the first phase of the process, and the teleology defined in the Knowledge Definition phase. The objective of this phase is to build the Entity Graph (EG) by merging the input teleology with the data values carried by the collected datasets representing the real-world entities to be included in the final KG. In the Entity Definition phase, the iTelos KGC process addresses the issue of data heterogeneity at the data value layer.

It is important to notice how the output of the four phases of the iTelos KGC process compose an EB as it is defined by the EB data model through the iTelos data representation process. Therefore, the KGC process provided by the iTelos methodology produces not only the EG satisfying the user's Purpose, but also the LT representing its language diversity, the Teleology (and KT) representing its knowledge diversity, as well as the formal version of the Purpose explaining why and how the previous outputs have been built like that.

4.1.2 Process users roles

It is important to detail better who are the users of the iTelos KGC process, and which roles they can cover along the whole process execution. In general, as already mentioned in the previous chapters, the user of such a process is a person who aims to generate KGs to support data-based services or applications. However, this definition is quite broad because it can include users having different levels of expertise and technical skills in data management and knowledge modelling, which are all competencies required to build a KG. Based on such prerequisites for building KGs, we can distinguish three main roles that can be played by the iTelos KGC process users:

- *Domain Expert*: this role is played by the person (one or more) who has the highest level of understanding about the domain of interest for which the initial Purpose has been defined. Usually, the user covering this role has no particular technical skills, however, its contribution is fundamental for the definition (and formalization) of the Purpose, that actually drives the whole process. The domain expert is also responsible for the validation of the final KG, with the objective of testing if it is suitable to satisfy the Purpose expressed. For these reasons, this role is mainly involved in the first and last phases of the iTleos KGC process.

- *Knowledge Engineer*: this role requires expertise in knowledge representation and knowledge modelling. The person (one or more) who plays this role is responsible for the modelling of the LT, the KT and the ETG, as well as partially involved in the formalisation of the initial

Purpose. A high level of expertise in ontologies and data schema, as well as in the tools for their definition, is required, being those the main data structures to be used by the person playing this role. This role is mainly involved in the first, the second and third phases of the iTleos KGC process.

- *Data Scientist*: this role requires expertise in data management, and ETL techniques (Extract, Transform, Load)[1], as well as data science techniques. The person (one or more) who plays this role is responsible for collecting the data required to build the final KG. If the knowledge engineer is focused on ontologies and data schema modelling (knowledge datasets), the data scientist is responsible for the quality of the datasets carrying data values, or data values datasets (i.e., row datasets like CSV, TSV, and others. But also object-oriented datasets like JSON and others), that will be used to properly define the real-world entities. Moreover, the data scientist ensures a high-quality level for the datasets she handles, by executing data cleaning and data formatting (standardization) activities. For this reason, this role is mainly involved in the first phase of the iTelos KGC process, as well as in the last one where this person is responsible for the merging of the ETG and the data values datasets to produce the EG.

The three roles described above can be played by different people, all participating in the same project by executing the iTelos KGC process, or even by a single person if she has all the competencies to cover the different roles. Even if each role is more active in some phases of the process, they are always involved in the four phases to help the other users better understand some of the work they have done in the previous phases. If the iTelos methodology is employed to implement a huge project, a fourth role could be needed, namely the *Project Manager*. The objective of this role is to coordinate the whole project by supervising the execution of all the iTelos KGC process phases and assessing the intermediate, and the final, outputs. Moreover, the project manager is responsible for the efficient communication and cooperation between the users covering the other roles. Figure 4.2 depicts a possible work breakdown between the iTelos KGC process roles, into a ten-weeks size KG development project.

[1] https://it.wikipedia.org/wiki/Extract, Transform, Load.

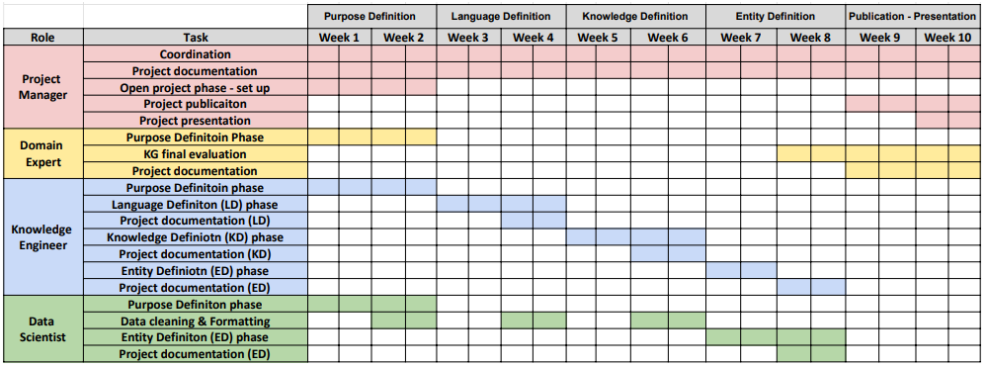


Figure 4.2: iTelos KGC process - Roles effort

4.1.3 Base principles

The iTelos KGC process, but in general the overall iTelos methodology is based on two base principles that actually link this process with the iTelos data representation process described in the previous chapter, and the iTelos data distribution process, which is detailed in chapter 5.

The first base principle is to provide a Purpose-specific KGC process based on data diversity. This first principle focuses on the methodology user and the context in which she lives. It aims at providing the user with a tool for the satisfaction of her Purpose (her goals). The key idea is to help the user to realize herself, by achieving her own objectives. The need for the first principle came from the fact that the reuse of existing data is the key feature adopted by the iTelos methodology to reduce the cost of building KGs. Nevertheless, the data reuse directly implies a re-interpretation of the data itself, from the data producer's "point of view" (or Purpose), to the "point of view" of the user who has to reuse it (the data consumer). As already discussed above, the representation of the data to be reused is affected by heterogeneity at different layers. However, such heterogeneity is the key to concretely defining, through the data, the local diversity of geographical, and social contexts and the interests of the individuals living in such a context, like the user of the iTelos KGC process. Our Purpose defines our own local (and personal) diversity, or, in other words, our diversity is expressed by the (different) Purpose(s) we have during our life. For this reason, the first base principle of the iTelos KGC process is to generate diversity-aware KGs, by maintaining the user Purpose at the centre of the overall process. The iTelos KGC process starts with the formal definition of the user Purpose with the objective of identifying those implicit or explicit aspects carried by the informal Purpose as it is initially expressed by the user. Such a formalized Purpose is the object that will be used to drive the iTelos data representation process, within its different layers, to generate an EB suitable to satisfy the

user's Purpose.

If the first principle is focused on the user's Purpose realization, the second one aims at sharing the personal results with a community. The second base principle of the iTelos methodology, implemented into the KGC process too, is the enabling of a circular data economy into a community of data users. The key idea under the second base principle is to facilitate the work of other users, in achieving their own Purposes, by sharing the personal results built in accordance with the first principles. Such results are shared in the form of diversity-aware data representing at different layers (language, knowledge and data value) a local context as well as local individuals. More concretely, the iTelos KGC process is actually creating a new more informative and exploitable type of data, that allows for a more smart integration with a reduced cost of pre-processing (quality stratified data enriched with a formal definition of the Purpose for which it has been created). However, the cost reduction introduced by the iTelos KGC process will scale in time only if such a new type of data is available in the data spaces (data sources considered by the iTelos KGC process), therefore available to be reused too. While the implementation of the first principle is achieved thanks to the iTelos data representation process following the EB data model, the second principle is implemented by applying firstly the iTelos KGC process for the creation of (or transformation of existing data into) diversity-aware data, and secondly for the exploitation (or consumption) of such a data for application-specific Purposes. In other words, the iTelos KGC process is designed to be used both by the data producer, to create the data to be distributed, and by the data consumer to integrate the available diversity-aware data to satisfy a specific Purpose. The joint execution of the iTelos KGC process by both data producers and data consumers enables a circular use of data among the users of the iTelos methodology.

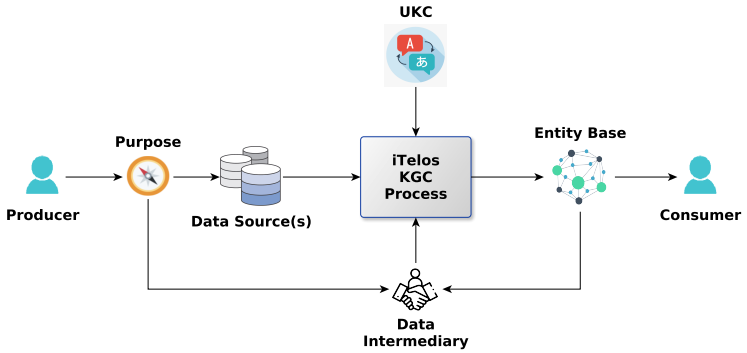


Figure 4.3: The iTelos KGC process for data producer.

Figure 4.3 depicts how the iTelos KGC process is considered when it has to support the data producer. From left to right, in the figure, it is possible to notice how the Purpose plays a crucial role also for the data producer. As already discussed above, the Purpose defines the contextual, or individual, local diversity to be represented in the data. The data producer aims at creating (or transforming existing data into) diversity-aware data able to represent such diversity. It is important to note that the data producer may not have in mind a specific application for the data to be supported. In this case, the Purpose describes as much as possible a geographical, social or individual context, rather than specifying application requirements. A list of data sources is provided as input together with the Purpose, to transform already existing data into diversity-aware data. Then the data producer executes the iTelos KGC process, which, with the support of the DI and the UKC produces an EB where the language, knowledge and data value diversity is represented through KGs, as defined by the EB data model. The DI, from the data producer side, supports the iTelos KGC process execution by providing the reference domain standard knowledge required for the interoperability of the KGs to be generated. Such process support is driven by the Purpose that indicates which are the information domains considered by the data. This interaction is described by two arrows in figure 4.3, one going from the Purpose to the DI and the second from the DI to the iTelos KGC process. Moreover, in this case, the DI collects the outputs of the iTelos KGC process, in order to be available for future data production, as well as for the data consumers. This interaction is depicted in the figure by the arrow going from the EB to the DI. At the end of the data producer use case described in figure 4.3 (right side) there is the data consumer who can benefit from the diversity-aware data (EB) produced.

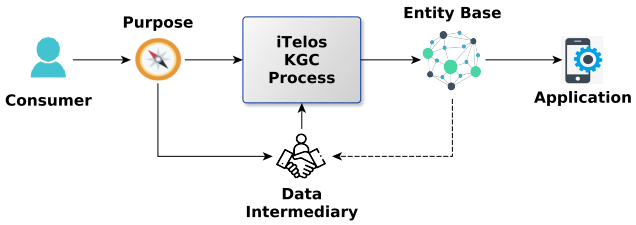


Figure 4.4: The iTelos KGC process for data consumer.

Figure 4.4 depicts how the iTelos KGC process is considered when it has to support the data consumer. Like in the previous case, starting from the left side of the figure, the Purpose expressed by the data consumer is the main input of the iTelos KGC process. However, the Purpose on the consumer side is focused on a specific application or service to be developed. Concerning the data producer's Purpose, the Purpose on the consumer side provides few

details about the broader geographical and social context, but more about the application-specific requirements to be satisfied. Looking at figure 4.4 immediately appears the missing input data sources. This is because the data consumer, as defined by the iTelos methodology, will be more facilitated by the reuse of the diversity-aware data distributed by the DI, thus actually considering the DI data space as a unique source of data (and second main input for the iTelos KGC process). As in the data producer case, the data consumer Purpose is used by the DI to understand which diversity-aware data has to be provided as input to the iTelos KGC process. The difference from the previous case is that the DI now provides the process with all the data (language data, knowledge data and data values) needed to build the KGs, not just the reference standard knowledge supporting interoperability on the producer side. It appears that the data consumer execution of the iTelos KGC process is mainly a matter of data integration, by considering as input only KGs already created to be highly integrable. As a result, the integration of diversity-aware data, that takes place at the data consumer's end, generating an application-specific EB, is like putting together a jigsaw puzzle. At the end of the data consumer use case, described in figure 4.4 (right side), is present the application for which the EB has been developed. However, the integration of KGs, on the consumer side, itself produces new diversity-aware data which can be distributed by DI, under the permission of the data consumer. The dashed arrow in figure 4.4 going from the EB to the DI, highlights such a possibility.

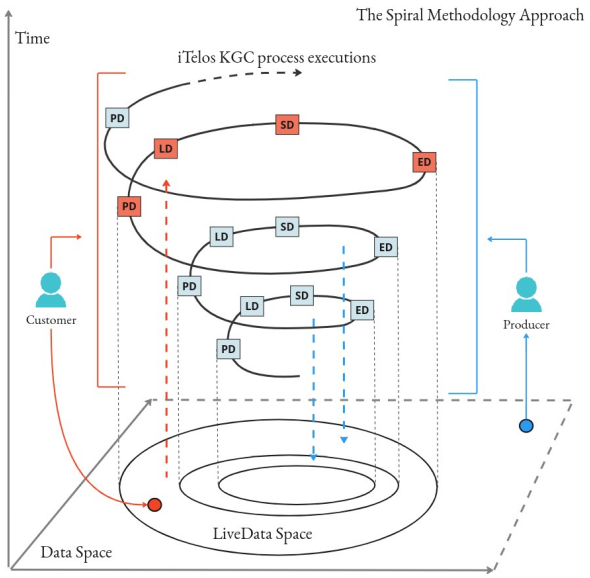


Figure 4.5: The iTelos methodology spiral approach.

The first and the second base principles underlying the iTelos methodology, and implemented through its processes, can be summarized into the cooperative execution of the iTelos KGC process, between data producer and data consumer, finalized to the creation of an increasingly wide diversity-aware data space. Such a data space is called LiveData (detailed in chapter 5) and it is handled by the iTelos DI through the iTelos data distribution process. Nevertheless, to better understand how the iTelos KGC process is the core instrument to support the two base principles, we report in this chapter how the data consumer and the data producer are linked together and can benefit from the LiveData space. Figure 4.5 depicts such interaction between the three main actors, namely the data producer (right side), the data consumer (left side) and the DI represented in the figure by the LiveData space (bottom side). In the centre of the figure, the iTelos KGC process takes place, represented as a spiral that includes multiple executions of the process. Each spiral loop represents an execution of the iTelos KGC process, thus including its four phases (Purpose Definition (PD), Language Definition (LD), Knowledge Definition (KD), and Entity Definition (ED)). The blue squares in the spiral loops represent the iTelos KGC process phases executed by the data producer, while the red ones are the phases executed by the data consumer. The data producer collects the existing data from the "external" *Data Space* (different sources with respect to the LiveData space), and uses the iTelos KGC process to transform such data into diversity-aware data that is collected by the DI into the LiveData space. This data producer process is represented in the figure by the blue arrows, where the solid ones go from the data space to the Producer and then as input to the KGC process, while the dashed one represents how the KGC process internally transfers the data to populate the LiveData space. It is important to notice how the LiveData space grows after each execution of the iTelos KGC process, collecting more and more diversity-aware data. On the data consumer side, instead, the iTelos KGC process, driven by the data consumer Purpose, collects the data from the DI's LiveData space to build the application-specific KG (or the entire EB). The red arrows in figure 4.5 show the data consumer execution. In particular, the solid ones represent, firstly, how the data consumer drives the selection of the data into the LiveData space (arrow from the consumer to LiveData), and secondly, the use of the iTelos KGC process by the data consumer (arrow from the consumer to the spiral). The red dashed arrows from the LiveData space to the red spiral loop, represent the iTelos KGC process data retrieval at the data consumer side. The *Spiral Approach* adopted by the iTelos methodology, enables a collaborative production and exploitation of diversity-aware data, through the growth over time of a dedicated data space (LiveData) that collects and distributes quality and interoperable data that can be integrated with a reduced cost of pre-processing.

In the next sections of this chapter, we detail the iTelos KGC process phases,

by describing their internal activities, as well as their intermediates inputs and outputs. For each phase, we will discuss how the process supports both the data producer and the data consumer's needs. The last two sections of this chapter will be dedicated to the definition of the evaluation activities, and the metadata definition activities, present in the structure of each phase of the iTelos KGC process.

4.2 Phase 1 - Purpose Definition

This section is dedicated to the description of the first phase of the iTelos KGC process, namely the Purpose Definition (PD) phase. As already mentioned above this phase takes as input the informal user's Purpose and a list of data sources, and aims at producing a formal version of the initial Purpose, as well as at collecting the data required to satisfy such a Purpose, from the indicated data sources. Figure 4.6 shows a more detailed version of the PD phase, by specifying its internal activities and process flow. As can be seen in the left-hand side of the figure, the inputs to the phase are represented by the objects numbered 0 and 1, namely the informal Purpose and the data source list, respectively. In the right-hand side of the figure, instead, the two outputs, represented by objects 2 and 3, are the formal Purpose and a data package containing three different types of data, namely language data (3L), schema (or knowledge) data (3S) and data values (3D). The key intuition in the PD phase is to formalize a Purpose that is able to guide the entire process along all its phases. In particular, in this phase, the Purpose represents a unique "point of view" to be used as an interpretative key when collecting data from multiple sources, each of which has its own interpretation and representation of the data it provides. In other words, the Purpose acts as a unity across the source layer of data heterogeneity, by considering the local diversity in each data source. The following subsections describe in detail, the internal activities, their process workflows, as well as the tools and standards provided by the iTelos methodology to concretely implement the first iTelos KGC process phase.

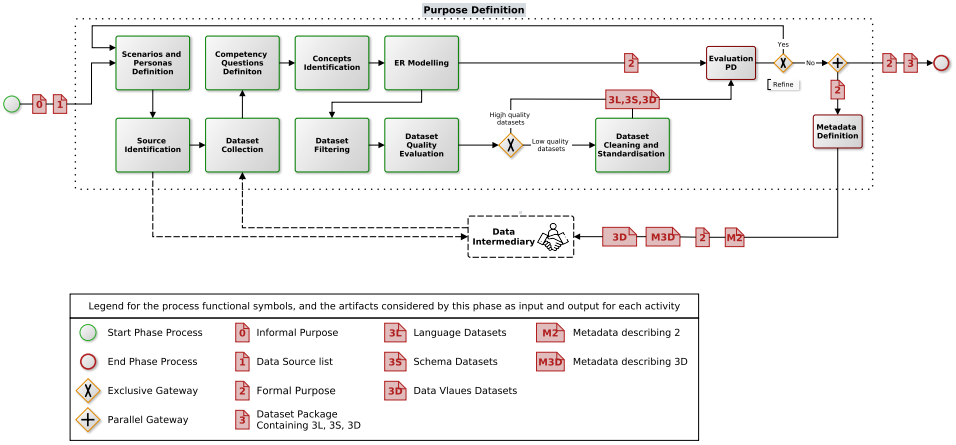


Figure 4.6: The Purpose Definition (PD) phase structure of the iTelos KGC process.

4.2.1 Purpose Definition - Activities

The PD phase of the iTelos KGC process includes different activities which aim at satisfying its twofold goal, namely formalising the initial user's Purpose and collecting the data required to build the final KG. In this subsection, we report a description of all the phase activities depicted in figure 4.6, by describing their inputs, outputs and objectives.

Scenarios and Personas Definition - The first activity of the PD phase, executed by the user playing the domain expert role, begins the process of formalization of the initial user's Purpose. Such an activity takes as input the object number 0, namely the informal Purpose. The informal Purpose is a natural language sentence expressed by the iTelos KGC process user (Domain Expert), which states the objective for which the process needs to be applied. The formalization of such initial Purpose follows the approach, depicted in figure 4.7, which aims at extracting specific information with the objective of extracting all the possible details about the data domain considered and the possible scenarios and personas (or actors) considered by the Purpose. Moreover, such an approach provides for the definition of possible Competency Questions (CQs) [48] representing more concretely the requirement set to be satisfied by the final KG. A dedicated activity is responsible

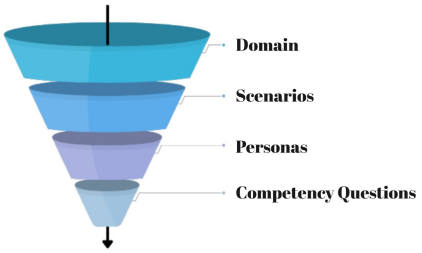


Figure 4.7: Purpose formalization approach

for the definition of the CQs (see below, Competency Question Definition). More in detail, the first three elements to be described following the Purpose formalization approach are:

- *Domain*: also defined as "Domain of Interest" (DoI), it refers to the area of knowledge or field of study of interest [40]. Examples of DoI, are the domains capturing knowledge about daily lives, such as music, tourism, and healthcare, or geographical domains, like Italy or Trentino autonomous province (Province of Trento, Italy), or even the city of Trento (Italy). The DoI is better detailed if it is defined with its boundaries:
  - Geographical boundaries: aspects that geographically constrain the problem. For example, the DoI extracted from the user's Purpose is contained within the city of Trento, thus it will not consider data regarding other cities.
  - Temporal boundaries: aspects that constrain the problem in time. For example, the DoI extracted for the user's Purpose considers data within a certain period of time (i.e., from 1992 to 2025).
  - Domain boundaries: domain-specific aspects constraining the problem. This boundaries are defined by the DoI itself. For example, if the DoI is "Healthcare", the user's Purpose can be restricted to (and thus consider data about) pharmacies only, without considering hospitals, even if hospitals are actually considered as part of the healthcare domain.
- *Scenarios*: a set of possible situations considered by the Purpose. The scenarios definition aim at detailing a given a DoI. For example, if the DoI is again the "Healthcare" a possible scenario could be the surgery department within a certain (or a set of) hospital during daytime (or during the night).
- *Personas*: also defined as scenario actors, the personas are semi-fictional subjects characterising the perception and needs of larger groups of end-users. The aim in defining the personas is to further specify the information implicitly represented by the informal Purpose, by providing a definition of the possible target users of the KG, suitable to satisfy the relative Purpose. An example of persona, related to the healthcare DoI is:

  "Matthew, 35 years old, is working as a surgeon in the hospital and used to work the night shift."

Source Identification - the second activity in the PD phase aims at identifying the data source provided as the second main input of the entire iTelos

KGC process. The input data source list is intended to be a list of data source names and, in the best case provide the link to access them. The goal of this activity, executed by the user playing the data scientist role, is to check each item of the data source list, and asses if the relative data source is actually available and accessible, as well as which kind of data it provides. It could be that a data source is no longer available, or that its access policy is changed. The data source may be a private repository, requiring permission from the owner to access it. In addition, one of the input data sources may provide very low-quality data, making it effectively useless to the project due to the high cost of accessing the data. In general, the sources considered can be different and distribute different types of data.

- *Low-quality sources*: the data distributed by these sources are less understandable (low-quality metadata) and less interoperable (non-standard, non-diversity-aware). The data is not always distributed, sometimes it is only published or visualized online. It is not always clear how to concretely access the data. Sometimes the source has to be processed to get the data (i.e., Scraping web pages), or it is required to ask for datasets directly to owners (if the owner contacts are indicated).

- *High-quality sources*: these sources are usually data catalogues where interoperable and reusable datasets are distributed (i.e. LiveData catalogues (see chapter 5), where the iTelos diversity-aware data is published). The data distributed by these sources requires less effort to be collected, adapted and integrated, due to clear policies of data distribution, direct distribution/download or distribution on demand through a request to the data owner (easy communication with data owners), as well as high-quality metadata, describing the available resources. This increases the findability and accessibility of the resources distributed.

The source identification activity focuses not only on the data sources received as input, but also on potential new data sources that should be considered to satisfy the user's Purpose. To this end, the activity benefits from the Purpose details extracted in the previous activity. Such more precise information, extracted from the informal Purpose, helps the user to better perform the source identification activity, being able to better understand what data is required to build the KG.

Dataset Collection - The third activity of the PD phase, follows on from the work done in the previous activity, by collecting the data from the sources previously identified. Even if the previous activity filtered out the part of the low-quality data sources, this activity, executed by the user playing the data scientist role, requires considerable effort. While some of the data to be collected can be directly downloaded from the sources, it could be the case

in which the data needs to be collected by processing the sources. For example, when the indicated source is a web page, the data collection activity requires a data scraping task to collect the required data directly from the web page source code. It could be the case, for specific Purposes, that the data are not available at all, so that it must be produced according to the Purpose. Among the several ways to produce the required data from scratch we mention the *iLog* methodology for fostering valid and reliable Big Thick Data [15]. As already mentioned above, describing the PD phase, the data collection activity aims at collecting not only data values dataset (i.e., row datasets like CSV, TSV, and others), but also object-oriented datasets like JSON and others), as well as data schema (knowledge data) and language data to be used in the next phases of the iTelos KGC process.

Competency Question Definition - The fourth activity of the PD phase, executed by the user playing the domain expert role, adds one more step in the formalization of the initial Purpose, by defining the last element considered by the Purpose formalization approach adopted by the process, as depicted in the figure 4.7. The goal of this activity is to extract and define concretely the requirements to be achieved by the final KG, shaped as Competency Questions (CQs). The CQs are a list of natural language questions, each of them defining a need that is satisfied by exploiting the KG to be produced. Each CQ is defined by considering one of the previously defined personas, acting in one, or more, previously defined scenarios. It clearly appears how the CQs define the set of natural language questions that will be then turned into possible KG's queries. During the definition of the CQs, one persona can be associated with more scenarios, as well as one scenario can include the interaction between two or more personas. This means that the more scenarios and personas that have been previously defined, the more opportunities there are to define a rich set of CQs. However, a rich list of CQs is not only a matter of quantity, but also a matter of diversity. In fact, a list of CQs where the same persona (thus having always the same needs) acts in the same scenario in different ways, introduces little variation regarding the information to be considered by the final KG to satisfy such CQs. On the other hand, instead, a set of CQs where all the personas have been associated with all the scenarios, represents properly the implicit diversity of the Purpose being defined, by introducing a huge variation regarding the information to be modelled and included into the final KG.

Concept Identification - The fifth activity of the PD phase is executed by the user playing the knowledge engineer role in the process. Such activity is the penultimate step to achieve a formal version of the initial user's Purpose. Even if the CQs, produced by the formalization approach implemented in the previous activities, already give a more detailed overview of the Purpose's requirements, after the Competency Question Definition activity the Purpose

(and its requirements) is still defined into a "textual" informal version. The goal of the Concept Identification activity and the subsequent ER Modelling activity is to produce a formal, and machine-readable, version of the Purpose. To this end, the current activity aims at identifying all the terms and expressing concepts, which represent the names of Entity Types (ETypes) as well as the names of their properties. The idea is to use such terms to model an Entity Relationship (ER) diagram in the next activity, for this reason, the terms collected by the current activities need to be representative of specific concepts, thus having a specific meaning to be then associated to ETypes and properties. Such terms have to be extracted from the CQs produced previously, thus including the definition of the scenarios and the personas. It is important to notice how the Concept Identification activity starts to put the focus on the ETypes and properties which will constitute the knowledge layer of the final KG. The terms collection is supported in this activity by a spreadsheet, where specific columns allow the user to specify from which scenarios, personas and CQs a relative term was extracted. A snapshot example of such a spreadsheet, called Purpose Formalization (PF) sheet, is reported in figure 4.8.

	A	B	C	D	E	F
1	**Scenarios**	**Personas**	**Competency Questions**	**Entities**	**Properties**	**Focus**
2	1,2,4	2,3,4	1,2,5,6	Bus	color, route_number, capacity,	Contextual
3	3	1	-	Airplane	type, ownership, registration_number, model	Common
4	-	1, 2, 4	1,3	Train	type, ownership, capacity, route	Core
5	1,2,4,5	2,3,4	1,4,5,6,7	Route	path, distance, start_point, end_point, speed_limit, intersections	Contextual
6	1	2, 4	1,2	Bus stop	location, served_routes	Core
7	-	1,4	3	Train station	location, served_routes,	Core
8	3	1	-	Airport	name, location, transporations	Common
9	5	-	7	Hospital	name, address	Common
10	5	-	7	Police station	name, address	Common
11						
12						
13						
14						
15						

Figure 4.8: Example of PF sheet.

As can be seen in the figure, there is a last column in the PF sheet which aims at categorizing the concepts identified (and represented by the relative terms) based on their focus on the relative Purpose. The *Focus*, as described in [32], is a value that indicates how much a generic object (in this case the concepts in the PF sheet) is related to a specific Purpose. In the PF sheet the Focus value is selected between three options:

- *Common*: the concept used to represent ETypes and properties with this Focus value are very general for the Purpose considered. In other words, this means that common concepts are useful for the Purpose, thus also for modelling the final KG, but they do not play a crucial role in supporting the Purpose requirements. They are usually concepts expressing general information, like time and space aspects.

- *Core*: the concept used to represent ETypes and properties with this Focus value are more important than the common one to satisfy the

Purpose requirements. The core concepts are used to represent ETypes and properties which constitute the base of the user's Purpose. They are usually concepts expressing information about the Purpose's DoI.

- *Contextual*: the concept used to represent ETypes and properties with this Focus value are very specific for the relative user's Purpose. The contextual concepts are used to define ETypes and properties which are fundamental for the satisfaction of the Purpose, without which it would not be possible to support the KG's queries.

It is important to note that as we move from common Focus values to contextual values, the information carried by the categorized concepts becomes more and more focused on the local diversity carried by the Purpose itself. Moreover, the Focus values provide information about the reusability of the concepts, and thus of the ETypes and properties represented by them, being common concepts more reusable for different Purposes, while the core and contextual concepts are less reusable due to their specificity about the relative Purpose. For this reason, the Focus values associated with the concepts in the PF sheet, represent an important information to be distributed, with the objective of enhancing the reusability of the diversity-aware data produced by the iTelos methodology.

ER Modelling - The sixth activity in the PD phase completes the process of formalizing the initial Purpose. The goal of this activity, executed by the user playing the knowledge engineer role in the process, is to produce an ER model representing a formal and machine-readable version of the user's Purpose. An ER model describes interrelated (real-world) entities of interest in a specific domain of knowledge. It is composed of classes of entities, or entity types (ETypes) which classify the real-world entities as instances of those ETypes and specify relationships that can exist between them. The ER model is, therefore, an abstract data model that defines a knowledge structure which can be implemented in a data/knowledge base. Although the ER model was born for relational database (DB) modelling, it is still a good instrument for modelling the structure of a KG, nevertheless the user, during the formal Purpose modelling has to be careful about the differences between database and KG modelling (see chapter 3). For example, the relationships between nodes (ETypes) are expressed in the KG modelling by using the object properties, while in relational DBs modelling they are expressed by using the the connection with primary and foreign keys. This difference is reflected also in the ER model where different graphical solutions can be adopted to represent such relationships. In the current activity, the user models the ER diagram of the Purpose, by taking as input the PF sheet with all the Purpose-specific concepts previously identified, as well as the list of CQs and the data collected by the Dataset Collection activity, to have a full overview about of the information to be included in the final KG (thus to

be represented in the Purpose ER model). With the execution of ER Modelling activity, the user achieves the formal definition of the initial, informal, Purpose. More in detail, the iTelos KGC process defines the formal Purpose as an object composed of the CQ list and the ER model. These two elements of the formal Purpose provide both the set of human-readable requirements and the standard formal version of the Purpose knowledge representation, respectively. Noticed that the latter element can be easily described using the RDF-OWL language, thus providing the machine-readable version of the formal Purpose.

Dataset Filtering - The generation of the ER model performed by the previous activity allows the process user(s) to have a clearer and precise understanding of the information, and the data, to be considered for the creation of the final KG. This situation enables the work to be done in the Dataset Filtering activity where the user, who plays the role of data scientist, under the guidance of the ER model, filters out the dataset collected in the Dataset Collection activity, to keep only those carrying the information represented by the ER model itself. This activity avoids working and spending effort on datasets that are not required to satisfy the formal Purpose. To this end, this activity takes as input the ER model and the set of datasets previously collected and produces in output a filtered set of such datasets.

Data Quality Evaluation - The eighth activity of the PD phase is executed by the user who plays the role of data scientist in the process execution, and it aims to ensure the quality of the data obtained from this phase. More in detail, this is an evaluation activity which can lead to the delivery of the data as they are, If the quality check is positive, or to perform an additional activity to achieve the desired level of data quality. The quality check performed in this activity aims to check if the datasets collected and handled by the previous activities are compliant with the FAIR principles [2]. Moreover, the quality check passes through the evaluation of the data in accordance with the 5 $\star$ open data scheme [3]. The comparison with both the FAIR and 5 $\star$ criteria has to be done by considering also the data owner or the iTelos KGC process users' data requirements (i.e., if the data needs to be kept private the 5 $\star$ open data scheme cannot be fully applied but only part of it will be considered.).

Dataset Cleaning And Standardisation - The last activity of the PD phase (the evaluation and metadata definition activities, for each phase, are defined separately at the end of the chapter.) is executed only if the quality check described in the previous activity has been negative. If that is the case,

[2] FAIR principles
[3] 5-star open data

the user who plays the data scientist role executes this activity by cleaning and standardising the data collected previously. The cleaning task included in this activity has the objective of removing the "noise" from the data to be handled by this activity. The data noise can be defined at different levels:

- dataset level: between the set of datasets to be cleaned there are corrupted datasets, namely datasets that cannot be read or opened, thus not processable. These datasets are recognized as noise and are not taken into account in the generation of the final KG.
- EType or entity level: in the datasets handled there are some ETypes (knowledge datasets) or entities (data values datasets) that are corrupted, thus having missing values or missing data structures. Those ETypes and entities are noise that also in this case will be removed from the set of data considered for the KG generation.
- Properties or attribute level: in the datasets handled there are some specific attributes (data values datasets) or properties (data or object properties into knowledge datasets) having corrupted values (like "null" values) thus resulting in useless for the generation of the final KG. Also in this case such values or properties are recognized as noise.

The standardisation task, instead, aims at aligning the data with respect to the FAIR and 5 $\star$ criteria (always by considering the user/Purpose specific requirements). It is important to notice how FAIR and 5 $\star$ are quality reference data standards supporting this activity. More in detail, they include the adoption of standard interoperable data formats, like CSV, TSV and JSON (and others), and also the adoption of standard data types for specific data values. For example, the date and time information can be expressed in different ways, and it is not rare to see the date encoded as a non-standard string (i.e., "23MAY2024", or "23052024"). The usage of standard date representation like Unix Time Stamp or ISO 8601, improves a lot the interoperability and reusability of the data.

4.2.2 Purpose Definition - Process

The above described activities are executed following a specific process that actually defines how to execute the PD phase, thus leading the user from the transformation of the phase's inputs into the above described final, and intermediate, outputs. To better describe such a process in this subsection we report the pseudocode describing the algorithm of the PD phase's process. The algorithm 1 is not included in the PD phase, but it is mandatory to describe it, as it shows the process that initialises the execution of the whole iTelos KGC process.

Algorithm 1 iTelosKGC(0, 1, UKC)

1: $LO = UKG.getLonguageOntology()$
2: $DI = DataIntermediaryLiveDataSpace$
3: $EB = PurposeDefinition(0, 1)$
4: Return EB

The pseudocode describing the algorithm 1 takes as input the objects number 0 and 1, in figure 4.6, namely the informal Purpose and the data sources list, respectively. Additionally, the *iTelosKGC* algorithm takes as input the UKC object allowing the user to access the language resources provided by the UKC. The algorithm starts by getting the UKC Language Ontology (LO) as it is described in chapter 3. Then, the algorithm obtains access to the second external component supporting the overall iTelos KGC process, the DI data space (the LiveData space as described in chapter 5). The initialization algorithm continues by defining the variable that will contain, at the end of the whole iTelos KGC process execution, the final result, namely the EB. The EB object, which will be returned by this top-level algorithm as the main output, is assigned to the result of the *PurposeDefinition* function, described here below by the pseudocode of the algorithm 2.

Algorithm 2 PurposeDefinition(0, 1)

1: $PS = ScenarionAndPersonaDefinition(0)$
2: $S = SourceIdentification(1)$
3: $3 = DatasetCollection(S)$
4: $CQ = CompetencyQuestionDefinition(0, PS, 3)$
5: $C = ConceptIdentification(CQ, 3)$
6: $ER = ERModelling(C, CQ, 3)$
7: $2 = \{CQ, ER\}$
8: $3 = DatasetFiltering(3, ER)$
9: **if** $DatasetQualityEvaluation(3)$ is False **then**
10: $\quad 3 = DatasetCleaningAndStandardisation(3)$
11: **end if**
12: **if** $EvaluationPD(2, 3)$ is False **then**
13: $\quad PurposeDefinition(2, 3)$
14: $\quad$ Return null
15: **end if**
16: $MetadataDefinition(2, DI)$
17: $LD = LanguageDefinition(2, 3)$
18: **if** LD is null **then**
19: $\quad$ Return null
20: **end if**
21: Return $\{2, LD\}$

The *PurposeDefinition* function defined by the algorithm 2, takes as input the objects numbered with 0 and 1, already described above. It is immediately clear how the different functions involved in the *PurposeDefinition* function pseudocode implement the activities with the same name described in the previous sub-section. The first algorithm statement executes the *ScenarionAndPersonaDefinition* activity by taking as input the informal Purpose (object number 0). Then the *SourceIdentification* activity is executed taking as input the data source list (object number 1) and identifying the available data sources. At this point, the identified data sources (object S in the algorithm 2) are provided as input to the *DatasetCollection* activity that collects the language, knowledge and data values datasets, identified by the objects named 3L, 3S and 3D in figure 4.6 composed together into a single stratified data package (object number 3). The next statement in the pseudocode is the definition of the CQs, executed by the *CompetencyQuestionDefinition* activity taking as input the informal Purpose (object number 0), the Personas and Scenarios definition (PS object), and the collected resources (object number 3). Then the algorithm continues by executing the *ConceptIdentification* activity taking as input the CQ and the dataset collected, and subsequently the *ERModeling* activity which takes as input the concept identified and defined previously (C object), the CQ list and the collected dataset. At this point, the algorithm 2 evaluates the result of the *DatasetQualityEvaluation* activity which takes as input the collected datasets. If the returned value is negative (False), the algorithm executes the *DatasetCleaningAndStandardisation* activity over the datasets collected to obtain a new version of them with the proper level of quality and interoperability. If instead, the output of the *DatasetQualityEvaluation* is positive (True) the algorithm continues its execution with the current version of the datasets collected. A second conditional statement is then executed by the algorithm, testing the result of the phase-specific evaluation activity, called *EvaluationPD*. Such evaluation activity, which will be better described in section 4.6, aims at checking if the datasets collected are suitable, in this phase, to satisfy the formalized Purpose. If the result of such a test is negative, it means that the datasets or the formalized Purpose need to be revised, thus the algorithm continues with a new execution of the *PurposeDefinition* function taking as input the formal Purpose and the datasets generated in the current algorithm iteration. If, instead, the result of the evaluation activity is positive the algorithm continues its flow with the execution of the *MetadataDefinition* activity to produce the metadata relative to the formalized Purpose produced by the current iTelos KGC process phase. The activity that, at each phase is focused on the definition of the metadata is better detailed in section 4.7. The last statements of the algorithm 2 aim to execute the next phase of the iTelos KGC process, by passing as input the formalized Purpose and the datasets collected to the *LanguageDefinition* function, described in details by the pseudocode of the algorithm 3. A last test in the

current algorithm aims at evaluating if the output of the *LanguageDefinition* function is properly defined. The algorithm describing the process of the PD phase returns as output a single object containing the formalized Purpose and the result of the process describing the LD phase.

The process above, describing the PD phase, has been designed to be suitable for both the data producer and the data consumer, thus being compliant with the two base principles of the iTelos methodology. On the data producer side, the formalization of the Purpose is focused on the details describing the geographical, social and individual context that needs to be expressed by the data. If the data producer is already aware of possible applications of the data she has to produce, then the application requirements will be part of the Purpose formalization process too. On the data consumer side instead, the Purpose formalization process is completely focused on a precise definition of the application-specific requirements to be satisfied. The key difference between the two top-level roles in this phase regards the effort to be spent for the data source identification, as well as for the dataset collection and their subsequent management. The data producer aims at producing new diversity-aware data, for this reason, she will consider what has been defined as external data spaces/sources, in section 4.1.1. The quality of the data collected by these sources is lower, thus requiring a major cost of pre-processing along the PD process activities. On the other hand, the data consumer considers as input data sources, the DI (internal) data space (LiveData), thus browsing and handling with diversity-aware data only. Such data has been already created by the iTelos KGC process, and, due to that, it is more suitable to be handled by the process itself (higher level of data quality and interoperability). This situation allows the data consumer to spend less effort on the data pre-processing.

4.2.3 Purpose Definition - Tools & Standards

The process implementing the PD phase, described by the algorithm 2, is executed by the joint collaboration of the users playing the roles of a domain expert, data scientist and knowledge engineer. To support the process execution, the iTelos methodology provides standard templates for the definition of some of the process intermediate outputs. More in detail the templates provide are:

- a project report template: this template is a document (provided PDF or LaTex) internally structured following an index which involves the description of each iTelos KGC process phase, as well as the description of the internal activities for each phase. Regarding the PD phase, the project report template provides specific sections for the textual definition of the DoI, the scenarios, the personas and the competence questions.

- The PF sheet template: already depicted in figure 4.8, it is used to collect all the concepts, extracted from the scenarios, personas and CQs, required to model the Purpose's ER diagram.
- Entity Relationship (ER) standard: the ER model is already a standard used to model the logical schema of relation DB. The PD phase is used to formalize the Purpose-specific information. In particular, we push the ER IDEF1X notation [4] for such a modelling task.

4.2.4 Purpose Definition - Example

We start in this subsection an example that will be carried out for each phase of the iTelos KGC process. In particular, in the current subsection, the example reports a simple Purpose used to drive the representation of an EB capable of satisfying it. The example Purpose is first expressed informally by a natural language sentence. Such an informal purpose identifies the data sources from which the data to be transformed into EB will be extracted and provided as input to the iTelos KGC process. The informal Purpose is defined as follows.

"A user is interested in developing a data-based service providing information about the train stations available in the Trentino Autonomous Province (Italy). The user collected the data to satisfy the above objective from the OpenDataTrentino [5] web portal, and OpenStreetMap [6]."

The formal version of the above informal Purpose, obtained by following the iTelos KGC process, is depicted by the graph in figure 4.9. The graph shows a graphical representation of the Purpose Teleology obtained from the above informal Purpose. The small example of Purpose Teleology depicted in figure 4.9 has two entity types (*Train Station*, and *Location*) linked by an object property (*has*). The data properties, representing the structure of the entity types, show how the *Train Station* type has two data properties defined following the schema of the data collected form OpenDataTretnino source (*Name* and *Code*, linked to OpenDataTretnino source by the dashed lines), and one data properties defined by the OpenStreetMap source data schema. The *Location* type structure, instead, has been fully defined following the OpenStreetMap source data schema (the two data properties are linked to the OpenStreetMap source by the dashed lines).

[4] https://en.wikipedia.org/wiki/IDEF1X
[5] OpenDataTretnino
[6] OpenStreetMap

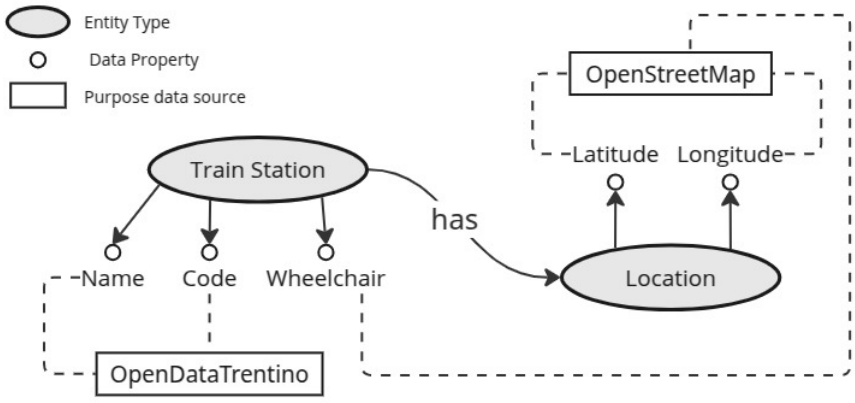


Figure 4.9: Purpose teleology example

Below we present the example data collected from the two data sources used to support the above purpose, representing the information about the railway station of the cities of Rovereto (Italy) and Trento (Italy).

```
<node id="1764381735" visible="true" version="11" changeset="141488181"
   timestamp="2023-09-19T22:43:54Z" user="Gurusraouta" uid="18913526" lat="46.0722416" lon="11.1193186">
 <tag k="alt_name" v="Trento Stazione FS"/>
 <tag k="name" v="Trento"/>
 <tag k="name:de" v="Trient"/>
 <tag k="name:fr" v="Trente"/>
 <tag k="name:it" v="Trento"/>
 <tag k="network" v="Rete Ferroviaria Italiana"/>
 <tag k="note" v="(progetto Angiolo Mazzoni)"/>
 <tag k="operator" v="RFI - Rete Ferroviaria Italiana"/>
 <tag k="operator:wikidata" v="Q1060049"/>
 <tag k="public_transport" v="station"/>
 <tag k="railway" v="station"/>
 <tag k="train" v="yes"/>
 <tag k="wheelchair" v="limited"/>
 <tag k="wikidata" v="Q1404881"/>
 <tag k="wikipedia" v="it:Stazione di Trento"/>
</node>
 <node id="1764381725" visible="true" version="8" changeset="159912603"
   timestamp="2024-12-04T10:17:42Z" user="AwFi" uid="18230056" lat="45.8908227" lon="11.0339061">
 <tag k="name" v="Rovereto"/>
 <tag k="network" v="RFI"/>
 <tag k="network:wikidata" v="Q1060049"/>
 <tag k="operator" v="RFI - Rete Ferroviaria Italiana"/>
 <tag k="operator:wikidata" v="Q1060049"/>
 <tag k="operator:wikipedia" v="en:Rete Ferroviaria Italiana"/>
 <tag k="public_transport" v="station"/>
 <tag k="railway" v="station"/>
 <tag k="railway:ref:DB" v="XIRR"/>
 <tag k="train" v="yes"/>
 <tag k="uic_ref" v="8300113"/>
 <tag k="wheelchair" v="yes"/>
 <tag k="wikidata" v="Q3970745"/>
 <tag k="wikipedia" v="it:Stazione di Rovereto"/>
</node>
```

Figure 4.10: OpenStreetMap Train Station entities

stop_id	stop_code	stop_name	stop_lat	stop_lon	zone_id
4813	3100PI	Rovereto. Stazione Trenitalia	45.89182	11.034134	3100
247	20125p	Trento Piazza Dante Stazione Fs	46.071756	11.119511	10110

Figure 4.11: OpenDataTrentino Train Station entities

4.3 Phase 2 - Language Definition

This section is dedicated to the description of the second phase of the iTelos KGC process, namely the Language Definition (LD) phase, depicted in figure 4.12 which provides more details about the phase's internal structure. This phase takes as input the outputs of the previous phase, thus the formal Purpose and the dataset package containing the datasets cleaned, formalized and standardised in the PD phase. The two inputs are the objects numbered 2 and 3, respectively, on the left side of figure 4.12. In the right-hand side of the figure instead, the objects labelled 2, 3S, 3D, and 4 represent the outputs of the LD phase process. The output objects are the formal Purpose (2), the schema and data values datasets (3S and 3D) that are propagated to the following phase, plus the Language Teleontology (4) which is the main output of the LD phase. The objective of the LD phase is to generate the second diversity-aware data object, focused on the language diversity layer, that will compose the EB, following the iTelos data representation process, defined in chapter 3. As already mentioned, such an object is the Language Teleontology (LT) which defines the concepts, and their representation, composing the domain vocabulary specific for the Purpose that is leading the iTelos KGC process execution (the formal Purpose received in input at this phase). The LD phase aims to address the data heterogeneity at the language layer. To this end, the key intuition is that the LT, produced in this phase, unifies the multiple local (contextual or individual) interpretations of the concepts interested by the user's Purpose, by specifying a representation as much as possible suitable for the Purpose itself. In other words, the LT acts as a unity across the language layer of data heterogeneity, by considering the local language diversity of the geographical, social, or individual context to which the Purpose refers. To this end, the LD phase exploits the UKC as top-level reference standard Language Ontology (LO), as already described by the iTelos data representation process in chapter 3. The following subsections describe in detail, the internal activities, their process workflows, as well as the tools and standards provided by the iTelos methodology to concretely implement the second iTelos KGC process phase.

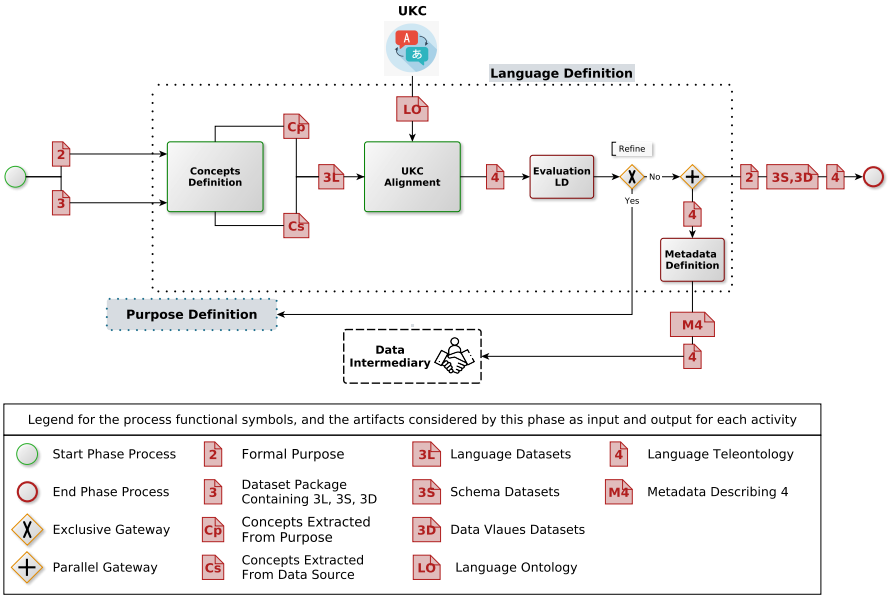


Figure 4.12: The Language Definition (LD) phase structure of the iTelos methodology.

4.3.1 Language Definition - Activities

The LD phase of the iTelos KGC process includes two main activities which aim at satisfying its goal, namely to generate the LT required to build the final KG, following the EB data model. The activities in the current phase have been designed to implement the language integration algorithm described in the section 3.3.2. In this subsection, we report a description of all the phase activities depicted in figure 4.12, by describing their inputs, outputs and objectives.

Concept Definition - The first activity of the LD phase, executed by the user playing the knowledge engineer role, takes as input the formal Purpose and the dataset package provided by the previous phase. It is important to remember that the formal Purpose is composed of the CQ list and the ER model, while the dataset package is a composition of data values, language and schema datasets. The objective of the first activity of the current phase is to collect, and formally define, all the concepts that need to be used to build the LT. Such concepts have to be used, in the next phases of the iTelos KGC process, to define ETypes, data and object properties, as well as specific semantic values for such properties. For example, if an EType has a property named "gender", the need is to find the right concept to express the meaning of the "gender" word, as well as the meaning of the words "male" or "female", or other values which could be used as values for the

"gender" property. The Concept Definition activity looks for the concepts to be defined in the activity's inputs. Therefore, the concepts are collected from the formal Purpose definition, as well as from all the data values and schema datasets included in the datasets package. It is important to state how the language datasets, included in the dataset package, are assumed to be already shaped as concept collections and, for this reason, are not considered as input for this activity. Nevertheless, it could be the case in which the concepts in the language dataset are not defined in the proper way, thus also the language datasets need to be handled in this activity. The formal definition of each concept collected follows the concept representation provided by the UKC. In particular, such a representation defines a concept with three main elements:

- the *concept ID*: it is a unique identifier to be associated with the concept. The concept ID is the language-agnostic representation of the concept. A temporary concept ID is defined by the iTelos KGC process user, during the current activity, to identify uniquely the concepts between the others defined for the current process Purpose. In the next activity (UKC Alignment) of the LD phase, the temporary concept ID will be so replaced with an official concept ID taking into account the ID of the concepts already defined formally in the UKC.
- The *concept's word*: it is a string value that represents in a natural or domain language, the relative concept. The word is the language-dependent representation of the concept, the one adopted by human users.
- The *concept's gloss*: it is a description of the concept's meaning. The concept's gloss is very important to let the human user understand the meaning associated with the given concept, which could not be properly understood just by reading the concept ID or its word.

	A	B	C	D	E
1	**ConcetID**	**Word-en**	**Gloss-en**	**Word-Other language**	**Gloss-Other Language**
2	UKC-45118	bus_stop	A place on a bus route where buses stop to discharge and take on passengers	<translation of Word-en in italian language>	<translation of Gloss-en in italian language>
3					
4					
5					
6					

Figure 4.13: Example of concept formally defined by the first activity of the LD phase.

Following the above formal concept representation, the concepts, collected from the current activity's inputs, are defined into a spreadsheet as depicted in figure 4.13. It is important to notice that the concepts can be listed down

with words and glosses for more than one language. This possibility is very helpful when the iTelos KGC process is adopted to satisfy Purposes considering international (cross-lingual) contexts, where the need is not only to translate a word from a language to one another but also to disambiguate the meaning of a concept coming from a different language and culture. The spreadsheet depicted in figure 4.13 represents the output of the Concept Definition activity, containing the formal representation of the concepts collected both from the formal user's Purpose and the source datasets. As depicted in figure 4.12, the two concept categories are listed within two different output files, namely Cp and Cs. This process design choice will be better detailed in the following sub-section describing the LD phase process flow.

UKC Alignment Definition - The second main activity of the LD phase, executed by the user playing the knowledge engineer role, takes as input all the files containing the formal definition of the concepts expressing the meaning of the information to be included in the final KG. More in detail, such files are, (i) the language datasets collected previously (the "3L" object in figure 4.12), (ii) the file containing the formal Purpose's concepts (Cp), and (iii) the file containing the concepts collected by the data values and schema datasets (Cs). Moreover, the fourth input of the current activity is the UKC LO representing the reference standard LO considered by the iTelos methodology. The objective of the UKC Alignment activity is twofold; firstly to reuse as much as possible the concepts already defined by the UKC into the LO, and secondly to use the UKC concepts to build the Purpose specific Language Teleontology (LT), shown in figure 4.12 as object number 4. This activity is actually responsible for the creation of the EB language layer, by implementing the first representational step of the iTelos data representation model. The first activity objective aims at reusing all the concepts already defined in the UKC's LO, that match those formally defined in the input files. If instead, the input concepts are not present in the LO, the user updates the UKC database by adding the new concepts, following the language integration algorithm already described in section 3.3.2. The execution of this activity updates the UKC database, by adding a new domain vocabulary that is specific to the user's Purpose. In other words, at the end of the activity execution, a portion of the UKC database (and thus also of the UKC's LO), identified by a specific namespace, maintains the definition of the concepts used to express the information considered by the user's Purpose, the one considered by the final KG. Moreover, these Purpose-specific concepts are linked to the rest of the UKC's LO structure through the relative concept hierarchies. This means that a Purpose-specific LT is now maintained within the UKC, which is then extracted to be provided as the output of the current activity, thus achieving the second activity objective. It is important to mention that, the update of the UKC database has to be supervised by a language expert to guarantee the right definition of the concepts hierarchies

within the UKC's LO structure. There are three important considerations to be made about the execution of the UKC alignment activity.

- The number of Purpose-specific concepts to be added in the UKC depends on how many concepts, about the Purpose's domain of interest, have been already uploaded in the UKC (or, in other words, represented into the UKC's LO).
- The concepts in the PF sheet (see figure 4.8 in the PD phase description) have been assigned to the "Common" *Focus* category, have more probability to be found in the UKC, being those concepts more general and used for a different Purpose. For the concepts categorized with "Core" and "Contextual" *Focus* values, such a probability decreases, and therefore it requires more effort from the user to update the UKC. In other words, the *Focus* can be used as a metric to evaluate the effort required by the user in this activity.
- An increasing adoption of the iTelos KGC process, for different Purposes, implies an increasing number of concepts added in the UKC. Therefore, the UKC will be updated with more, Purpose-specific, domain vocabularies and, over time, the effort required by users to define new concepts will actually be reduced.

4.3.2 Language Definition - Process

In this subsection, we report the process workflow that executes the activities described above. Such a process describes actually how the LD phase is executed, with the objective of generating the LT which defines the domain vocabulary specific to the user's Purpose. To better describe such a process in this subsection we report the pseudocode describing the algorithm of the LD phase's process.

Algorithm 3 LanguageDefinition(2, 3)

1: $CP = ConceptDefinition(2)$
2: $3SD = Join(3S, 3D)$
3: $CS = ConceptDefinition(3SD)$
4: $4 = UKCAlignment(CS, CP, 3L, LO)$
5: **if** $EvaluationLD(4, 2)$ is False **then**
6: $PurposeDefinition(2, 3)$
7: Return null
8: **end if**
9: $MetadataDefinition(4, DI)$
10: $KD = KnowledgeDefinition(2, 3SD, 4)$
11: **if** KD is null **then**
12: Return null
13: **end if**
14: Return $\{4, KD\}$

The algorithm 3 shows the pseudocode of the main LD phase function which takes as input the formal version of the user's Purpose (object number 2) and the dataset package from the previous phase (object number 3). In the first statement, the algorithm defines the CP object as a list of formally defined concepts extracted by formal Purpose. The value of CP is produced in output by the *ConcpetDefinition* function, described by the algorithm 4, which takes as input the formal Purpose (2). Such a function takes as input a collection of concepts, which can be concretely represented by a language, schema, or data values dataset, as well as by an ER model and a CQ list defining the formal version of the user's Purpose. The objective of such a function is to extract and formally define the concepts present in the input provided. The *ConcpetDefinition* function extracts concepts from the collection one by one, and for each concept, it defines three elements: an ID, a representative word, and a gloss. Then, the function creates a single object as the composition of such three elements, representing the formal version of the concept processed. The formal concept is then added to a list that at the end of the function execution is returned as output. The pseudocode of the *LanguageDefinition* function continues its execution by creating an object (3SD) that composes the schema datasets (3S) and the data value datasets (3D), collected from the input dataset package (3). Then the *ConcpetDefinition* function is applied again, by taking in input the object created in the previous statement, to extract and formally define the concepts from the input datasets. The result of the second execution of the *ConcpetDefinition* function is stored in the CS object as depicted in the algorithm 3. At this point, the *UKCAlignment* function is executed, by taking as input the objects CS, and CP, as well as the UKC's LO and the language datasets (3L) extracted from the input dataset package. The

UKCAlignment function returns, in object number 4, the LT produced by following the algorithm described in section 3.3.4.

Algorithm 4 ConceptDefinition(ConceptCollection)

```
1: FormalConceptList = []
2: for each Concept c in ConceptCollection do
3:     word = WordDefinition(c)
4:     gloss = GlossDefinition(c)
5:     id = IDDefinition(c)
6:     formalConcept = {word, gloss, id}
7:     FormalConceptList.add(formalConcept)
8: end for
9: Return FormalConceptList
```

A conditional statement is then executed by the algorithm, testing the result of the phase-specific evaluation activity, called *EvaluationLD*. Such evaluation activity, which will be better described in section 4.6, aims at checking if the LT (object number 4) produced is suitable, in this phase, to satisfy the formalized Purpose (object number 2). If the result of such a test is negative, it means that the LT or the formalized Purpose need to be revised, thus the algorithm continues going backwards with a new execution of the *PurposeDefinition* function taking as input the formal Purpose and the dataset package received as input. This process comeback allows the user to refine the input and outputs of the already executed activities to achieve the expected results. If, instead, the result of the evaluation activity is positive the algorithm continues its flow with the execution of the *MetadataDefinition* activity to produce the metadata relative to the Purpose-specific LT produced by the current iTelos KGC process phase. This activity, which at each phase is focused on the definition of the metadata, is better detailed in section 4.7. The next statements of the algorithm 3 aim to execute the next phase of the iTelos KGC process, where the input is the formalized Purpose (2), the dataset package composed by the schema and data value datasets (3SD), and the LT (4), to the *KnowledgeDefinition* function, described in details by the pseudocode of the algorithm 5. A last test in the current algorithm aims at evaluating if the output of the *KnowledgeDefinition* function is properly defined. The algorithm, describing the process of the LD phase, returns as output a single object containing the Purpose-specific LT (4) and the result of the process describing the KD phase.

Also for this phase, the process described above has been designed to be suitable for both the data producer and the data consumer, thus being compliant with the two base principles of the iTelos methodology. The key difference between the two top-level roles, in this phase, regards the effort to be spent

for the formalization of the process, as well as for their alignment with the UKC and the subsequent generation of the LT. On the data producer side, the number of concepts to be formalized and aligned with the UKC is higher, this is caused by the work that the data producer does over low-quality data and non-diversity-aware data received in input in the LD phase. As a consequence, also the effort, required by the data producer user in this phase is higher. On the data consumer side, instead, the dataset package received as input, in the current phase, contains already diversity-aware data and, in particular, language datasets where the concepts have been already formalized. Moreover, being the data received as input already produced by the iTelos KGC process, the language datasets concepts will be already aligned with the UKC (thanks to the UKC update executed previously by the data producer). This means that the data consumer has to formalize and align with the UKC's LO, only the concepts carried by the formal Purpose (user's Purpose-specific concepts). This results in a significant reduction in the effort required in the LD phase.

4.3.3 Language Definition - Tools & Standards

The process implementing the LD phase, described by the algorithm 3, is executed by the users playing the roles of knowledge engineer with the collaboration of the domain expert user, to help in the concepts disambiguation process. To support the process execution, the iTelos methodology provides standard templates and tools for the definition of some of the process intermediate outputs. More in detail the tools and templates provided are:

- a project report template: this template document has been already discovered in the previous phase. However, like for the PD phase, the document index has a dedicated section for a description of the activities involved in the LD phase, as well as for the description of the final and intermediate outputs.

- The concept definition spreadsheet template: already depicted in figure 4.13, it is used to collect and formally define all the concepts, extracted from both the formal Purpose and the input dataset package.

- The UKC database: access to the UKC database is provided to execute the UKC Alignment activity in the LD phase. UKC database can deployed and accessed as a stand-alone component in a local machine, as well as accessed in a read-only mode through the web portal [7]. The stand-alone deployment allows for maintenance and updates of a local instance of the UKC's LO. However, a UKC expert has to support the iTelos KGC process user during the usage of this component. On the other hand, the web portal provides a user-friendly graphical interface,

[7] http://ukc.disi.unitn.it/

which allows non-expert users to interact with the UKC, by looking for concepts which can be reused to represent the information required by the user's Purpose. Nevertheless, in the latter case, the user is able to produce a (or more) list of formalized concepts, aligned with the UKC's LO on IDs, words and glosses, but she has to ask for the support of the UKC expert to properly update the central UKC database, that will update also the view of the concepts available on the web portal.

4.3.4 Language Definition - Example

The example started in section 4.2.4 continues in this subsection, by showing the main output of the second phase of the iTelos KGC process, namely the LT. More in details, figure **??** shows the list of concepts considered by the Purpose which have been aligned with the UKC.

UKC-ID	Word-en	Gloss-en	Word-it	Gloss-it
21898	Train Station	terminal where trains load or unload passengers or goods	Stazione Ferroviaria	terminale in cui i treni caricano o scaricano passeggeri o merci
2	Name	a language unit by which a person or thing is known	Nome	un'unità linguistica con la quale si conosce una persona o una cosa
33626	Code	a coding system used for transmitting messages requiring brevity or secrecy	Codice	un sistema di codifica utilizzato per la trasmissione di messaggi che richiedono brevità o segretezza
24978	Wheelchair	a movable chair mounted on large wheels; for invalids or those who cannot walk; frequently propelled by the occupant	Sedia a rotelle	una sedia mobile montata su grandi ruote; per gli invalidi o per coloro che non possono camminare; spesso spinta dall'occupante
50	Location	a point or extent in space	Luogo	un punto o un'estensione nello spazio
45423	Latitude	the angular distance between an imaginary line around a heavenly body parallel to its equator and the equator itself	Latitudine	la distanza angolare tra una linea immaginaria intorno a un corpo celeste parallela al suo equatore e l'equatore stesso
45429	Longitude	the angular distance between a point on any meridian and the prime meridian at Greenwich	Longitudine	la distanza angolare tra un punto di un qualsiasi meridiano e il meridiano primo di Greenwich

Figure 4.14: Entity Base - Language Teleontology example

The LT produced by the third phases is, in concrete terms, part of the structure of the UKC LO in relation to the concept listed in the table above (Figure 4.14). Such concepts are then represented by the information provided by the UKC ID, word and gloss (provided for different languages where possible), and by their position in the UKC LO hierarchical structure. Figure 4.15 shows a graphical example of the structure of the UKC LO, incorporating the concepts of the example Purpose specific LT. It is important to note that, up to a certain level of generality (the dotted lines indicate multiple Is-A relationships between two concepts), the concepts used to represent the two Purpose's ETypes are part of the same tree structure.

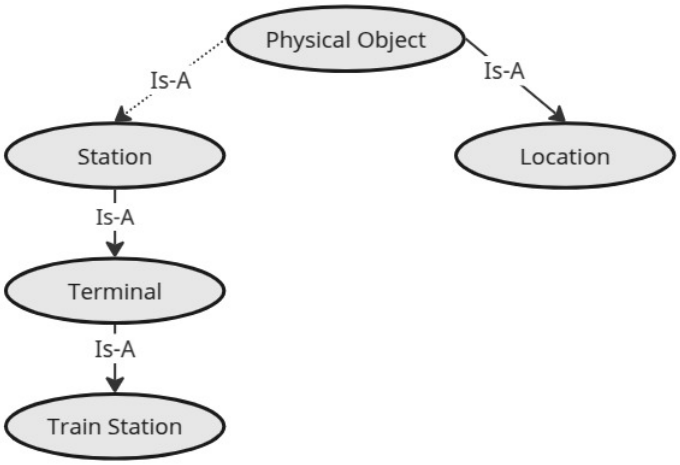


Figure 4.15: Entity Base - UKC concept structure example

4.4 Phase 3 - Knowledge Definition

This section is dedicated to the description of the third phase of the iTelos KGC process, namely the Knowledge Definition (KD) phase, depicted in figure 4.16 which provides more details about the phase's internal structure. This phase takes as input the outputs of the previous phase, thus the formal Purpose (object number 2 in figure 4.16), that drives the creation and the evaluation of each phase's output, the LT produced in the previous phase (object number 4), and the dataset package. It is important to notice that the input dataset package is composed only by the data value datasets (3D) and the schema datasets (3S). The language datasets are no longer included in the package since they have been transformed into the LT in the previous phase. In the right-hand side of the figure instead, the objects labelled 2, 3D, 4 and 5 show which are the outputs of the KD phase process. They represent the formal Purpose, the data value dataset and the LT that are propagated ahead to the next phase, plus object number 5 that represents the teleology, or Entity Type Graph (ETG) of the final KG. The teleology constitutes also the knowledge layer of the EB produced by the iTelos KGC process, as defined by the EB data model in the iTelos data representation process (see chapter 3). The KD phase aims at addressing the data heterogeneity at the knowledge layer. To this end, the key intuition is that the Purpose-specific teleology, produced in this phase, unifies the multiple local (contextual or individual) representations of the Entity Types (ETypes) interested in the user's Purpose. To this end, the teleology is modelled by specifying a representation as much as possible suitable for the Purpose itself, but also reusing as much as possible already existing domain-specific standard EType representations. Such an approach aims at maintaining the

focus both on the diversity of the user Purpose, which leads the entire KGC process and on the diversity expressed by already existing knowledge models (ontologies and data schema). In other words, teleology acts as a unity across the knowledge layer of data heterogeneity, by considering the local knowledge diversity (ETypes representation) of the geographical, social, or individual context to which the Purpose refers. The following subsections describe in detail, the internal activities, their process workflows, as well as the tools and standards provided by the iTelos methodology to concretely implement the third iTelos KGC process phase.

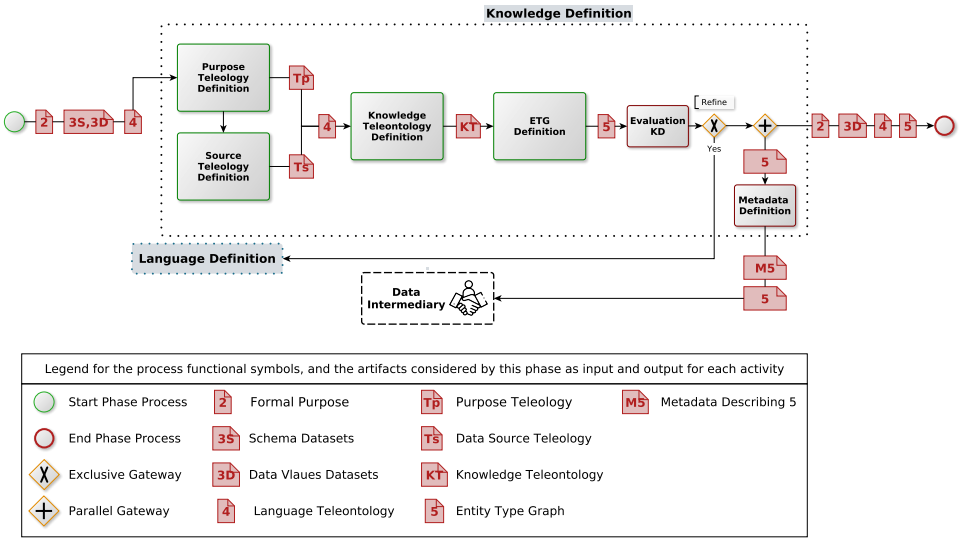


Figure 4.16: The Knowledge Definition (KD) phase structure of the iTelos methodology.

4.4.1 Knowledge Definition - Activities

The KD phase of the iTelos KGC process includes four main activities which aim at satisfying its goal, namely to generate the teleology required to build the final KG, following the EB data model. The activities in the current phase have been designed to implement the knowledge representation and integration approach described in the section 3.3.3. In this subsection we report a description of all the phase activities depicted in figure 4.16, by describing their inputs, outputs and objectives, to show how the knowledge representation model defined above is integrated into the overall iTelos KGC process.

Purpose Teleology Definition - The first activity of the KD phase, executed by the user playing the knowledge engineer role, takes as input the formal Purpose (object number 2 in figure 4.16) and aims to produce a

Purpose-specific teleology. The teleology modelling, as it was described in section **??**, is used in this activity to create a knowledge representation model that includes all and only the ETypes and the properties considered in the formal version of the user's Purpose taken as input. Such a teleology is shaped into an RDF-OWL file identified by the Tp object in figure 4.16.

Source Teleology Definition - The second activity of the KD phase, executed by the user playing the knowledge engineer role, takes as input the dataset package composed by the data values and schema datasets (objects number 3S and 3D in figure 4.16) and aims to produce a source-specific teleology. The teleology modelling approach is the same as in the previous activity, however, in this activity, it is used to create a knowledge representation model that includes all and only the ETypes and the properties considered in the datasets extracted (and formalized) from the data sources, during the first phase (see section 4.2). Such a teleology, is shaped into an RDF-OWL file identified by the Ts object in figure 4.16, and it includes standard knowledge representations and well-known reference ontologies about the domain of interest considered by the Purpose. It is important to notice that source-specific teleology is produced only after the generation of the Purpose-specific teleology. This design choice has been made to guide the user first through a more formal representation of the Purpose, and then to the formal definition of the knowledge available for such a Purpose, by introducing an additional filter over the datasets considered (3D and 3S). In fact, at the end of the second activity of the KD phase, the two teleologies produced, namely Tp and Ts, could include different representations of the same ETypes, and relative properties (actually this is the case for a considerable number of the ETypes considered). This happens because the two teleologies are used to express the knowledge diversity regarding the same domain but with a different "point of view". The Tp models such diversity from the user's "point of view", or in other words it represents the user's personal (or individual) context related to the Purpose, while the Ts models the reference context related to the Purpose, thus describing how the same knowledge is interpreted by other agents (the data source) acting in the same environment as the user.

Knowledge Teleontology Definition - The third activity of the KD phase, also executed by the user playing the knowledge engineer role, takes as input the source-specific teleology (Ts) and the LT (object number 4 in figure 4.16). The objective of the current activity is to merge the input knowledge models into a single object, called Knowledge Teleontology (KT), having two key features:

- *Reusability*: the representation of the ETypes and their properties in the KT are reused as much as possible from those defined in Ts. The

reuse of domain reference standard ETypes representation from Ts enhances the interoperability of the KT, and thus the possibility to reuse it (in future) for different Purposes.

- *Language based*: the ETypes and the properties in KT are named by using the concepts defined in LT. For this reason, the conceptual hierarchical structure defined by LT is adopted in KT as well, thus defining the knowledge (or EType) hierarchical structure. The KT, as already described in section 3.3.3, is a model where the data heterogeneity has been handled both at the language and knowledge layer. The concepts, disambiguated, formally defined and aligned with the UKC's LO, are used in the LT to fix the interpretation of the Etypes and the relative properties, based on the same user's Purpose (considered by all the iTelos KGC process phases, from the beginning to the end).

To summarize, the KT is generated by merging the reference domain knowledge models adopted to represent ETypes and properties following domain-specific and well-known standards, and the LT which provides the formal concept definition within a hierarchical knowledge structure.

ETG Definition - The last core activity of the KD phase executed by the user playing the knowledge engineer role, takes as input the KT modelled by the previous activity, and the Purpose-specific teleology (Tp), to produce in output a teleology perfectly suitable to satisfy the initial user's Purpose. The output teleology, namely the ETG of the final KG (depicted in figure 4.16 by object number 5), is produced by merging the ETypes and properties definitions expressed by the two input knowledge models. More in detail, the representations of the ETypes and their properties in the ETG, are reused as much as possible from those defined in KT, if and only if their expressiveness (the knowledge they represent) is equal or sufficient (based on a user-defined threshold) to the expressiveness of the Etypes used to represent the same real-world entities in Tp. In case the ETypes and the properties represented in KT are not suitable for the relative Purpose, the ETypes and properties from Tp will be used to represent the same real-world entities for which KT has been modelled. By keeping the ETG modelling focused on the Purpose, the iTelos KGC process develops a knowledge model suitable to satisfy the requirements, defined by the users, to build the final KG (the EB). It is important to notice that, being a teleology the ETG is not focused on the modelling of the ETypes hierarchical structure, as it was defined in the KT. The ETG models all and only the ETypes interested in the formal Purpose, thus actually the ones represented in Tp. This modelling choice has been taken to produce an ETG that is not too structurally complex to be processed by the applications and services which will exploit the final KG build with the ETG itself. To summarize, the ETG teleology is generated by merging the Purpose-specific knowledge model, representing the Purpose to

be achieved for the final KG, and the interoperable standard domain-specific model represented by the KT. The result of this merge is a knowledge model which is both highly reusable and interoperable and suitable to satisfy the initial user's Purpose.

4.4.2 Knowledge Definition - Process

In this subsection, we report the process workflow that executes the activities described above. Such a process describes actually how the KD phase is executed, with the objective of generating the teleology which defines the ETG of the final KG. To better describe such a process in this subsection we report the pseudocode describing the algorithm of the KD phase's process.

Algorithm 5 KnowledgeDefinition(2, 3SD, 4)

1: $TP = PurposeTeleologyDefinition(2)$
2: $TS = SourceTeleologyDefinition(3SD)$
3: $KT = KnowledgeTeleontologyDefinition(4, TS)$
4: $5 = ETGDefinition(KT, TP)$
5: **if** $EvaluationKD(5, 2, 4)$ is False **then**
6: $3 = composePackage(3SD, 4)$
7: $LanguageDefinition(2, 3)$
8: Return null
9: **end if**
10: $MetadataDefinition(5, DI)$
11: $3D = getDataValueDatasets(3SD)$
12: $ED = EntityDefinition(2, 3D, 4, 5)$
13: **if** ED is null **then**
14: Return null
15: **end if**
16: Return $\{5, ED\}$

The algorithm 5 shows the pseudocode of the main KD phase function which takes in input the formal Purpose (2), the dataset package composed by the data values and schema datasets (3SD), and the LT (4). The first statement of the algorithm defines the TP object representing the Purpose-specific teleology. The TP object is produced in output by the *PurposeTeleology-Definition* function, which takes in input the formal Purpose to implement the first activity of the KD phase. Similarly, the second phase activity is implemented by the *SourceTeleologyDefinition* function which takes in input the data values and schema datasets and, as shown in the second algorithm statement, produces the source-specific teleology that is then assigned to the TS object. The third statement, instead defines the KT object by executing the *KnowledgeTeleontologyDefinition* function which takes in input the LT

and the source-specific teleology, to produce the knowledge teleontology. The main output of the KD phase, namely the ETG (object numbered with 5 in the algorithm 5), is produced by the *TeleologyDefinition* function that takes in input the KT and the Purpose-specific teleology, as already detailed above in the description of the relative activity. A conditional statement is then executed by the algorithm, testing the result of the phase-specific evaluation activity, called *EvaluationKD*. Such evaluation activity, which will be better described in section 4.6, aims at checking if the ETG (object number 5) produced is suitable, in this phase, to satisfy the formalized Purpose (object number 2). If the result of such a test is negative, it means that the ETG or the formalized Purpose need to be revised, thus the algorithm continues going backwards with a new execution of the *LanguageDefinition* function taking as input the formal Purpose and the dataset package (object 3 in algorithm 5) composed by the data values and schema datasets (3SD) and the LT (being the already processed version of the language datasets). Also in this phase, this process comeback allows the user to refine the input and outputs of the already executed activities to achieve the expected results. If, instead, the result of the evaluation activity is positive the algorithm continues its flow with the execution of the *MetadataDefinition* activity to produce the metadata relative to the ETG produced by the current iTelos KGC process phase. This activity, that at each phase is focused on the definition of the metadata, is better detailed in section 4.7. The next statements of the algorithm 5 aims to execute the next phase of the iTelos KGC process, by passing as input the formalized Purpose (2), the dataset package composed only by data value datasets (3D), the LT (4) and the ETG (5) to the *EntityDefinition* function, described in details by the pseudocode of the algorithm 6. A last test in the current algorithm aims at evaluating if the output of the *EntityDefinition* function is properly defined. The algorithm, describing the process of the KD phase, returns as output a single object containing the produced ETG (4) and the result of the process describing the KD phase.

With the objective to be compliant with the two base principles of the iTelos methodology, the process of this phase, like the previous ones, has been designed to be executed by both the data producer and the data consumer. The key difference between the two top-level roles, in this phase, regards mainly the "quality" of the source datasets (3D and 3S) received in input, and, as a consequence, the effort to be spent to formalize such datasets into domain-specific teleologies. On the data producer side, the datasets received as input are collected from reference standard knowledge and data sources. However, such sources provide those resources (ontologies and data schema) with a different level of "quality". The "quality", in this case, defines how much the ETypes and their relative properties are understandable by the user who is looking for those. The meaning of such ETypes and properties could be not clearly understandable due to different natural (or domain)

languages used to represent them, as well as for the lack of metadata detailing the domain of interest considered by the resource. In this case, the effort to be spent on the Source Teleology Definition activity is considerable, although inevitable. On the data consumer side, instead, the datasets received have been already processed by the iTleos KGC process. In other words, the 3S object, depicted in figure 4.16 and used in the algorithm 5, represents a set of source-specific teleologies already aligned with the UKC's concepts. This means that they are reference domain-specific knowledge representations (ETypes and properties) that have been already merged with a LT aligned with the UKC's LO, thus having a clear and unambiguous definition. In other words, the Source Teleology Definition activity is costless, and also the Knowledge Teleontology Definition activity will be a matter of concept ID matching between the LT concepts and the concepts used to define the ETypes and their properties into the source-specific teleologies received in input. The KD phase, as it is executed by the data consumer, becomes a semi-automatic process driven by the user who is still required for the definition of the Purpose-specific teleology. The cost of execution in the current phase is considerably reduced on the data consumer side, thanks to the diversity-aware data produced and distributed by the data producer.

4.4.3 Knowledge Definition - Tools & Standards

The process implementing the KD phase, described by the algorithm 5, is executed by the users playing the role of knowledge engineer with the collaboration of the domain expert user, to help in the Purpose-specific teleology definition. To support the process execution, the iTelos methodology provides reference data standards and tools for the definition of the process intermediate outputs. More in detail, the tools and templates provided are described here below.

- RDF and OWL reference data format and language: the Resource Description Framework (RDF) is a method to describe and exchange graph data, became a standard for data interchange on the Web[8]. The iTelos methodology adopts RDF to define the knowledge layer resources, like ontologies, data schema, teleologies and teleontologies, as well as the final KG produced. The W3C Web Ontology Language (OWL), instead, is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things[9]. OWL is the language adopted by the iTelos methodology to represent knowledge layer resources in RDF files. All the core activities in the KD phase (detailed in the current section) adopt RDF and OWL to generate their outputs.

[8] https://www.w3.org/RDF/
[9] https://www.w3.org/OWL/

- Protègè knowledge modelling tool[10]: to support the user in the definition of the teleologies and teleontologies to be produced in the KD phase, the iTelos methodology provides the Protègè, as reference standard tool for the modelling of knowledge resources. Moreover, the tool is available online free of charge.

4.4.4 Knowledge Definition - Example

The example started in section 4.2.4 continues in this subsection, by showing the outputs of the third phase of the iTelos KGC process, namely the KT and the ETG. More in details, figure 4.17 shows the KT generated by adopting the OpenStreetMap data schema as reference knowledge to enhance the interoperabiltiy and reusability of the diversity-aware data produced by iTelos. In figure 4.17 the EType *"pointRailwayStation"* includes the data properties for both the Purpose's ETypes (see figure 4.9). For this reason, the user decided to adopt that single ETypes to support the example Purpose, being in this way interoperable with other OpenStreetMap data.

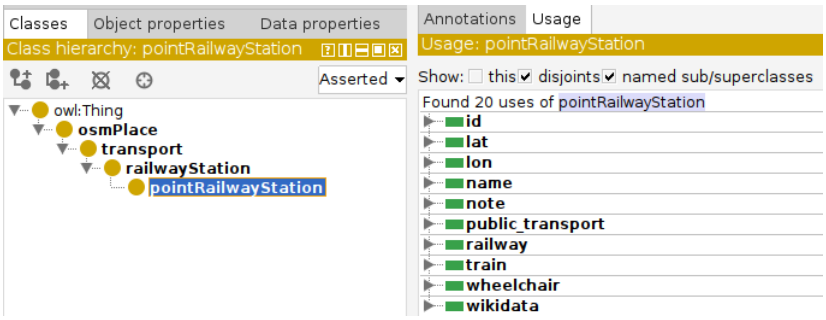


Figure 4.17: Entity Base - Knowledge Teleontology example

The KT shown above is then transformed into the ETG as the main output of the third phase. The ETG, shown in figure 4.18, is obtained by pruning from the KT all those ETypes that are not strictly relevant to the purpose to be supported, resulting in a knowledge structure more focused on the purpose, less complex to process, but at the same time following the standard reference knowledge adopted (OpenStreetMap, in the proposed example). To this end, only the EType *"pointRailwayStation"* and its properties are included in the ETG.

[10] https://protege.stanford.edu/

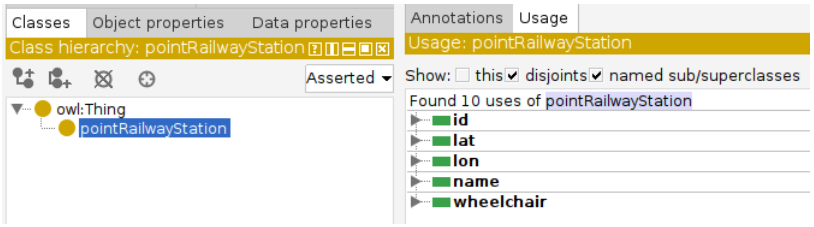


Figure 4.18: Entity Base - ETG example

4.5 Phase 4 - Entity Definition

This section is dedicated to the description of the last phase of the iTelos KGC process, namely the Entity Definition (ED) phase, depicted in figure 4.19 which provides more details about the phase's internal structure. Like in the previous cases, the ED phase takes as input the outputs of the previous phase, namely the formal Purpose (object number 2 in figure 4.19), which also in this last phase drives the creation of the phase's output, the dataset package composed only by data value datasets (3D), the LT (4) and the ETG (5) produced by the previous phase. It is important to notice, how the language and schema datasets, previously included in the initial sources datasets package (object number 3 in figure 4.6), have been transformed into the LT and the ETG, respectively. The input data value datasets (3D) will be transformed too in the current phase, to produce the final process output. The outputs of the last phase of the iTelos KGC process, as depicted in the right-hand side in figure 4.19, are the four elements composing the EB, as described by the EB model in chapter 3. Such output, and EB, components are, the formal user Purpose (object number 2), the LT (object number 4), the ETG (object number 5), and the last phase main output, namely the Entity Graph (EG) (object number 6). The ED phase aims at addressing the data heterogeneity at the data value layer, by generating the final KG's entities shaped as instances of the ETypes modelled into the input ETG. Such entities are concretely represented by the values, extracted from the data value datasets, and mapped on the ETG's ETypes and properties. As already discussed in section 1.2, multiple data values can be adopted to represent the same property. For this reason, the key intuition, in the ED phase, is that the EG produced unifies the multiple local (contextual or individual) representations of the entities interested in the user's Purpose. Also in this last phase, the Purpose leads the decision taken by the user regarding the most suitable data value representation to be adopted, however, to increase the interoperability and the reusability of the final KG, the process pushes the user to reuse as much as possible standard data types, as well as standard identifiers to uniquely identify the entities generated. In other words, the EG produced acts as a unity across the data value layer of data heterogeneity,

by considering the local data value diversity (entity representation) of the geographical, social, or individual context to which the Purpose refers. The following subsections describe in detail, the internal activities, their process workflows, as well as the tools and standards provided by the iTelos methodology to concretely implement the last iTelos KGC process phase.

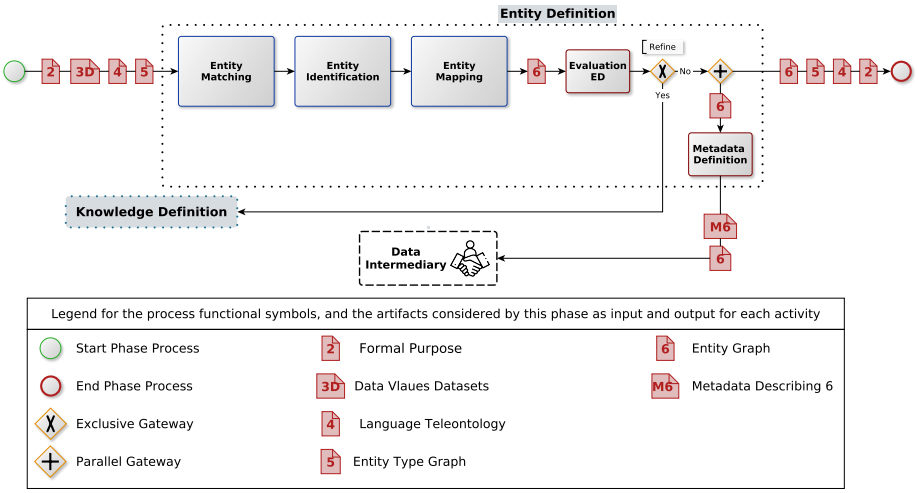


Figure 4.19: The Entity Definition (ED) phase structure of the iTelos methodology.

4.5.1 Entity Definition - Activities

The ED phase of the iTelos KGC process includes three main activities which aim at satisfying its goal, namely to generate the EG as the final KG and the last component of the EB. The activities in the current phase have been designed to implement the entity matching algorithm described in the section 3.3.4. In this subsection we report a description of all the phase activities depicted in figure 4.19, by describing their inputs, outputs and objectives, to show how the Purpose-specific real-world entities are generated by the last phase of the iTelos KGC process.

Entity Matching - The first activity of the ED phase, executed by the user playing the data scientist role with the support of the knowledge engineer, takes as input the data value datasets (object number 3D in figure 4.19), and the ETG (object number 5) produced in the previous phase. The objective of this activity is to recognize, between the data value datasets, the entities having the same, or similar, schema as the ETG's ETypes modelled to represent the same real-world entities. Such an objective is achieved by matching the schema of the data values datasets with the ETG, using the entity matching algorithm as already described in section 3.3.4. The output

of the activity is an updated version of the data values datasets, where the entities instantiating a specific EType are all grouped into the same datasets (or a set of datasets if the number of entities is considered too big for a single dataset). This last step of the current activity aims to build data value datasets, where the entities are easily recognisable and thus suitable to be merged with the ETG to build the final KG, as will happen in the last activity of the ED phase.

Entity Identification - The second activity of the ED phase is executed by the user playing the data scientist role and takes as input the data values dataset updated in the previous activity. The entities in such input datasets have been properly ordered by grouping those instantiating the same ETypes into the same datasets. However, there is still a crucial issue to be solved to achieve a formal and high-quality entity representation, that involves the properties and properties values used to uniquely identify each real-world entity to be considered in the final KG. We consider two main approaches for the generation of such identification properties. The first one aims to define the value of a single property, called *Identifier*, used to uniquely identify all the entities of specific ETypes. Nevertheless, since it is not possible to be sure that an identifier value (when it is available) is unique for all the digital systems in the world, the key point is the generation of an identifier value able to uniquely identify an entity in a specific context. The boundaries of such a context are not detailed and depend on how many applications, services and, more in general, data systems can take place on it. The iTleos methodology supports the user in the generation of identifier values, by providing standard ways to build such values. More in detail, the type of identifiers considered by the iTelos methodology is based on the definition of *URI*. A Uniform Resource Identifier (URI) is a unique sequence of characters that identifies a logical or physical resource used by web technologies. The iTelos methodology allows the usage of URI for the identification of entities, and also for the identification of ETypes. Examples and more details can be found directly in online documentation[11]. To be more specific, a URI can be defined as:

- *URL*: A Uniform Resource Locator (URL) is a URI that specifies the means of acting on or obtaining the representation of a resource, i.e. specifying both its primary access mechanism and its network location.
- *URN*: A Uniform Resource Name (URN) is a URI that identifies a resource by name in a particular namespace.

The second approach for the generation of identification properties is based on the composition of different properties, of the same EType, which together

[11] Wikipedia URI

are able to uniquely identify the entities instantiating the relative EType. This composition of properties is called *Identifying Set*. More precisely, an Identifying set is a set of EType properties which, through their values, identify uniquely an entity (defined for such an EType) within the whole set of entities considered. The identifying set is a technique that can be used to recognize the existence of a (or more) entity (ies) in a specific dataset, by identifying the unique composition of values with respect to all the other entities related to the same EType. The composition of the approaches, for the generation of identification properties, is adopted by the Entity Identification activity to update the data values datasets received in input by defining an identifier value to each entity considered for the generation of the final KG. The updated version of the data values datasets, where the entities have been grouped by the previous activity, and uniquely identified in the current one, is provided as output of the Entity Identification activity.

Entity Mapping - The last activity of the ED phase is executed by the user playing the data scientist role and takes as input the data values dataset updated in the previous activity, and the ETG. This activity aims at merging into a single graph-based data object, the knowledge layer and the data layer of the final KG. Such a single object is the EG as it has been described in section 3.3.4. All the activities, in the current and previous phases, had the objective of providing the right input to this last activity. The EG produced by this activity is a KG that has been created to satisfy the initial user Purpose. Moreover, it is the fourth and last component of the EB, produced at the end of the iTelos KGC process, representing the entity values layer. This activity is executed with the support of a specific data mapping tool called Karmalinker (described below in section 4.5.3) which allows the user to link each EType's property with the attribute of the data values datasets collecting the values relative to that specific property, thus actually mapping each entity value with its relative EType's property. In addition, the data mapping tool allows the user to perform minor updates in the data values datasets fields, and values, to better map them with the properties of the ETG. Such minor updates are executed by short Python scripts applied directly to the dataset fields and values (i.e., string or numeric data modification, or the merge of two column values into a single column, within a CSV dataset). At the end of the Entity Mapping activity, each EType of the ETG is correctly associated with its entity instances. The EG, as output of the current activity is produced as Turtle [12] (.ttl) RDF file.

[12] Wikipedia Turtle

4.5.2 Entity Definition - Process

In this subsection, we report the process workflow that executes the activities described above. Such a process describes actually how the ED phase is executed, with the objective of generating the EG representing the entities of the final KG. To better describe such a process in this subsection we report the pseudocode describing the algorithm of the KD phase's process.

Algorithm 6 EntityDefinition(2, 3D, 4, 5)

1: $3D = EntityMatching(3D, 5)$
2: $3D = EntityIdentification(3D, 5)$
3: $6 = EntityMapping(3D, 5)$
4: **if** $EvaluationED(6, 2)$ is False **then**
5: $3SD = composePackage(3D, 5)$
6: $KnowledgeDefinition(2, 3SD, 4)$
7: Return null
8: **end if**
9: $MetadataDefinition(6, DI)$
10: Return 6

The algorithm 6 shows the pseudocode of the main ED phase function which takes in input the formal Purpose (2), the dataset package composed by the data values datasets (3D), the LT (4) and the ETG (5). The first statement of the algorithm aims at updating the data values datasets received in input, by executing the *EntityMatching* function. Such a function takes in input the data values datasets and the ETG, and returns in output the data value datasets where the entities have been recognized and grouped together, thanks to the matching with the ETypes modelled in the ETG. The second statement of the algorithm implements the Entity Identification activity by executing the relative function (*EntityIdentification* in the algorithm 6). Such a function follows the above described activity by taking in input the already updated data values datasets and the ETG, to produce in output a new version of the data values datasets where the entities have been enriched with identifier properties. Then, the third statement defines the main output of the ED phase, namely the EG, by executing the *EntityMapping* function taking in input the last updated version of the data values datasets and the ETG. As for the previous phase also the ED phase's algorithm continues its execution with the conditional statement which tests the result of the phase-specific evaluation activity, called *EvaluationED*. Such evaluation activity, which will be better described in section 4.6, aims at checking if the EG (object number 6) produced is suitable to satisfy the formalized Purpose (object number 2). If the test's result is negative, it means that the EG or the formalized Purpose need to be revised, thus the algorithm continues going backwards with a new execution of the *KnowledgeDefinition* function taking

as input the formal Purpose, the dataset package built again by composing the data value datasets and the ETG as schema dataset component (object 3SD built by the statement number 5, in the algorithm 6), and the LT (object number 4). Like in the previous phases, this process's comeback allows the user to refine the input and outputs of the already executed activities to achieve the expected results. If, instead, the result of the evaluation activity is positive the algorithm continues its flow with the execution of the *MetadataDefinition* activity to produce the metadata relative to the EG produced by the current iTelos KGC process phase. This activity, which at each phase is focused on the definition of the metadata, is better detailed in section 4.7. The next statement of the algorithm 6 is actually the last statement of the whole iTelos KGC process execution, which returns the EG (object number 6). The output of the *EntityDefinition* function is returned to the *KnowledgeDefinition* function, thus composed with the ETG into a single object. Then, such an object is returned as the output of the *KnowledgeDefinition* to the *LanguageDefinition* function and thus composed, into a single object, with the LT. That object is then returned to the *PurposeDefinition* function where it is additionally composed with the formal Purpose, and returned, in turn, to the main *iTelosKGC* function (algorithm 1) to be defined as the EB, being the main output of the whole iTelos KGC process. In this way, the process described by the algorithms in this phase and in the previous ones builds the EB by composing the formal user Purpose, the LT, the ETG and the EG, representing the main objective to be satisfied, the domain language of the entities to be considered, the knowledge (schema) of such entities, and the values that concretely represent such entities, respectively.

As for the previous phase, also the ED phase is compliant with the two base principles of the iTelos methodology. For this reason, the process of this phase has been designed to be executed by both the data producer and the data consumer. Even if the process execution in this phase seems to be fairly mechanical, there are key differences that result in more or less effort spent by users at this phase. Such differences reside in the data values datasets and regard how much the entities, represented in such datasets, are already grouped and uniquely identified. On the data producer side, the entity values still have to be processed by the Entity Matching and Entity Identification activities, thus the effort to be spent on those activities is a considerable part of the whole phase execution. On the data consumer side, instead, the entity values received in the input are already shaped as EGs (they have been already processed by a previous execution of the iTelos KGC process), thus the Entity Matching and Entity Identification activities are only run to check that such entities conform to the ETG in terms of entity schema and identifier properties. As a consequence, the effort to be spent by the user in this phase is focused on the Entity Mapping activity only, since the input EG needs to be concretely mapped with the (new) Purpose-

specific ETG. In other words, the cost of execution, in the current phase, is considerably reduced on the data consumer side, thanks to the diversity-aware data produced and distributed by the data producer.

4.5.3 Entity Definition - Tools & Standards

The process implementing the ED phase, described by the algorithm 6, is executed by the users playing the role of a data scientist with the collaboration of the knowledge engineer, and the domain expert user, to evaluate in the phase-specific evaluation activity. To support the process execution, the iTelos methodology provides a reference to a reference tool specific to the data mapping activity to be executed in the Entity Mapping activity. The data mapping tool, called *KarmaLinker* consists of the Karma data integration tool [66] extended with more features (i.e., Natural Language Processing on short sentences). Karmalinker is available as an open-source tool by requesting it from the Knowdive research group, of the Department of Information Engineering and Computer Science, of the University of Trento (Italy). The tool takes two main inputs:

- a reference knowledge layer resource, namely an ontological model (ontology, data schema, teleology or teleontology) concretely defined by an RDF-OWL file.
- One, or more, data layer resources, namely the data values datasets to be mapped with the knowledge layer resource, as described in the Entity Mapping activity. The tool accepts different dataset formats, as data layer input, like CSV, TSV, Excel Spreadsheet and JSON.

The *Karmalinker* tool produces two outputs. The first one is the RDF-TTL file that represents the EG where the knowledge layer input and data layer input have been merged together. The tool can also export such output as JSON-LD datasets format [13] . The second output instead is an RDF-TTL file which includes the list of all the operations that have been executed by using the the tool into a single session. The mapping operations as well as the minor dataset updates can be stored, by the user, into a sort of "EG generation recipe", called a data mapping model, which can be then downloaded as a separate output. One of the key features, provided by the Karmalinker tool, is the possibility to re-apply the same data mapping model over a new data values dataset, if and only if, such a dataset has the same data schema as the one used to record the mapping model. This functionality can be executed in a full-automatic mode, by a command line version of the tool which takes in input the new data values dataset and the data mapping model to be applied. Such automatic execution allows for a further reduction of the cost of process execution in the ED phase, in special cases, when an

[13] JSON-LD

EG has to be extended by adding new entities represented by input data values dataset with a fixed data schema but different data values. In this case, the Karmalinker tool will work automatically by re-running the same mapping operations that were specified in the data mapping model created after the first generation of the EG to be developed and updated, on the new data values.

4.5.4 Entity Definition - Example

The example started in section 4.2.4 continues in this subsection, by showing the output of the fourth phase of the iTelos KGC process, namely the EG. Moreover, at the end of this subsection we provide a graphical example of the complete EB produced following the four phases of the iTelos KGC process over the proposed example. More in details, figure 4.20 shows the EG generated by merging the ETG previously produced (see figure 4.18) and the dataset collected from the two sources considered by the example (see figures 4.10 and 4.11). It is visible how both the two entities, namely *pointRailwayStation:1* and *pointRailwayStation:2*, are structured through the same schema following the ETG structure.

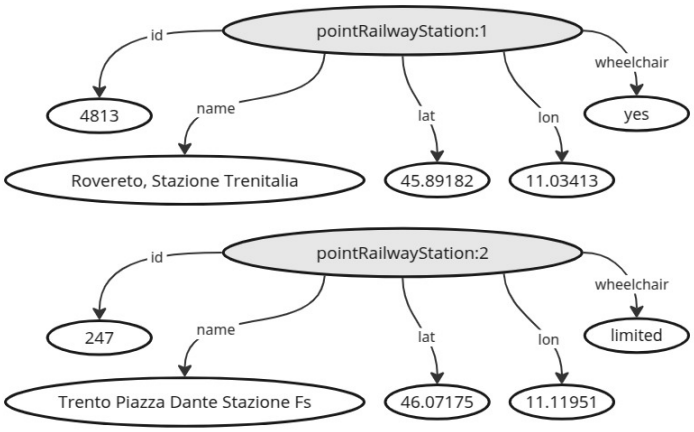


Figure 4.20: Entity Base - EG example

The EG, concretely represented by an RDF Turtle file, shows the real entities produced by the iTelos KGC process starting from the initial datasets. However, the full diversity-aware structure produced by the process can be seen by looking at the whole EB, where all information layers are linked together. A graphical representation of the EB generated for the current example is shown in figure 4.21. The figure is divided into four main sections, one for each component of the EB.

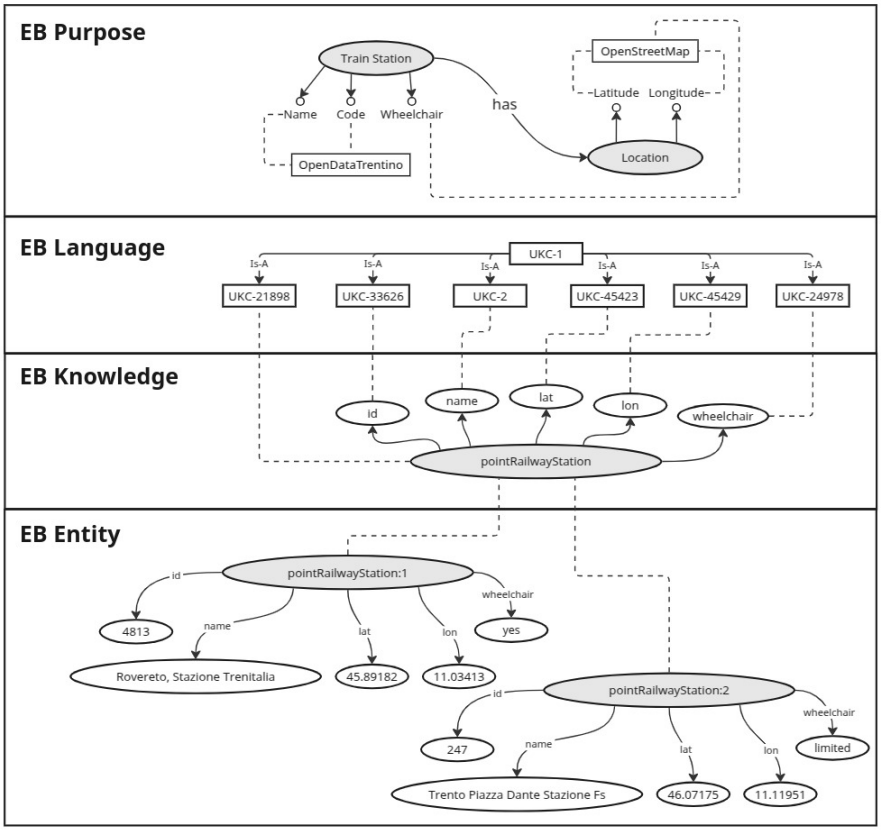


Figure 4.21: Entity Base - Complete structure example

The first section at the top, representing the EB Purpose component, shows the model of the Purpose created in the first phase of the process. Even though the Purpose component is not directly linked to the other EB components (see figure 4.21), the Purpose expresses the "user's point of view" during the process and is therefore used to drive the generation of all the other EB components, as already described in the previous sections of this chapter. The second section, namely the EB language in figure 3.2, depicts the LT containing the concepts expressed by the Purpose, aligned with the UKC hierarchical structure. Each concepts is represented using the language agnostic UKC-ID. Moreover, the "Is-A" relationships indicate how each concept is part of the UKC structure having on top the most general concepts identified by the UKC-ID 1. The third section, namely the EB knowledge in figure 4.21, depicts instead the ETG that has been created to support the initial Purpose while enhancing interoperability with existing reference standards, such as OpenStreetMap in the current example. It is important to note how the ETG Etypes and properties are linked to the concepts represented in the language layer above (dashed lines). This link shows how the

terms used to name the ETG's ETypes and properties are defined by precise UKC concepts. The last section at the bottom of the figure 4.21, represents the entity component EB by representing EG. Each entity defined in the EG is linked to the EType used to represent it. For this reason, in figure 4.21 there are links (dashed lines) between the two entities and the Etype that represents their schema at the knowledge level. The final result is the complete EB, a data structure that links the three information layers, where the diversity of information is concretely expressed at the level of language knowledge and data values.

4.6 iTelos Knowledge Graph Evaluation

The above sections reported the structure and the execution of the iTelos KGC process, by describing its phases and the activity involved. However, two important parts of the overall process behaviour still need to be described, namely the evaluation activities and the metadata definition activities. While the latter is described in the next phase (see section 4.7), the current section aims to describe how the evaluation activities, mentioned in the structures and into the algorithms of each phase, regulated the entire process flow, by assessing the quality of each phase output. As already anticipated in the description of each phase of the process, the evaluation activities check if, in a certain phase, the output produced is suitable to be forwarded to the next phase, or if it has to be refined. In the former case, the process execution continues linearly to the next phase. In the latter case, the process execution returns backwards to the previous phase, sending back what has been produced, in the last phase executed, in order to be refined. Such behaviour is depicted in figure 4.22 by the arrows coming out from the exclusive gateway at the on of each phase, called "Refine", where the positive condition arrows ("yes" tagged arrow) lead to a refinement executed by the previous phases, and the negative condition arrows ("no" tagged arrows) allow the process to continue linearly. Thanks to the evaluation activities, the iTelos KGC process allows the user to fix inconsistencies which normally appear when the data and knowledge resources are handled along the process. At each phase, the user verifies to have satisfied the requirements to proceed with the following phase, and she can return to the previous phases, until the first one (Purpose Definition) in the worst case. This means that each mistake that can be made along the whole process can be always fixed, regardless of the phase in which the user is currently working. Each evaluation activity is fundamental for the correct execution of the relative process phase. For this reason, each evaluation activity has its own inputs to be evaluated and adopts specific metrics and evaluation techniques. In the current section, we describe which are such metrics and techniques, as well as, how they are applied in the four evaluation activities included in the structure of

the iTelos KG process.

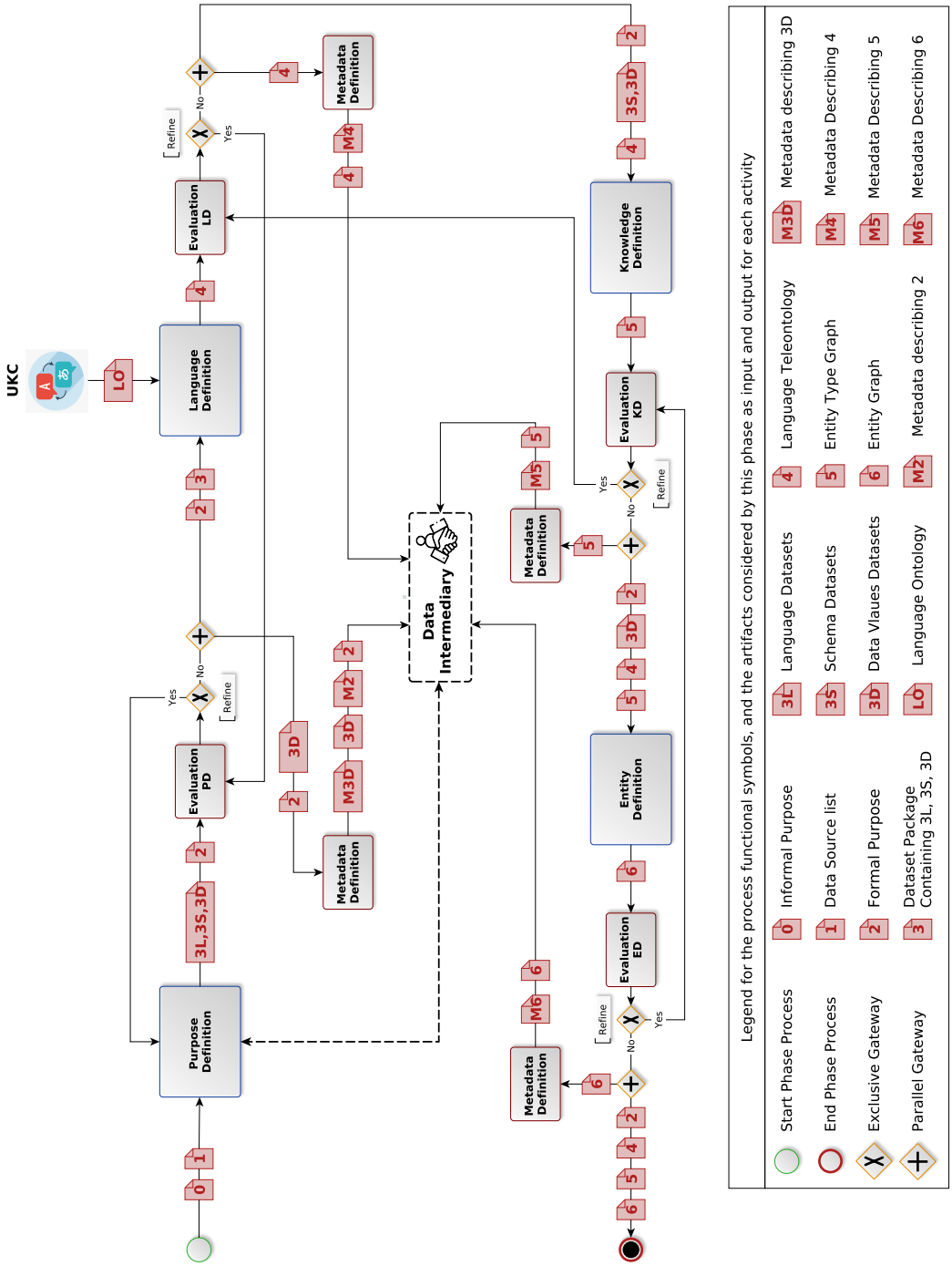


Figure 4.22: The iTelos KGC process with its evaluation activities and metadata definition activities.

4.6.1 Evaluation Metrics and Techniques

The evaluation activities included in the iTelos KGC process, are based on a set of metrics and techniques used to evaluate different aspects of the final, and intermediated, outputs of the process. For this reason, it is important first to introduce such metrics and techniques, to better understand (see next section) how they are applied to the process outputs, within the different iTelos KGC process phases.

The first distinction to be made between the evaluation metrics and techniques adopted by the iTleos KGC process is on the KG's layers. Some of the below described metrics will be applied to the KG's knowledge layer, thus evaluating how efficient the structure (or scheme) of the KG is according to specific criteria. More details about the knowledge level KG evaluation have been described in [100], while some of the metrics used to evaluate the KG knowledge layer in this work have been inspired from [41]. Other techniques, instead, aim to evaluate the efficiency of the data value layer of the KG. Here below we report the metrics and techniques adopted for the evaluation of both the knowledge and data layer of a KG produced by the iTelos KGC process.

Knowledge Layer Evaluation

Coverage - The Coverage is a metric used to evaluate the KG's knowledge layer. It is computed as the ratio (value in the range [0,1]) between the intersection of α and β and the whole α sets:

$$Cov = (\alpha \cap \beta)/\alpha = C/(A + C) \tag{4.1}$$

Where:

- α is a portion of knowledge to be satisfied. In other words, it can be any kind of schema or ontological models such as a data schema, an ontology, a teleology, or a teleontology.
- β is again a knowledge resource that is evaluated, by checking how much it is overlapped with the α.

To better understand the use of the Coverage metric, we report, in figure 4.23 the two limit cases where the metric is applied to a case where its result is zero, and another case where the metric returns a value of one.

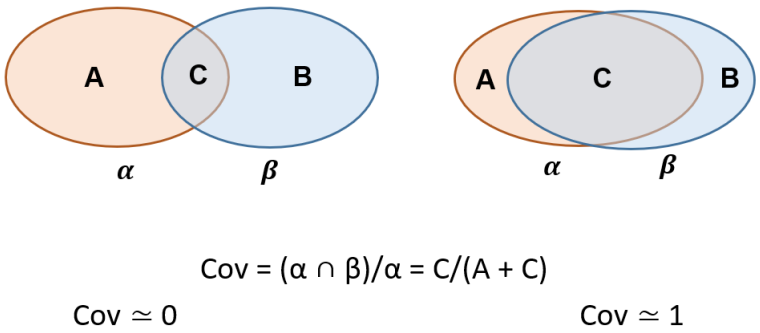


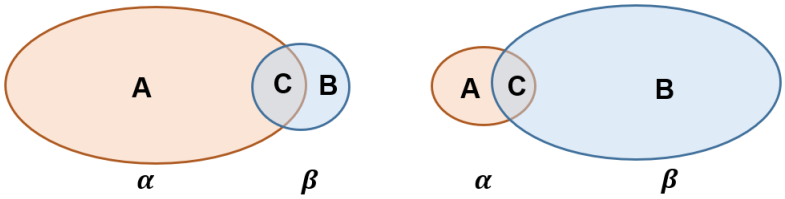
Figure 4.23: Coverage metric - limit cases

From the figure 4.23, it is possible to make some observations. The first one is that, high values of Coverage mean that the the β knowledge resource is appropriate to represent the knowledge expressed by α, in other words, the information they carry expresses (almost) the same knowledge. On the other hand instead, low values of Coverage mean two possibilities: (i) the knowledge expressed by β is not suitable to satisfy the requirements expressed by the α knowledge resources; or (ii) the knowledge expressed by α is too "vague", meaning that it is somehow unexplored and too broadly defined. For this reason, even if the β knowledge resource is well defined, it is not able to properly cover the same knowledge expressed by α.

Extensiveness - Extensiveness is a metric used to evaluate the KG's knowledge layer. It is computed as the ratio (value in the range [0,1]) of the proportional amount of knowledge provided by β with respect to the whole knowledge defined in a graph α:

$$Ext = (\beta - (\alpha \cap \beta))/((\alpha + \beta) - (\alpha \cap \beta) = B/(A + B + C) \tag{4.2}$$

To better understand the use of the Extensiveness metric, we report, in figure 4.24 the two limit cases where the metric is applied to a case where its result is zero, and another case where the metric returns a value of one.



$$Ext = (\beta - (\alpha \cap \beta))/((\alpha + \beta) - (\alpha \cap \beta)) = B/(A + B + C)$$

$Ext \simeq 0$ $Ext \simeq 1$

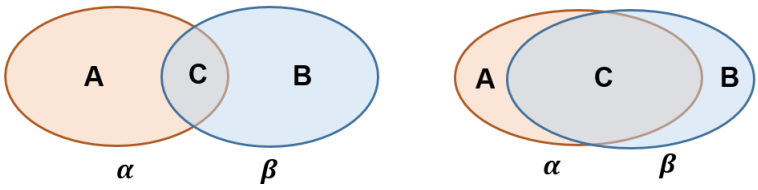
Figure 4.24: Extensiveness metric - limit cases

From the figure 4.24, it is possible to make some observations. The first one is that high values of Extensiveness mean that the contribution of the β knowledge resource is predominant with respect to the content of the knowledge into the graph α. Therefore, β brings a major contribution to the generation of the graph α. On the other hand, instead, low values of Extensiveness mean that β brings a limited contribution to α, it can be derived that the source of the knowledge provided by β is not actually aligned with the Purpose for which α has been created.

Sparsity - The Sparsity is a metric used to evaluate the KG's knowledge layer. It is the ratio (value in the range [0,1]) computed as the sum among the percentage of α not defined in β and vice-versa:

$$Spr = ((\alpha+\beta)-2(\alpha\cap\beta))/((\alpha+\beta)-(\alpha\cap\beta)) = (A+B)/(A+B+C) \quad (4.3)$$

To better understand the use of the Sparsity metric, we report, in figure 4.25 the two limit cases where the metric is applied to a case where its result is zero, and another case where the metric returns a value of one.



$$Spr = ((\alpha + \beta) - 2(\alpha \cap \beta))/((\alpha + \beta) - (\alpha \cap \beta)) = (A + B)/(A + B + C)$$

$Spr \simeq 1$ $Spr \simeq 0$

Figure 4.25: Sparsity metric - limit cases

From the figure 4.25, it is possible to make some observations. First of all, it is clear how the Sparsity is a useful metric for measuring the differences between two types of knowledge elements (e.g. data property set, for the same EType). In addition, high values of Sparsity mean that there is an important difference between the considered type of elements defined in α and the ones defined in β. Low values of Sparsity mean that there is a good match between the considered type of elements defined in α and the ones defined in β.

Data Layer Evaluation

Evaluating the KG's data layer means understanding how "dense" or "connected" is the KG, at the end of the iTelos KGC process, and during the KG's construction. It is important to note that even if the ETG (KG's knowledge layer) created by the iTelos KGC process defines the data and object properties that describe the connection between the ETypes, these properties could be associated with null or missing values, resulting in an EG (KG's data layer) that is not connected as it is its ETG. For this reason, the data layer evaluation aims to investigate the KG's connectivity, by checking the values that instantiate the ETG properties. The evaluation of the KG's connectivity improvement brought by a specific data value dataset merged into a KG, can be done in two different moments. (i) It can be performed at the end of the iTelos KGC process, thus considering the final KG. Or, (ii) it can be performed during the construction of the KG, thus by considering the order of the data value dataset that are, step by step, merged into a partially completed KG. It is important to notice that the improvement of connectivity brought by a single data value dataset, added into the KG, can be different when the dataset is added to a partial KG (during its construction), with respect to the connectivity improvement evaluated over the same dataset's values integration with the final KG. This is because during the KGC process, two (or more) datasets could carry the same real-world entities but with different values used to describe those entities. In such a case, the values of the first dataset merged into the KG, could be overwritten by the values of the second dataset merged, because the second value set was the most appropriate one. In this case, the connectivity improvement carried out by the first dataset changed after the integration of the second data value dataset. The situation just described does not happen if the connectivity evaluation is performed at the end of the process, on the final KG. However, the connectivity evaluation performed during the KG construction can provide some important insights about the impact of a specific dataset merged into the KG, thus allowing the user to understand whether or not to prioritise the integration of this dataset. More in detail, the KG's connectivity can be evaluated over two dimensions:

- *Entity connectivity*: in this case, the objective is to evaluate how much

the entities are connected to each other. In other words, the grade of connection between the different entities in the KG. To this end, the key idea is to evaluate the values of the object properties. In fact, if an object property, of a specific entity, has a null value it does not define a valid connection between the two entities connected (at the knowledge level only) by such an object property.

- *Property connectivity*: in this case, the objective is to evaluate how much the entities are connected to their properties. In other words, the grades of connection between each single entity and its properties values. To this end, the key idea is to evaluate the values of the data properties. In fact, if a data property of a specific entity, has a null value, it means that the relative entities are connected to such a property only at the knowledge level, but it misses such a link in the data layer. Potentially an Entity having properties with all null value has no connection with its properties, thus actually it does not exist.

The Entity and Property connectivity are evaluated, by the iTelos KGC process, through the *connectivity matrix*. An example of a connectivity matrix is depicted in figure 4.26.

	EType A	EType B	EType C	....	EType N
EType A	#	*	*	*	*
EType B	*	#	*	*	*
EType C	*	*	#	*	*
....	*	*	*	#	*
EType N	*	*	*	*	#

Figure 4.26: The connectivity matrix

Each cell (X, Y) in the matrix can have one of the two values described here below:

- If $(X = Y)$ the number # is the amount of non-null data property values, for all the entities represented by the EType X (or Y).
- If $(X \neq Y)$ the number * is the amount of non-null object property values, for the object properties having the EType X as domain (source EType) and the EType Y as range (destination EType).

The Entity connectivity is calculated EType by EType, as the sum of the * values for each EType row in the matrix. The resulting row sum, relative to the EType X, is then divided by the number of object properties defined for the EType X. The sum of the Entity Connectivity of all the KG's ETypes, defines the values of Entity Connectivity of the whole KG. The Property connectivity, instead, is calculated EType by EType, dividing the # values

for the cell (X, X) by the number of data properties defined for the EType X. The sum of the Property connectivity of all the KG's ETypes, defines the values of Property connectivity of the whole KG. Here below are the formal definitions of the Entity and Property connectivity.

Entity Connectivity $EC(X)$ for the EType X:

$$EC(X) = \frac{\sum_{Y=1}^{N}(X, Y)}{OP(X)} \tag{4.4}$$

Where: (X, Y) is a cell in the connectivity matrix, and $OP(X)$ is the number of object properties of the ETypes X.

Entity Connectivity $EC(KG)$ for the whole KG:

$$EC(KG) = \sum_{X=1}^{N} EC(X) \tag{4.5}$$

Property Connectivity $PC(X)$ for the EType X:

$$PC(X) = \frac{(X, X)}{DP(X)} \tag{4.6}$$

Where: (X, Y) is a cell in the connectivity matrix, and $DP(X)$ is the number of object properties of the ETypes X.
Property Connectivity $PC(KG)$ for the whole KG:

$$PC(KG) = \sum_{X=1}^{N} EC(X) \tag{4.7}$$

4.6.2 Evaluation Activities

If the first key distinction about the iTelos KG evaluation is between the data and knowledge layer, the second key distinction concerns the requirements to be assessed by exploiting such an evaluation. As already discussed in the previous chapters, the whole iTelos methodology is based on the generation and distribution of diversity-aware data. Such data is produced by considering firstly its local diversity and the Purpose for which it has to be created, the so called "primary Purpose", and secondly the reusability and interoperability of such data, namely the "secondary Purpose". To this end, the evaluation activities have been designed to assess how much the iTelos KGC process outputs are suitable to satisfy both the primary and secondary Purposes. In this section, we report how the different evaluation activities, depicted in figure 4.22, evaluate the iTelos KGC process outputs, by exploiting the metrics described in section 4.6.2.

EvaluationPD - The first evaluation activity, in the iTelos KGC process, is present in the PD phase, as depicted in figure 4.6. It is implemented by the *EvaluationPD* function in the algorithm 2 which takes in input the formal version of the user's Purpose (object number 2) and the dataset package composed by language, schema and data values datasets. As already mentioned in the PD phase algorithm description, this activity aims to check if the datasets collected in the first process phase are suitable to satisfy the formalized Purpose. To this end, the activity evaluates separately the three types of resources collected (language, schema and data values). For the language resources, the EvaluationPD activity measures the number of ETypes and properties defined in the Purpose-specific ER model, which can be named by using the terms of the concepts represented in the language datasets. The number obtained by this evaluation indicates if the language datasets collected are suitable enough to support the user's Purpose. To evaluate the suitability of the schema (knowledge) resources collected, instead, the activity adopts the KG's knowledge layer evaluation metrics described in section 4.6.1. More in detail, the Coverage metric is used to evaluate how much the schema datasets collected cover the ER model which represents the formal user's Purpose. The Extensiveness, instead, is used to evaluate the contribution given by a specific schema datasets to the final KG with respect to the user's Purpose represented by the ER model. The Sparsity is then used to evaluate the differences between each schema dataset collected and the ER model representing the Purpose to be satisfied. Thanks to this set of knowledge layer evaluation, the user can understand if the schema resources collected are enough to define later the ETG, or if more knowledge models need to be considered. About the data value datasets, the EvaluationPD activity evaluates the connectivity improvement brought by each dataset, with respect to the ETypes and properties defined in the ER model which represent the user Purpose. Such evaluation is concretely executed by using the connectivity matrix described in section 4.6.1. Thanks to the evaluation of the data value resource, the user can better understand if more datasets are needed to fulfil the original Purpose or if the collected datasets are sufficient to fulfil it. It is important to notice, that during the EvaluationPD activity, the user may acknowledge that the Purpose itself (represented by the ER model) needs to be refined, for example, because important information, to be included in the final KG, is missing in the current Purpose version.

EvaluationLD - The second evaluation activity is present at the end of the LD phase, as depicted in figure 4.12. This activity is implemented by the *EvaluationLD* function in the algorithm 3, where it takes in input the LT (object number 4) and the formalized Purpose (object number 2). As already mentioned in the LD phase algorithm description, this activity aims to check if the LT produced is suitable to satisfy the formalized Purpose. Being the LT the second component of the final EB that is produced by the iTelos KGC

process, the evaluation of the LT focuses on both the primary and secondary Purposes mentioned above. More in detail, the LT is a graph structure and for this reason, it can be evaluated by using the knowledge layer evaluation metrics described in section 4.6.1. To evaluate the LT's primary Purpose, namely the suitability of the LT itself to satisfy the user Purpose, the Coverage metric is applied to evaluate how much the LT concepts (which can be seen as KG ETypes) cover the ETypes modelled in the formal Purpose ER model. The LT's secondary Purpose, instead, refers to its reusability and interoperability level. To evaluate such key aspects of the LT, the EvaluationLD activity adopts the Extensiveness metric to understand how many concepts already defined in the UKC's LO have been reused to define LT. In other words, such activity aims to evaluate the contribution given by the UKC in the construction of the LT. The more concepts from the UKC's LO have been reused, the more standard, interoperable and reusable the LT will be. To get more information about the LT, in the current evaluation activity, the user can use also the Sparsity metric to understand the differences in the concepts used, between the LT and the portion of the UKC's LO that provides the concepts for the same domain of interest.

EvaluationKD - The third evaluation activity is present at the end of the KD phase, as depicted in figure 4.16. This activity is implemented by the *EvaluationKD* function in the algorithm 5, where it takes in input the ETG (object number 5), the formalized Purpose (object number 2) and the LT (object number 4). As already mentioned in the KD phase algorithm description, this activity aims to check if the ETG produced is suitable to satisfy the formalized Purpose. This evaluation activity aims to evaluate the knowledge layer of the final KG (EG) produced. To this end, the knowledge layer evaluation metrics described in section 4.6.1 are applied over the two activity inputs. As for the previous evaluation activity, also the current one aims to evaluate both the primary and secondary Purpose of the third component of the final EB produced by the iTelos KGC process. The ETG's primary Purpose is to satisfy the user's Purpose by modelling all the ETypes and their properties indicated in the formal Purpose ER model. To evaluate this first key aspect the Coverage metric is used to assess how much the ETypes defined in the ETG are similar (or equal) to the ETypes modelled by the formal Purpose ER model. The ETG secondary Purpose, instead, refers to the amount of ETypes, and properties for each EType, that have been reused from the previously collected schema (knowledge) datasets. As already mentioned in the KD phase description, such schema datasets are standard reference ontologies and data schema applied in the same domain of interest as the user Purpose. Therefore, the more ETypes and properties are reused from such reference standards knowledge resources during the construction of the ETG, the more standardised, interoperable and reusable the ETG will be. To concretely implement such evaluation the EvaluationKD

activity uses the Extensiveness metric the contribution of each reference standard knowledge resource to the ETG. As described in the KD phase algorithm (see algorithm 5) the ETG is constructed by integrating not only the reference standard knowledge resources but also the LT received from the LD phase. This is crucial to handle the data heterogeneity at the language level, by creating language diversity-aware data. Nevertheless, the reuse of reference standard knowledge resources can introduce ETypes and properties represented by using concepts which are not yet aligned with the UKC. This missing concept alignment is verified by the user, in the EvaluationKD activity, by using the Sparsity metric to check the differences between the ETG and the LT. If the result of this evaluation shows that some concepts need to be aligned with the UKC's LO, then, as depicted in figure 4.16, iTelos KGC process flow is directed backwards to the LD phase with the aim of bridging the gap. For this reason, the *EvaluationKD* function in the algorithm 5 takes the LT as additional input.

EvaluationED - The last evaluation activity is present at the end of the ED phase, as depicted in figure 4.19. This activity is implemented by the *EvaluationED* function in the algorithm 6, where it takes in input the EG (object number 6) and the formalized Purpose (object number 2). As already mentioned in the ED phase algorithm description, this activity aims to check if the final EG is suitable to satisfy the formalized Purpose. If the previous evaluation activities checked the suitability, and the interoperability of the language and knowledge layer of the final KG, the last evaluation activity is focused on the data value layer. For this reason the data layer evaluation metrics described in section 4.6.1 are used to evaluate the connectivity of the EG produced in the ED phase. The evaluation of the final EG allows the user to understand if the final KG produced is actually able to provide the information which satisfies the Purpose initially expressed. It is important to notice how the composition of all the evaluation activities allows the user of the iTelos KGC process to do a complete assessment of all three layers (language, knowledge and data values) of the EG produced, to verify that the local diversity, represented by the Purpose, is concretely available in the process final output.

4.7 Metadata Definition

The last section of this chapter describes the last set of activities included in the iTelos KGC process, as depicted in figure 4.22. Such activities, namely *Metadata Definition* activities, have been designed to support the distribution of the iTelos KGC process final and intermediate outputs. They are directly connected with the Data Intermediary (DI) data distribution network (LiveData), which will be detailed in chapter 5. More concretely these

activities are responsible for the creation and publication of the metadata which describes the diversity-aware data produced by the iTelos KGC process.

The distribution of the iTelos KGC outputs is fundamental to enabling the circular data economy described in section 4.1.3, where the data producer and consumer cooperate with the aim of satisfying the needs of others. The Metadata Definition activities act as a bridge between the iTelos KGC process, where the diversity-aware data is produced and the DI which aims at distributing such a resource. Nevertheless, the distribution of the resources produced by a user, by using the iTleos KGC process, must be allowed by the user itself, or better by the owner of the data created (that most of the time is the process user too). The role of the DI is to liaise with the user to find, where possible, the data distribution modality that best suits the user's interest, while strongly considering the user's privacy. More details about the mission and services offered by the DI are provided in chapter 5. To support the distribution of the resources produced by the iTelos KGC process, the Metadata Definition activities perform two key tasks. The first one is the definition of the metadata that is used to show the diversity-aware data on the DI data catalogues. The second is to concretely send the metadata, and the relative diversity-aware data, to the DI so that they can be properly visualized and distributed online. The former task, focused on the definition of metadata, is the main objective of this set of activities. In fact, the "quality" of the metadata, created for a specific resource, strongly influences the visibility, the findability and, as a consequence, the reusability of the resources itself. This is why the metadata definition is also one of the core elements of the FAIR principles (already mentioned in section 4.2). More formally, metadata is "data that provides information about other data", but not the content of the data itself, such as the text of a message or the image itself [14]. Therefore, the "quality" of metadata, reflects how much such metadata is suitable to describe and to provide information about, the relative data. As reported by the metadata definition (Wikipedia), the information carried by metadata can be focused on different aspects of the data described, thus multiple types of metadata exist. Here below we report some of the most common.

- Descriptive metadata: the descriptive information about a resource. It is used for discovery and identification. It includes elements such as title, abstract, author, and keywords.
- Structural metadata: metadata about containers of data and indicates how compound objects are put together, for example, how pages are ordered to form chapters. It describes the types, versions, relationships,

[14] Wikipedia Metadata

and other characteristics of digital materials.

- Administrative metadata: the information to help manage a resource, like resource type, permissions, and when and how it was created.
- Reference metadata: the information about the contents and quality of statistical data.
- Statistical metadata: also called process data, may describe processes that collect, process, or produce statistical data.
- Legal metadata: provides information about the creator, copyright holder, and public licensing, if provided.

The iTelos KGC process aims at covering the metadata definition for all the above mentioned categories. Moreover, it adds two very important metadata categories, namely the *provenance* metadata and the *stratified* metadata. Provenance metadata aims to describe the source of the data and how it has been handled and possibly modified over time, linking this information to the users or organisations that have handled it. The provenance metadata is fundamental for the reliability of the data, it could make the difference between verified information and untrustworthy information. The stratified metadata, instead, is a metadata category very specific for the diversity-aware data produced by the iTelos KGC process. More precisely, it provides information about how the different layers of diversity are linked together between the different components of an EB produced by the iTelos KGC process. Since an EB is a composition of four elements, namely the Purpose, the LT (language diversity layer), the ETG (knowledge diversity layer) and the EG (data values diversity layer), when an EB needs to be distributed, it appears as four different resources (separate files) that are displayed individually in the DI data catalogues. The stratified metadata is used to show, to the catalogue users, the relationship between the four resources published. More in detail, the stratified metadata defined for one of the four EB components, indicates which are the resources that have been produced together in the same iTelos KGC process execution, relative the the other diversity layers. For example, the stratified metadata for specific LT indicates which is the Purpose for which such LT has been created, as well as the relative ETG and EG. Thanks to the definition of this type of metadata, the DI catalogue users can navigate among the different resources published being aware of the diversity of the information published, at different levels (Purpose-specific, language, knowledge and data values layer).

In the iTelos KGC process depicted in figure 4.22 four Metadata Definition activities are visible at the end of each phase execution. The activities behaviours, in this case, are similar for all four process phases, what does change is the input and the output, since each phase-specific implementation

of the Metadata Definition activity is focused on the generation and distribution of metadata about the relative phase-specific output. Here below we report a description of the Metadata Definition activity as it is executed in each phase of the iTelos KGC process.

PD Metadata Definition - The first execution of the Metadata Definition activity is present in the PD phase as depicted in figure 4.6 and in the algorithm 2. The activity, in this phase, takes in input the formalized Purpose (object number 2 in figure 4.22) and the data package (object number 3) containing the standardized datasets for the three different types of data (language, knowledge and data values). The Metadata Definition activity defines the metadata for all the resources produced in output by the PD phase, represented in figure 4.22 by the objects M2 and M3D, respectively. Then, the activity provides to the iTelos KGC process user, access to the DI data distribution network, so that she can decide to transfer the metadata created, to the DI for the publication on the DI catalogues. The DI, in agreement with the data owner, could request also access to the Purpose and the data package produced, to enable their distribution.

LD Metadata Definition - The second execution of the Metadata Definition activity is present in the LD phase as depicted in figure 4.12 and in the algorithm 3. The activity, in this phase, takes in input the LT, produced in the LD phase, and defines the metadata for the LT represented in figure 4.22 by the objects M4. Then, as in the previous execution of the Metadata Definition activity, it provides access to the DI data distribution network, so that the process user can decide to transfer the metadata created, to the DI for the publication on the DI catalogues. The DI, in agreement with the data owner, could request also access to the Purpose and the data package produced, to enable their distribution.

KD Metadata Definition - The third execution of the Metadata Definition activity is present in the KD phase as depicted in figure 4.16 and in the algorithm 5. The activity, in this phase, takes in input the ETG, produced in the KD phase, and defines the metadata for the ETG represented in figure 4.22 by the objects M5. Again, access to the DI data distribution network is provided, to allow the user to distribute the metadata produced in this phase, and eventually (based on the agreement between the data owner and the DI) the knowledge resource generated by the execution of the KD phase.

ED Metadata Definition - The last execution of the Metadata Definition activity is present in the ED phase as depicted in figure 4.19 and in the algorithm 6. The activity, in this phase, takes in input the EG, produced in the ED phase, and defines the metadata for it, as represented in figure 4.22 by the objects M6. Similar to previous executions, also in the last one, the

activity provides access to the DI data distribution network for the publication of the metadata produced in this phase, and the distribution of the EG created, if allowed by the data owner.

It is important to notice how the execution of the Metadata Definition activity, within each phase of the iTelos KGC process, is regulated by the interest and the privacy protection of the user as owner of the data to be distributed. The data owner can also decide at any process phase not to publish all or part of the metadata produced. As a consequence, the diversity-aware data distributed may also be published in full, in part or not at all. More details about the data distribution possibilities offered by the DI to the data owner (and iTelos KGC process user) are described in chapter 5. To conclude we indicate the possibility of visualizing more in detail the metadata defined for each specific resource published, in the appendix B.

5

The iTelos Data Intermediary Process

Contents

This chapter aims to describe the third of the three processes composing the iTelos methodology, namely the iTelos data intermediary process. If the data representation process and the KGC process, described in chapters 3 and 4, have the objective of representing and integrating the diversity-aware data handled by the iTelos methodology, this last process encapsulates the two processes defined above, with the objective of defining a clear and efficient data reuse workflow. Such a workflow is based on four main phases, namely data retrieval, data production, data integration and data distribution. The four phases of the process are detailed in section 5.5. The iTelos data intermediary process is centred on the role of the iTelos Data Intermediary (DI), as it is described below in this chapter, with the objective of enabling the circular data economy initially considered with the iTelos KGC process (see section 4.1.3). The key idea is to enable the reuse of the data represented by the iTelos data representation process and built by using the iTelos KGC process, within a process for data retrieval, production, integration and data distribution. The iTelos DI, as a provider of the process, plays a key role in enhancing the reuse of the diversity-aware data produced by the iTelos methodology, thus enabling the cost reduction of data generation and integration that is only possible by considering the reuse and composition of data already produced by the iTelos KGC process itself. For this reason, a dedicated infrastructure has been defined to support the role of the iTelos DI in the work that she has been asked to do. This chapter is organized as follows. Section 5.1 defines the iTelos DI role with its objectives and responsibilities within the iTelos methodology. Section 5.2, instead, describes the infrastructure which is used by the iTelos DI to enable the reuse

of data. The section 5.3.2 concludes the chapter by defining the iTelos data intermediary process, as well as how it is used by the iTelos DI thanks to data intermediary infrastructure.

5.1 The iTelos data intermediary

As already mentioned above, the iTelos DI plays a key role in the execution, but also in the definition of the last iTelos methodology process. The DI, as defined by the iTelos methodology, is a human, or an organizational, actor technically supported by infrastructure and a human-driven process. While the next two sections are focused on the description of the infrastructural and processual components, this section aims, instead, at describing the characteristics of the human iTelos DI actor.

The iTelos DI actor is a third-party role positioned between the data producer and data consumer roles, described in section 4.1.3. It has the objective to "fill the gap" between the two roles, by creating connections between the consumer, which has the Purpose of being satisfied by diversity-aware data (data-based applications and services to be implemented into a specific context), and the data producer having the expertise and the skills to actually generate the required data. As discussed in chapter 1, it is often difficult to connect the two roles which operate usually in separate and quite different work contexts. An example of such a situation is the public administration that is interested in digital solutions to improve the quality of the services given to their citizens. However, people working in public administration are usually non-technical actors, so they have a clear understanding of the problems to be solved (a Purpose) but lack the expertise to implement a solution properly. On the other side, the data producer is a technical person with digital skills having, however, difficulties in understanding the domain or context-specific problem to be solved, due to the lack of expertise in such a domain or context. This situation limits considerably the work of both the two actors, as well as the possibility of solving problems with data-based applications and services. The iTelos DI aims at linking the data consumer's Purpose with the data and technical expertise required for that Purpose satisfaction, provided by the data producer. To this end, the interests of iTelos DI do not include any objectives other than those related to the cooperation between data producers and data consumers. For this reason, the iTelos DI has to be a non-profit third party, so as not to introduce its own interests into the cooperation between the data producer and the data consumer, with the ultimate goal of satisfying the consumer's Purpose and valuing the data producer's work, under appropriate compensation. Thanks to the coordination provided by the iTelos DI, both the data producer and the data consumer are induced to cooperate, the former seeing an increase in demand for its

work, while the latter finds a way to fulfil its Purpose.

The DI intermediary as a coordinator, is the view of the iTelos methodology regarding this role already recognized for its importance by the European Commission, as described in chapter 2. The key intuition is that the cooperation between data producer and data consumer, cannot take place without a non-profit third party. The iTelos methodology recognizes such cooperation as one of the key pillars, not only for enhancing the reuse of data, thus reducing the cost of data, applications and services production, but also to preserve the information diversity represented by the data supporting such applications and services. In the era of AI, the iTelos DI has the fundamental responsibility of protecting such diversity, by allowing and ensuring the generation, distribution and protection of diversity-aware data. With diversity protection, in such a context we mean the ability to preserve the geographical, cultural and individual information that characterises a local context, as well as the individuals living in it. This approach contrasts with the more common centralised approach in AI systems, where a single data producer collects the data from different sources and pre-processes such data with the aim of integrating and modelling it with its own representational point of view, thus losing some of the diversity that such data carries, or sometimes distorting the reality that such data represents. The iTelos DI is designed to enable data production, pre-processing and integration by local diversity experts only, who are then connected to users interested in using such data to support AI-based systems. In other words, the iTelos DI provides both data producers and data consumers with the processes (iTelos Data Representation and KGC processes) and tools that lead to the creation of interoperable and reusable diversity-aware data, thus avoiding a single central interpretation of information and instead supporting a distributed and local sharing of information diversity for a more realistic and fair use of AI-based systems.

5.2 The iTelos data intermediary infrastructure

The iTelos DI role, as it has been described in the previous section, acts as a dedicated infrastructure designed by considering the principles the iTelos DI aims to support. Such an infrastructure is shaped as a worldwide data distribution network called *LiveData*, composed of different local nodes connected and interacting together. The LiveData data distribution network has been designed to support the creation, integration and distribution of the diversity-aware data handled by the iTelos methodology. The key idea behind the LiveData data distribution network architecture is the governance of the data, namely who is responsible for the creation, management and distribution of the local diversity-aware data. In line with the iTelos DI principles, described in section 5.1, the LiveData network is based on the

important constraint that ownership of diversity-aware data must remain in the hands of the people who have the right knowledge to handle it. In other words, the data that carries information about a specific domain of interest and is related to a specific context has to be defined and handled by people experts in such a domain, and living in (more in general, interacting with) such a context. In other words, the domain experts, already defined as a key role in the iTelos KGC process (see chapter 4), are the only people who can really express, through the data, the information diversity which can be then exploited by data-based applications and services, into their relative context and domain of interest. The domain experts have the responsibility to define the data, and to declare how the data can be used and distributed. This domain expert-centric (or "diversity-centric") approach is implemented into the LiveData data distribution network, by adopting the data mesh architectural approach [71, 23]. The choice to adopt the data mesh model to design the LiveData network comes from the four key principles of data meshes which are perfectly in line with the iTelos DI objective. As defined by [71], such principles are based on (i) the decentralization of the data ownership and management, to allow data experts to handle each data product in the best way; (ii) the maintenance of a self-served platform for the development of data products and the maintenance of the overall data mesh; (iii) the quality and security assurance of the data products in the data mesh; and (iv) the interoperability of the data product and the technologies adopted to handle it. The LiveData data distribution network aims to implement the four key principles of the data mesh model, thanks to a specific architecture adopted by each node of the network, in which the iTelos KGC process, with its tool framework, is integrated.

The data mesh implemented in the LiveData network is designed as a composition of local nodes, each of those defined for a specific *data ecosystem*. The ecosystem term has been extracted from the description of Open Data Ecosystem (ODE) provided in the work by Shaharudin et al [92]. In such an inspiring work the authors report "the ecosystem evokes an image of the well-being of the entity and, on the other hand, fulfilling one's own needs through the richness and vitality of the ecosystem" (Poikola et al., 2011,p. 14 [86]). They also highlighted the ever-changing nature of an ecosystem: "With the ecosystem, we wish to highlight not only the technological systems and institutionalized organisations but also the living, dynamically changing network of interaction" (Poikola et al., 2011, p. 14 [86]). This hybridity of self-interested yet interdependent actors is an essential notion of Actor-Network Theory (ANT) asserting that "every stable social arrangement is simultaneously a point (an individual) and a network (a collective)" (Callon and Law 1997, p. 165 [16]). We adopted, in this work, such a description of the ecosystem to define a LiveData network's node as a LiveData Ecosystem (LDE), meaning a geographical region (like a country, a province or a city),

an organization or a company, as well as a single person who wants to handle her own data, participating in the whole LiveData network, thus being part of a community of data users. More concretely, an LDE is defined as a context in which a domain expert is responsible for satisfying the need to produce and consume data to support data-based applications and services for the community of people living in such a context, of which the domain expert itself is a part. The key principle behind this design choice is that only the local nodes of the data distribution network should be the owners of, and therefore responsible for, the diversity of information carried by the data produced in their relative context. For this reason, only a domain expert (one or more relative to a single context) can be an administrator of a node of the LiveData data distribution network, as well as the owner of data distributed through it. This decentralised approach was designed to improve the quality of data distribution and to avoid the distortion of information diversity that occurs after reinterpretation of the data (and metadata) distributed by a more centralised solution. Once again, as with the distribution of data, the individual's Purpose is crucial in defining the diversity and knowledge for a specific context. However, such knowledge, as well as the individual who creates it, will not grow without sharing it with the community, thus allowing the other community members to benefit from it and, consequently, to benefit from the knowledge shared by the other community members. In parallel to this local data governance model, to enable the different nodes of the network to share data and knowledge, a central agent is required to establish and coordinate the interactions between the local domain experts, namely the iTelos DI. To summarize, the governance model applied to the LiveData data distribution network has to be both local and global. The local side ensures that the ownership of data remains in the hands of the people having the required knowledge to handle that data, thus enhancing the data quality level. The global side ensures the connections and interaction between the local domain experts, with the objective of enhancing the data sharing, thus increasing the value and the usability of the data produced, and distributed.

The lack of quality in data distribution has been investigated by an analysis conducted over existing data distribution portals (like data catalogues). What has been discovered is that the available data distribution web portals often provide low-quality data, in terms of missing, or "noisy", information carried by the distributed datasets. Another issue is the quality of the metadata used to describe the data distributed. Most of the time the information provided by the metadata is not sufficient to provide to the users (interested in reusing the published data) the appropriate level of understanding to decide whether the data is what they are looking for or not. Nowadays, many different data distribution web portals are created to distribute data about many different domains of interest, which in turn needs to be contextualized

in a geographical and social context. This heterogeneous situation, in which the same information can be represented and/or interpreted differently depending on the point of view of the data producer, increases the complexity of reusing the data distributed by such web portals. The meaning of the data (dataset attributes, dataset structures and values) and the relative metadata is, usually, understandable only for the data producer who generated and described the data distributed, thus remaining not discoverable for the data consumers interested in reusing the data.

To overcome the above mentioned limitation related to data distribution portals, the key idea underlying the LiveData data distribution network is to support local domain experts in the creation and distribution of local diversity-aware data. The goal is to concretely represent such a diversity in the data distributed, and at the same time guarantee the interoperability of such data with other diversity-aware data created, and distributed by data producers operating in different contexts (different LDE). For this reason, each node of the LiveData network distributes the diversity-aware data produced following the EB data model, defined in chapter 3, and integrated into KGs using the iTelos KGC process (see chapter 4). Moreover, each LiveData network node is responsible for the quality of the metadata used to describe the data it distributes. The information provided by metadata should be able to inform users looking for data for re-use about the diversity that such data contains. The following sections aim to detail the architecture of a LiveData node, and the architecture of the LiveData network, and conclude with the description of the process adopted by the iTelos methodology to distribute and reuse diversity-aware data.

5.3 The LiveData node architecture

This section defines the architecture of a single LiveData node, by describing its components and how they are used by the iTelos DI to support diversity-aware data reuse. More in detail a LiveData node is defined by the components depicted in figure 5.1. As the figure depicts, there are four main components that compose the LiveData node architecture, namely, the *Data Repository*, the *KGC Process*, the *Services*, and the *Data Catalogue*. In the rest of this subsection, a detailed description of each subsection is reported.

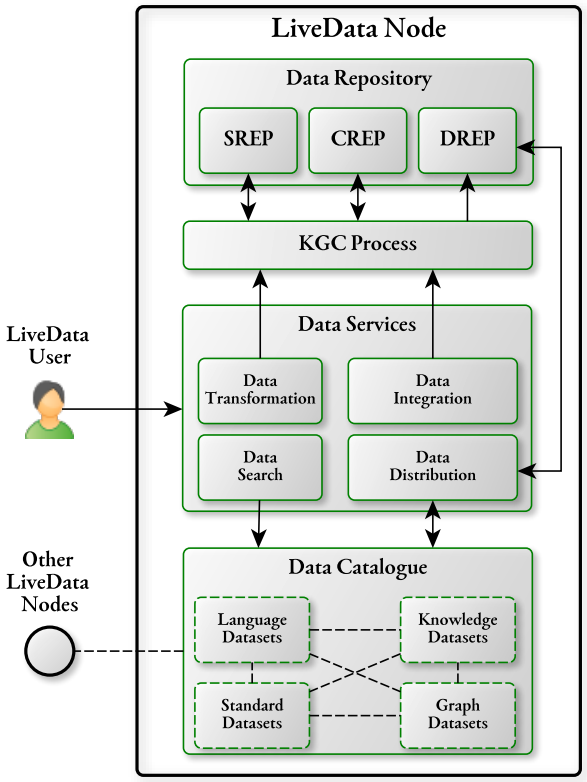


Figure 5.1: The LiveData Node structure.

5.3.1 Data Repositories

The data repository is the architectural component used for the storage, and maintenance of the LiveData contents, namely the diversity-aware KG data produced by the iTelos KGC process. The LiveData repository can be implemented by using several data management technologies ensuring data storage, versioning, as well as data security and eventually data anonymization (in the case of sensible data). Nevertheless, the repository's logical structure is uniquely defined and it has to be observed regardless of the technology used to implement it. Such a structure takes into account the four main types of diversity-aware data handled by iTelos methodology (Purpose, Language, Knowledge and Data values), as well as the different phases of data reuse, from the data retrieval to the data distribution through the data transformation and integration. In addition, the LiveData repository aims to support the various phases of the iTelos KGC process, as well as to properly maintain the diversity-aware data in accordance with the EB data model in order to enable access to such data. To this end, the LiveData repository, for each node, is internally structured in three main logical

partitions, as depicted in Figure 5.1.

- *Source Repository (SREP)*: this repository partition contains low-quality data (non-diversity-aware) collected, by the data producer, from external sources that need to be transformed into diversity-aware data. The data stored in the source repository has a lower quality and interoperability level. However, following the LiveData approach, SREP is again divided into three sub-sections for the storage of low-quality data values datasets, external language datasets (not generated by the iTelos process) and external reference ontologies or data schemas. The SREP is the repository partition from which the iTelos KGC process receives the input data to be transformed into diversity-aware KGs.

- *Core Repository (CREP)*: this repository partition instead contains the diversity-aware KG data that has been transformed by the iTelos KGC process (see chapter 4). This repository partition is again internally organised to maintain all the types of data produced by the iTelos KGC process: (i) the cleaned and formatted data values datasets; (ii) the formal Purpose data; (iii) the language datasets (Language Teleontology); (iv) the knowledge datasets (Purpose Teleology, Knowledge Teleontology and ETG); and (v) the final Entity Graph. The CREP is the repository partition used to store the output of the iTelos KGC process, as well as the partition from which the diversity-aware data are retrieved for reuse Purposes.

- *Distribution Repository (DREP)*: the last repository partition contains the diversity-aware data ready for distribution, together with the metadata used to describe such data. When the LiveData administrator wants to distribute data stored in CREP, a copy of that data is stored in DREP. In other words, DREP contains a subset of the data stored in CREP. In addition, for each dataset copied into DREP, a dedicated set of metadata is created and stored in the same repository partition. It is important to note that in Figure 5.1, DREP is directly connected to the LiveData distribution service, which is responsible for publishing the metadata on the LiveData catalogues.

Internally, each partition of the LiveData repository is in turn structured to maintain the separation between the different types of diversity-aware content. A more detailed definition of the internal structure of each LiveData repository partition is provided, here below; together with the data directory subtrees for each repository partition depicted in figures 5.2, 5.3 and 5.4.

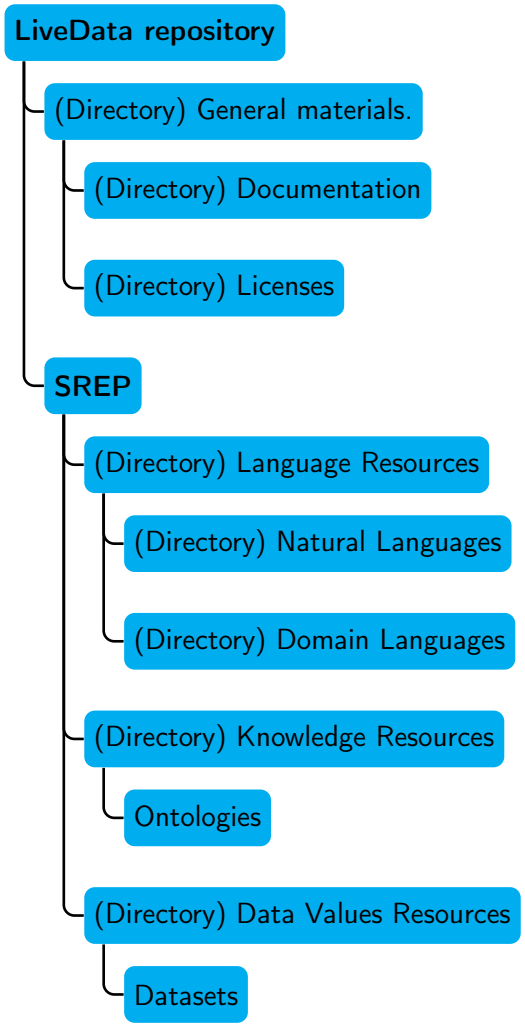


Figure 5.2: LiveData repository - SREP partition

Figure 5.2 depicts the subtree representing the structure of the LiveData repository for the first main partition, namely SREP, and for another top-level partition, called *General materials*. As general materials resources, we mean those artefacts, like documentation, reports or license documents, which need to be associated with the data to guarantee its proper distribution and understanding. The SREP partition, instead, is internally structured with three main directories, namely *Language Resources*, *Knowledge Resources* and *Data Values Resources*. The language resource directory is, in turn, split into two subdirectories called *Natural Languages* and *Domain Languages*. This division inside the language resource directory has been thought to distinguish between the data, collected from external data sources, representing natural languages (i.e., Italian, English, Spanish vocabularies) and domain languages adopted for specific domains of interest. For example,

in the healthcare domain, there are different domain vocabularies defining coding values for drugs and diseases (i.e., LOINC [1], SNOMED [2]). The directories that contain the knowledge and data value resources, on the other hand, do not take into account any internal subdivision. However, the LiveData administrator is free to act within these directories to add any required internal subdivisions.

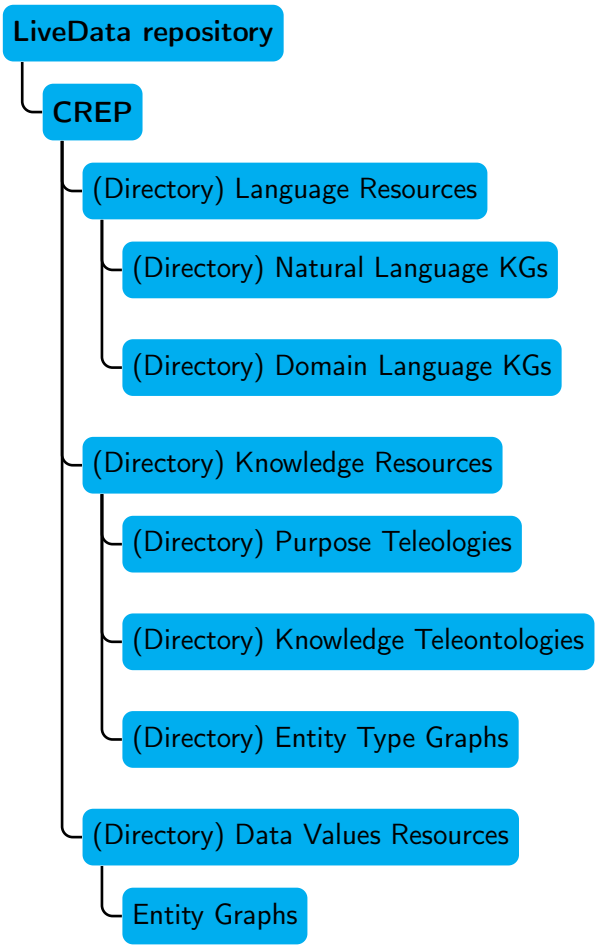


Figure 5.3: LiveData repository - CREP partition

Figure 5.3 depicts the subtree representing the structure of the LiveData repository for the second main partition, namely CREP. The CREP partition is, again, internally structured with three main directories, namely *Language Resources*, *Knowledge Resources* and *Data Values Resources*. For the same reason already discussed describing the internal structure of the SREP partition, the language resource directory, also in this partition, is di-

[1] https://loinc.org/
[2] https://www.snomed.org/

vided into two subdirectories, one for the natural language KGs (Language Teleontologies) and the other for domain language KGs. The knowledge resources directory, instead, has to store the three types of knowledge models handled by the iTelos KGC process, namely Purpose Teleologies, Knowledge Teleontologies and Entity Type Graphs. For this reason, this directory is internally divided into three subdirectories, one for each type of knowledge model to be stored. In the end, the data values resources directory contains the EGs produced by using the iTelos KGC process.

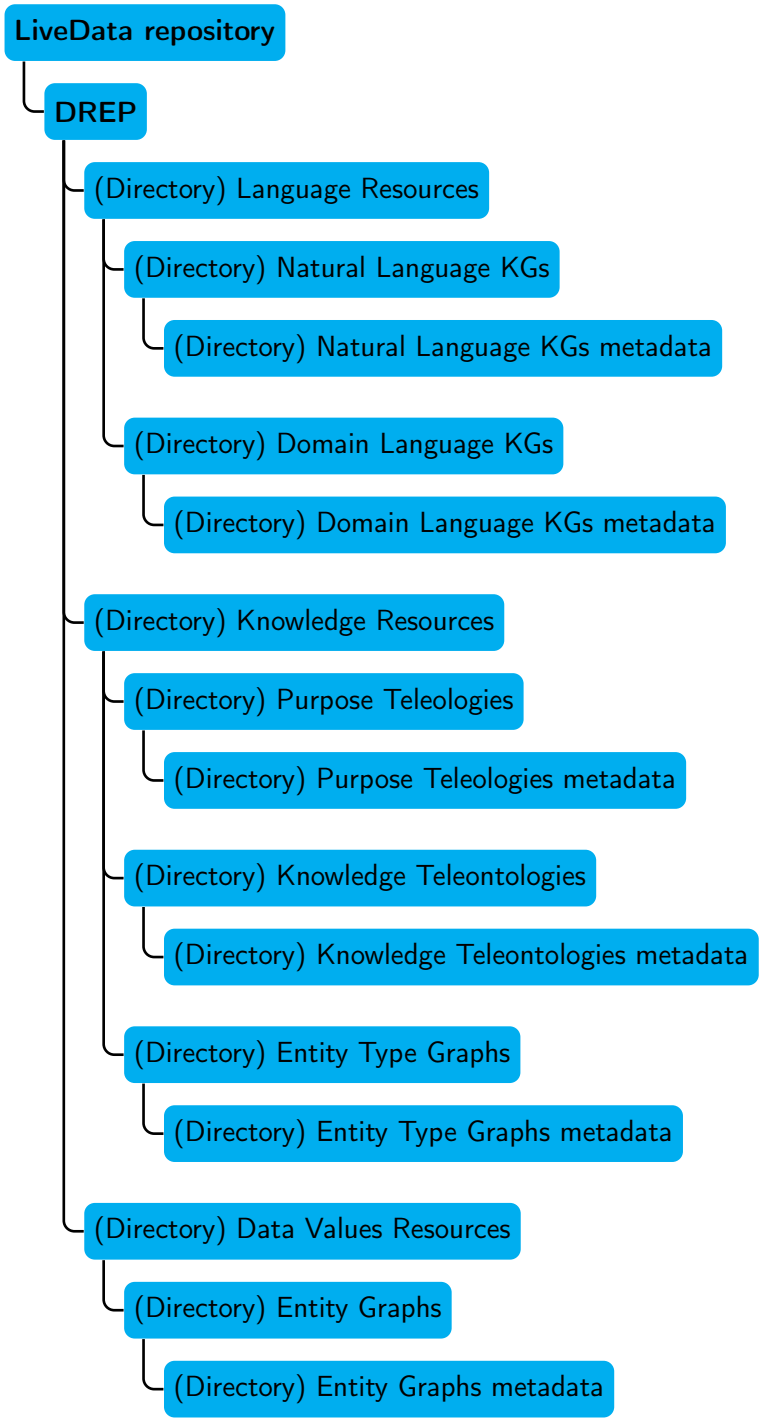


Figure 5.4: LiveData repository - DREP partition

Figure 5.4 depicts the subtree representing the structure of the LiveData repository for the third main partition, namely DREP. The peculiarity of

this partition is that it maintains not only the data to be distributed but also the relative metadata used to describe these resources, making them understandable and explorable by users interested in their reuse. For this reason, the internal structure of the DREP LiveData partition follows the same structure as the CREP one, with the addition of a dedicated metadata subdirectory for each type of language, knowledge and data values resource. Each metadata subdirectory contains the metadata describing the resources stored at the same level, in the subtree of the metadata subdirectory itself.

5.3.2 KGC Process

The second main component of a LiveData node architecture is the KGC process. This component is concretely defined by the process and the tools framework implementing the iTelos KGC process (see chapter 4). The iTelos KGC process, included in the infrastructure provided by the iTelos DI, allows the LiveData users to concretely create, and integrate together, the diversity-aware KG data following, as a consequence, the iTelos data representation process, thus be compliant with the EB data model. In other words, this LiveData component links together the three iTelos methodology processes, thus supporting the iTelos methodology users in the whole process of data reuse, by providing concrete support for data retrieval, data representation, data integration as well as for data distribution thanks to link with the other components of the LiveData node, more focused on the distribution of diversity-aware data. As depicted in figure 5.1, the KGC process component, in the LiveData node, is directly connected to the data repository component. This connection concretely implements the interactions between the iTelos KGC process and the iTelos DI depicted in figure 4.22. More in detail, the KGC process component is connected to the SREP repository partition to handle the storage of the non-diversity-aware data collected from external data sources (not included in the LiveData network), during the first phase of the iTelos KGC process (see section 4.2). The LiveData KGC process component is then linked to the CREP repository partition, to store and reuse the diversity-aware data produced along the different phases of the iTelos KGC process. Moreover, the connection between the LiveData KGC process component and the DREP repository partition allows the storage of the metadata, for each diversity-aware data, generated by the different phases of the iTelos KGC process. As shown in figure 5.1, the LiveData KGC process component is also associated with two services, the *Data Collection* and *Data Transformation* services (detailed below in section 5.3.4), which use the iTelos KGC process and its tool framework to retrieve data already stored in the LiveData repository and to transform (non-diversity-aware) or integrate (diversity-aware) such data, respectively.

5.3.3 Data Catalogue

The data catalogue is the most important component of the LiveData node architecture. Depicted at the bottom in figure 5.1, it is responsible for publishing diversity-aware data to be distributed in the LiveData network, through the visualization of its relative metadata. More concretely a LiveData catalogue is a web portal designed to make the diversity-aware data searchable, discoverable and understandable. The current section puts a special focus on the user interface of each catalogue's webpage. The design of the graphic user interface (GUI) of a LiveData catalogue, is described by reporting the graphical aspects which are in common to each LiveData catalogue of the LiveData network. Moreover, the description of the catalogue web pages goes in parallel with the definition of the metadata used to illustrate different kinds of information related to the diversity-aware data distributed. This section shows the UI mock-ups including the features that need to be considered to produce a complete first version of the LiveData catalogue architecture. However, the current version of the LiveData data distribution infrastructure contains a beta version of such catalogues, which does not yet implement all of the features described in this section. The current version of the LiveData catalogues [3] is undergoing a gradual update process. Moreover, more details about the setup and development of the LiveData node catalogue are provided in appendix A.

The efficiency of the data distributed is definitely a matter of "quality" of the data itself. By considering the data, such a quality can be maintained by following the criteria defined by the 5-star Linked Open Data, as well as the FAIR principles to enhance the interoperability and reusability of the data. Nevertheless, high-quality data is useless if it is not searchable, or if it is not understood by the data users who can potentially exploit it. In other words, the efficiency of data regards also the metadata used to define it. The information provided by metadata describing the data distributed (into many existing data distribution catalogues) is usually defined as a set of informative fields (key-value pairs) that users can read on data distribution catalogues or other web portals. The metadata set can be richer or poorer depending on the number of metadata fields provided for each data item. However, the quality of metadata is not only a matter of large or small metadata sets. Most of the time it is difficult to understand the information provided by the metadata because the data producer, who also defines the metadata, omits the context (the background environment) in which the data exists by taking it for granted. Such a context is where the data can have a stronger impact. For example, data about the public transportation service can be better understood by knowing in which city, or area, the data has been collected and in which period of time. Information regarding the

[3] LiveData main catalogue

context of the data can be provided at the general level by describing why a catalogue has been created, at the dataset level with the dataset's specific metadata, but also at a more specific level by considering metadata describing portions of data into the single datasets. In this section are reported the details about the LiveData catalogue design and about the LiveData catalogue metadata definition, with the objective to make the LiveData data discoverable not only readable.

The LiveData catalogue is a web portal with a graphical user interface (GUI) made up of different web pages, with the aim of providing the users of the catalogue with different levels of information depending on the web page that the user is visualising. The users who want to navigate a LiveData catalogue need to understand the definitions of "resource" and "dataset" as they are the main elements of information collected and distributed in the catalogue.

- A *dataset* is a single file containing data. A LiveData catalogue collects, manages and distributes datasets according to the iTelos data distribution process (see section 5.5).
- A *resource* is a collection of datasets related to the same Purpose, intended as the Purpose that guides the execution of the iTelos KGC process, as well as one of the four components of an EB, according to the EB data model. For example, a resource can be "the information about the clients of the university canteen", named "University canteen clients", thus including different datasets for the different types of users (students, professors and university staff).

Each LiveData catalogue is divided into three main web page layers which consider the above definition of dataset and resource. The three web page layers are defined as: the catalogue landing web page, the catalogue search web page and the catalogue resource-specific web page. All three web pages are detailed in the following paragraphs.

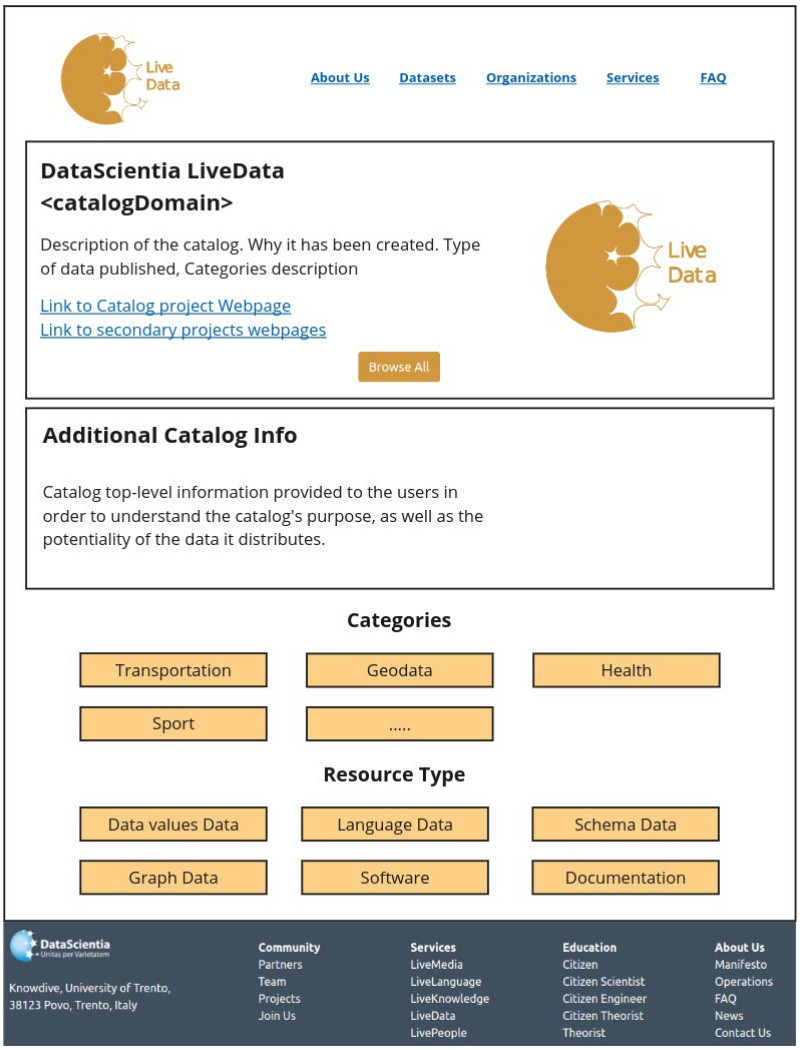


Figure 5.5: The LiveData catalogue landing page.

The LiveData catalogue landing web page

This is the first web page that users interact with while navigating a LiveData catalogue. The catalogue landing page, depicted in figure 5.5, shows first of all the catalogue name and logo, plus a short description of the catalogue itself. Such a description is crucial to let the users understand why the catalogue has been created and to collect and distribute data about which geographical, organizational or social domain of interest. Thanks to this first catalogue introduction the users are able to understand if the catalogue is suitable (contains the right resources) to satisfy their own Purpose.

As shown in figure 5.5, the initial catalogue description can contain links to other web pages describing projects, initiatives, organizations and other information connected to the catalogue itself or provide more details about why the catalogue exists. After the initial catalogue description, the "Browse All" button allows the users to jump into the catalogue thus accessing its resources. Below the initial catalogue description, as visible in figure 5.5, one or more additional sections can be added to the catalogue landing page to provide more detailed information to the users. More in detail these sections are used to specify what is defined as "catalogue level metadata", defined as follows.

- *Catalogue level metadata*: This metadata set provides information regarding the data distribution portal (catalogue or other web portal), by describing the reasons for its construction, and which types and categories (data domains) of data it is responsible for providing. Moreover, as part of this metadata, it is possible to find the name of the organization(s) and the administrator(s) responsible for maintaining the data distribution portal. This information can be provided as a set of metadata written into a delicate catalogue's web page (or in the catalogue landing page like in the LiveData catalogue case), where the users can read all the necessary information to better understand which data they can find in the catalogue itself.

The lower part of the LiveData catalogue landing page offers the users other options to access the catalogue resources. In fact, as visible in figure 5.5, under the catalogue level metadata section, the landing page offers to the users two different sets of buttons:

- *Categories*: Each button in this group represents a category of resources collected and distributed by the LiveData catalogue. Some examples of categories are: transportation, tourism, healthcare, geodata, sport, and others. The buttons in this group act as filter functionalities allowing the user to jump into the catalogue (more precisely into the second web page layer) where only the resources of the category selected are shown.

- *Resource Type*: Similar to the previous group of buttons, in this group each button represents a possible type of resource available in the catalogue itself. In figure 5.5 it is possible to see the "Data values Data" button, the "Language Data" button, the "Schema Data" button and the "Graph Data" button. Those buttons allow access to the data produced by the iTelos KGC process, namely, the data values datasets cleaned and formatted, the language datasets, the knowledge datasets, and the Entity Graphs (EGs), respectively. Additionally, the button "Software" is present to access software resources, like code libraries

developed specifically for the data published in the catalogue. The "Documentation" button, instead, allows the user to access documentation resources that the users can analyse to better understand the other (types of) resources available in the catalogue itself.

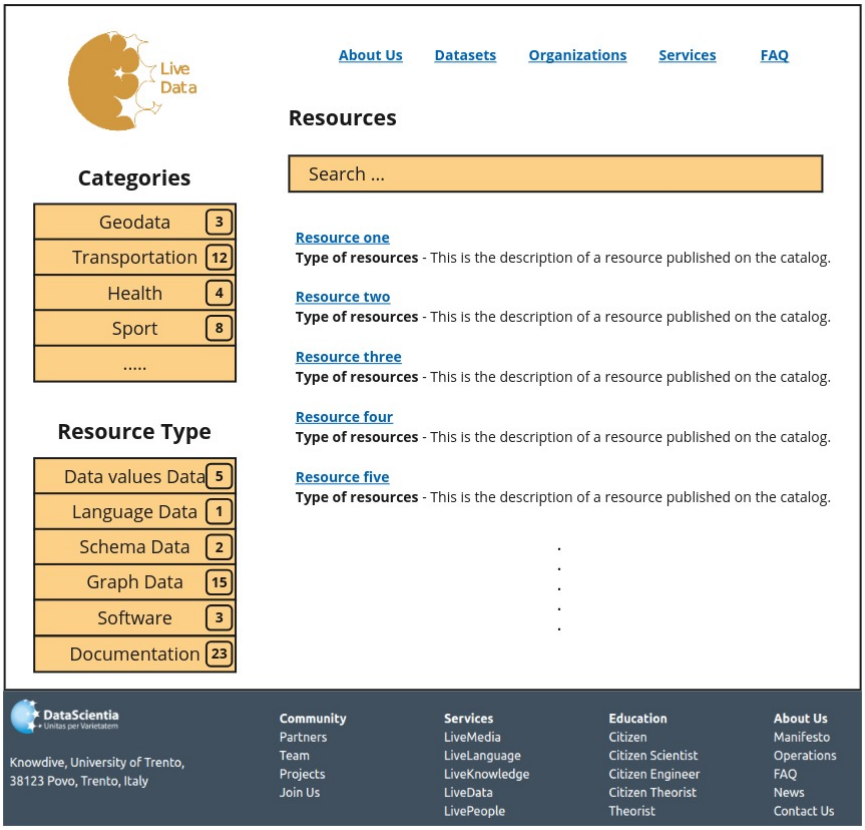


Figure 5.6: The LiveData catalogue search web page.

The LiveData catalogue search web page

The second layer of the LiveData catalogue web pages is accessible through the "Browse All" button, or through one of the several buttons available in the filter groups provided at the end of the catalogue landing page. Such a layer is designed to allow the users to search the resources they are looking for, between those distributed by the catalogue. For this objective, the LiveData catalogue architecture offers the web page depicted in figure 5.6. The search web page is mainly divided into two sections. The first section, in the centre of the web page, shows the users a list of all the resources

available in the catalogue. Each resource is displayed with its name and a brief description. Above the resource list, a search bar allows the users to search specific resources by using keywords. When the user types something in the search bar, the list below is filtered by showing only the resources having, in the name or in the description, the keywords written in the search bar. By clicking the name of a resource in the list, the users can jump into the third web page layer of the catalogue, illustrating the information regarding a specific resource. The second section, at the left side of the web page, allows the users to apply the filters that were already available in the landing page, by clicking the labels grouped by category of resources and type of resources. Each label corresponds to a category or a type of resource respectively. By clicking a label, the filter is applied over the resource list, displayed in the centre of the web page, where only the resources corresponding to a specific category or type remain visible. The users can select only one label per filter. In other words, one label is in the category group and another label can be selected in the resource type group. Moreover, each filter label is associated with a resource counter (visible close to each label, in figure 5.6) indicating how many resources of that category, or type, are available in the catalogue.

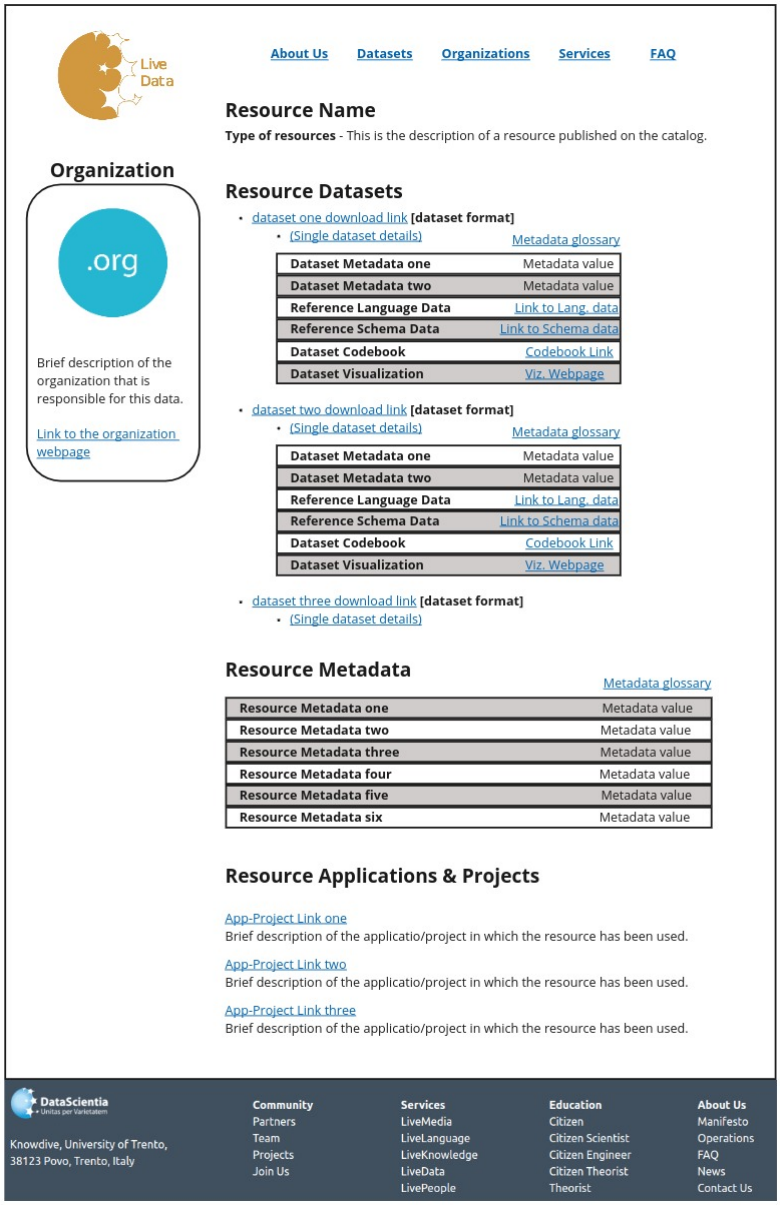


Figure 5.7: The LiveData catalogue resource specific web page.

The LiveData catalogue resource specific web page

The third layer of the catalogue web pages has been designed to provide information about a specific resource. A mock-up example of a single resource web page is depicted in figure 5.7. As clearly appears in the figure, the left side of the web page is a space dedicated to the information about the organization that is responsible for the specific resource. Such a space reports the

organization logo plus a brief description of the organization and its mission (objective/Purpose), as well as the possible links to other organization's web pages. The core part of the web page as depicted in figure 5.7, shows on top of the resources name and its brief description, like it was displayed in the resources list in the previous web page. The current web page is divided into three main sections described here below.

Resource Datasets - This section aims at providing the download links of the different datasets that can be part of a resource. This section actually is a list of datasets, where each dataset is presented to the users through two elements; (i) the download link immediately followed by a label indicating the format of the file that will be downloaded by clicking on the link; and (ii) the datasets level metadata. Such metadata are specific information regarding the single dataset (for example, the dataset size and the dataset's last modification date). Here below is a list of the most important dataset-level metadata provided by the LiveData catalogue.

- *Metadata glossary*: Even if a catalogue's data is described by a rich metadata set, it can be difficult for the catalogue's users to understand the meaning of metadata keywords. For example, by reading the metadata keyword "Data Owner", it is not clear which is exactly the role of the owner related to the data, what the owner can do and if the owner is the reference person for specific uses of the data. To mitigate this lack of information, a best practice is to create a dedicated catalogue's web page where all the metadata fields are described, with the objective of disambiguating their meaning and increasing the user's level of understanding about the data distributed. To be even more efficient each catalogue's web page reporting different metadata for different types of datasets (or resources) can provide a link to the metadata glossary (metadata description web page) so that it is immediate for the users to access such information. The LiveData catalogue provides access to the metadata glossary, as depicted in figure 5.7, through a link located at the top-right corner of each different metadata table (listing different metadata).

- *Reference Data values, Language and Schema metadata*: As reported by the research works [12, 45] that led to the creation of the iTelos diversity-aware data, the information that can be extracted, and used, from the data is not always maintained in the single datasets that most of the nowadays catalogues and data web portals are used to provide. The information is stratified into different layers which can be separated into different datasets (diversity layers) to be exploited individually or as a composition (diversity-aware data). The LiveData data distribution is based on such stratification too, and for this reason supports the distribution of 4 types of resources: language data, schema

data, data source and graph data. Such stratification is reflected also at the metadata level, by the definition of particular metadata fields able to link datasets with their own schema, language and data values datasets (when they exist). Thanks to these metadata fields, the LiveData catalogue's user can perceive the stratification described above, and thus be able to exploit the resources individually or as a composition. This metadata is provided as links to other (or other catalogue) resources, namely "Reference Language Data" and "Reference Schema Data" (if the explored dataset is a graph-based dataset, there will also be the "Reference Data values Data"). As it is visible in figure 5.7, such links are available in the dataset-level metadata table.

The metadata visualized in this section is defined to link the catalogue web pages describing datasets generated by the same execution of the iTelos KGC process. To better understand such a catalogue feature, we define the iTelos transformation of a given "low-quality" (non-diversity-aware) dataset, named X, into a diversity-aware data package, composed of four content types, where S is the data values dataset, L is the language dataset, K is the knowledge dataset and G is the composed graph-based dataset. The metadata set published on the catalogue web page for dataset G contains the link to the catalogue web pages describing datasets L, K and S, thus linking the graph-based dataset to the LiveData contents used to build it. Following the same approach dataset K has specific metadata linking to the web page for dataset L, which has been used to define its ETypes and properties names. Such connections between the LiveData contents, represented by dashed lines in figure 5.1, implement at the catalogue level, the compositionality of the contents handled by LiveData, or, in other words, the stratified composition of an EB, as defined by the EB data model. It is important to note how, the metadata defined for a dataset in the current LiveData node, can link to the dataset's web page published on the catalogue of another node since the external dataset has been reused to generate content of the current LiveData node. For example, a knowledge dataset of one node can be reused to generate the graph-based dataset of other nodes. For this reason, in Figure 5.1, the LiveData catalogue is linked to other LiveData nodes by a dashed line, describing the metadata links between two or more nodes.

- *Codebook*: most of the time catalogues, and other data web portals, provide metadata about the data, by describing characteristics of the datasets (CSV file, JSON file, RDF file, and other kinds of datasets). However, the users are not able to understand how to concretely exploit the data, because they don't know how such data is structured inside the datasets. The users will discover that only once the data is downloaded. To save time, and the cost of data download (the

download may be chargeable), a best practice is to create a catalogue's web page for each dataset distributed, explaining the content of the dataset. Such an explanation can be done by describing the meaning of the properties composing the dataset schema, as well as describing the values that can be associated with such properties. Such a set of information is called codebook and it is implemented as a web page, and accessed through the link available in the dataset-level metadata table, as depicted in figure 5.7.

- *Visualization page*: to integrate the information provided by the inner metadata, another best practice is to provide, on the same web page or in a separate one (through a dedicated link), the possibility to explore the data by navigating its content. Concretely speaking, this can be offered to the users through a data visualization tool allowing the navigation of a portion of the datasets. The implementation of this functionality for the catalogues increases the level of understanding of the users about the data distributed. Nevertheless, this feature requires considerable effort due to the fact that different types of data require different visualization tools. For example, a tabular dataset and a graph-based dataset cannot be navigated in the same way. In figure 5.7 the link to each dataset's visualization page is available in the dataset-level metadata table.

Resource Metadata - The resource metadata section provides the metadata at the resource level. Together with the relative metadata glossary (like for the dataset-level metadata) such information is provided as a tabular list of key-value pairs, where each row represents a resource-specific metadata. For this type of metadata, the iTelos methodology provides a predefined metadata schema to be adopted as a minimum and mandatory requirement, which can be then extended by the data producer to add more resource-specific metadata. Figure 5.8 shows a real use case example of resource level metadata, extracted from an existing LiveData catalogue [4] developed to collect and distribute data about the organization of the university of Trento (Italy) as well as its education and research activities.

[4] LiveData UNITN

Metadata glossary

License	Creative Commons Attribution Share-Alike
Category	Digital University
Resource Type	Graph
Maintainer	Simone Bocca
Maintainer Email	simone.bocca@unitn.it
Creator	Simone Bocca
Creator Email	simone.bocca@unitn.it
Publisher	DataScientia Foundation
Owner	University of Trento
Validator	Simone Bocca
Keyword	papers, research, pubblication
Domain	University of Trento
Language	Italian, English
Generating Activity	DataScientia LiveData UNITN Catalog Publication
Modification Date Time	01/05/2024

Figure 5.8: Resource specific metadata in LiveData catalogue.

Resource Application & Project - The last section of the LiveData catalogue resource-specific web page collects the so-called "project and application level metadata". This type of metadata provides information regarding the projects and applications that have been done by using the data distributed by a catalogue. The number of applications and the number of projects, together with their descriptions are part of the information described by this metadata. This metadata can be provided on the same web page together with the other catalogue's metadata or on a separated catalogue's web page. The user, by reading this information can better understand how the data distributed by the LiveData catalogue can be used. If there are no known projects or applications already developed with such data, it can be useful in any case to provide some examples of usage of the data exploitation, in order to give some hints to the users who are interested in understanding how to produce concrete value (application and data-based services) by using the data. As depicted in the lower part of figure 5.7, the project and application metadata can be provided as a list of projects and/or applications which have been developed by using the resource described by the relative web page. Each item of such a list is represented with the name of the application/project and its brief description. The item's name, in the example provided by figure 5.7, is a link which leads the user to a separate

web page where the relative project or application is illustrated. Nevertheless, other solutions can be adopted to show the information regarding each application and project.

5.3.4 Data Services

The services are the components of the LiveData node architecture that allows LiveData users to interact with the node, thus actually implementing the communication between the LiveData network and its users. The LiveData users can be both the administrators of the nodes, who use the LiveData services to collect, transform and distribute new data and users who are not involved in the administration of the LiveData network. The latter category of users is interested in using the LiveData services to browse and download the diversity-aware data published on the LiveData catalogue. To support the needs of such users each LiveData node provides four services, as depicted in figure 5.1. It is important to mention that the implementation of the LiveData services is dependent on the software and hardware environment in which the LiveData node is deployed. While the above described LiveData node components are more technology-independent, the LiveData node services define the interaction between the LiveData components, as well as between such components and the LiveData users. For this reason, their implementation has to be aligned with the local context. Although the first stable implementation of the LiveData node services is considered by this research as part of future work, we report here below the definition and objectives of each LiveData node service.

- *Data Transformation*: the data transformation service is used by the LiveData node user, playing the role of data producer, to collect local low-quality (non-diversity-aware) data with the aim of transforming it into diversity-aware data (following the EB data model) and making it available in the LiveData network. Such a service, as shown in Figure 5.1, is connected to the KGC process, which provides the process and tools (iTelos KGC process) to implement the collection and transformation of data to be stored in SREP, CREP and DREP as already mentioned describing the LiveData node repository partitions. It is important to note that the goal of the data transformation service, is to collect and transform existing non-diversity-aware data into data compliant with the EB data model, thus, in other words, to apply the iTelos KGC process as a data producer.

- *Data Integration*: the data integration service is used by the LiveData node user playing, instead, the role of data consumer, to integrate together the diversity-aware data previously transformed by the data transformation service. This means that the data integration service

works only with the data available in the LiveData repository, without considering any other external data source. Again, as shown in figure 5.1, the data integration service is connected to the LiveData KGC process component, which provides access to the iTelos KGC process along with its supporting framework. We note now how the data transformation and data integration services, are the LiveData node endpoints providing access to the iTelos KGC process, as data producer and data consumer, respectively.

- *Data Distribution*: the data distribution service is used by the LiveData node user to access the diversity-aware data to be distributed from the DREP repository partition, with the aim of making it available to the users (or systems) that require it. As depicted in figure 5.1, the LiveData node data distribution service is connected to (if not directly integrated with) the LiveData node catalogue, where it implements the access policies and the download of the required diversity-aware data. The data download can be implemented in different ways depending on the data distribution policy defined by the data owner. For example, one LiveData node can implement this by submitting a download request to the data owner, while another node may implement this by an automatic download procedure. Moreover, the data distribution service is connected to the DREP repository partition, to retrieve and distribute the data stored in such a repository partition.

- *Data Search*: the data search service is used by the LiveData node users to look for diversity-aware data suitable to satisfy their own Purpose. The search service is used directly in the LiveData node catalogue, thus allowing the users to browse the data by exploring the values of the metadata defined to describe it. For this reason, in figure 5.1 the data search service is connected to the LiveData node catalogue. The search service can be implemented in different ways, with more or fewer search features, with the objective to improve the quality of the services given to the LiveData catalogue users. The first basic implementation of such a service is present in the LiveData catalogue search page, where catalogue users can search for resources of interest by the values of the metadata used to define them.

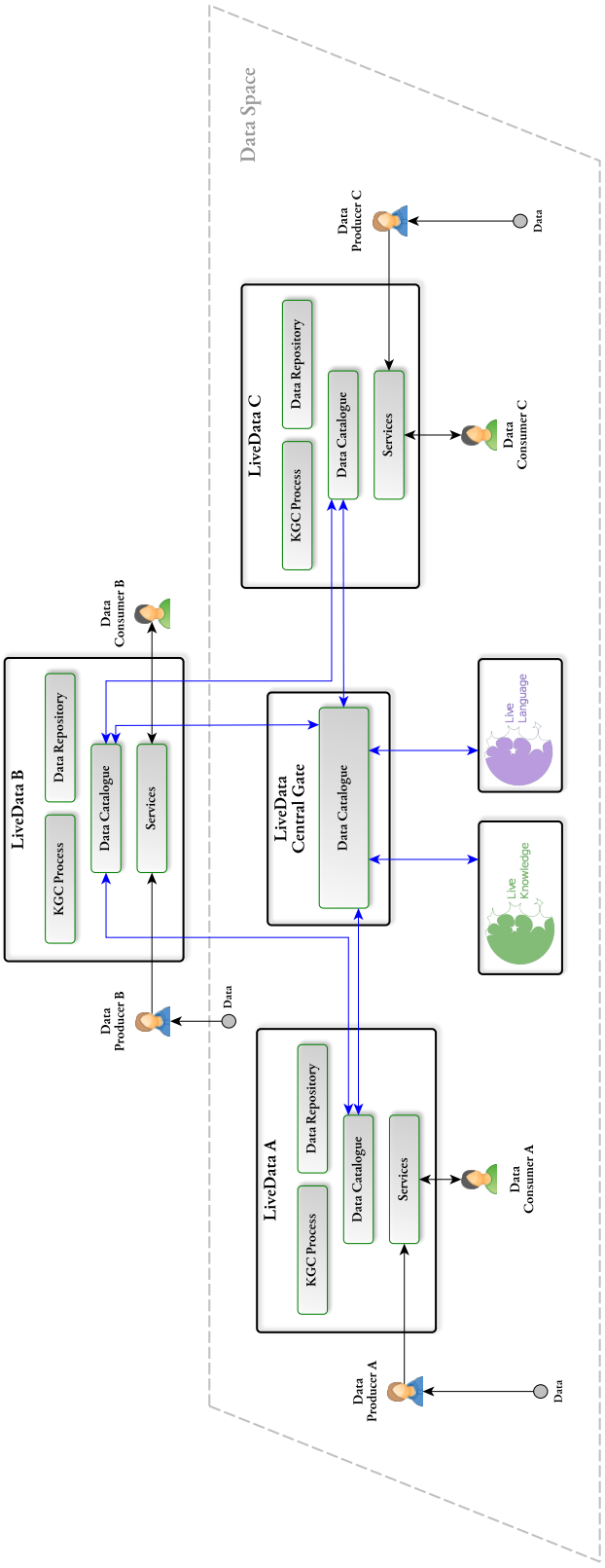


Figure 5.9: The LiveData network architecture.

5.4 The LiveData network architecture

The objective of this section is to describe the architecture of the LiveData data distribution network, by describing how the LiveData nodes are distributed over the LiveData data mesh network, as well as how they interact with each other thanks to the iTelos DI coordination. Figure 5.9 depicts a representative example of a portion of the LiveData data distribution network, including different nodes, as well as the users interacting with such nodes. This figure is actually a more detailed representation of the *LiveData Space* represented in figure 4.5 showing how the iTelos KGC process interacts with "external" data sources (out of LiveData network), and "internal" data sources (nodes of the LiveData network). The composition of the two figures provides a complete overview of the interactions between the iTelos KGC process and the LiveData network, driven by the Purpose of the iTelos methodology users, both data producers and consumers. For this reason, in both the two figures, a general ("external") Data Space, is present in the background, representing the space of data composed by all the other available data spaces. From this background data space, the iTelos data producers collect low-quality (non-diversity-aware) data (grey dots connected with data producers by black arrows in figure 5.9), with the objective to transform it into diversity-aware data, following the iTelos data representation process (based on the EB data model), implemented concretely by the iTelos KGC process, which is available within the architecture of each LiveData nodes. The key idea for the future of the LiveData network is to work towards the definition of an agent-based system that can automatically support the users of the data community defined by the iTelos DI infrastructure, in the creation, integration and distribution of their diversity-aware data. Another important thing that is visible in figure 5.9 is the fact that different types of LiveData nodes are present in the network. Even if such nodes have the same architecture, as described in section 5.3, different types of nodes act with different Purposes within the network. In this section we describe the different types of nodes present in the LiveData data distribution network, how they act in the network, as well as how they interact with each other.

LDE nodes - The first and most common type of nodes, in the LiveData network, are those used to handle diversity-aware data for a specific LDE. The LDE, already defined in the previous section, defines the spatial, temporal and social boundaries of the diversity-aware data handled and distributed by this node. In figure 5.9, the LDE nodes are represented by the solid line boxes named LiveData A, LiveData B and LiveData C, respectively. Each LDE node in the LiveData network is structured following the LiveData node architecture thus including the four main components, namely the data repository, the KGC process, the data catalogue and the data services. As described in section 5.3, and visible in figure 5.9, the LDE nodes can be con-

nected to each other through the links (blue arrows in figure 5.9) concretely implemented by the values of the metadata used to describe the resources published on the LiveData node's catalogues. Such metadata values, in fact, can link to resources published by other LiveData nodes catalogues. For example, if a knowledge resource (i.e., KT or an ETG) is adopted in two different LDEs. Moreover, a LDE node, in the LiveData network, interacts with the iTelos methodology users (black arrows in figure 5.9), both data producers and data consumers, living in the ecosystem for which the LDE node has been defined. As already described in section 5.3, the users interact with the LiveData nodes through the services provided by the node architecture, allowing each user to transform and integrate, as well as to search (through the catalogue) and download the data they are interested in. An example of an LDE node is the LiveData node, called LiveData UniTN [5], developed to collect and distribute data about the organization of the university of Trento (Italy) as well as its education and research activities.

Mediators nodes - The second type of nodes present in the LiveData network are called Mediator nodes. These nodes can be defined as instruments of the iTelos DI used to improve the interoperability and reusability of the diversity-aware data produced by the other nodes of the network. Mediator nodes are nodes within the LiveData networks administrated directly by the iTelos DI (as a third non-profit role) producing, and distributing standardized reference resources. More in detail, the mediator nodes are two nodes within the LiveData network, one distributing knowledge-layer resources (like KTs, and ETGs), represented by the *LiveKnowledge* [6] green component in figure 5.9, and another one distributing language-layer resources (like LTs, natural and domain languages vocabularies), represented by the *LiveLanguage* [7] purple component in figure 5.9. The knowledge and language resources provided by the mediator nodes are reference resources that can be reused to represent information about a specific domain of interest. Such resources have a high level of quality in terms of knowledge and language representation, modelling well-known concepts and ETypes largely adopted in specific domains of interest. Moreover, all the resources distributed by the LiveData mediators nodes have been already aligned with the UKC's LO, thus being more interoperable and easily integrable adopting the iTelos KGC process. Examples of knowledge resources distributed by the LiveKnowledge mediator node are the DBpedia KT [8], and the Open Street Map (OSM) KT [9]. An example of a language resource distributed by Live-

[5] LiveData UNITN
[6] LiveKnowledge catalogue
[7] LiveLanguage catalogue
[8] DBpedia LiveKnoweldge KT
[9] OSM LiveKnowledge KT

Language, instead, is the vocabulary of Tunisian Arabic language [10]. The iTelos DI, through the LiveData mediators node, support the generation of interoperable diversity-aware data for the LDE nodes of the network. The term "mediators" used to define this type of LiveData nodes, comes from the agent systems community, in particular from the work of Finin, Tim, et al. [31] where facilitators and mediators agents are described as "agents that perform various useful communication services" with the goal of enhancing the sharing of knowledge. This definition is actually in line with the objective of the iTelos DI mediators nodes, by considering also the future vision of an agent-based LiveData network.

Individual nodes - The third type of nodes included in the LiveData network is called individual nodes, namely LiveData nodes developed and focused on the diversity of a single person. This is a very specific and particular type of LiveData node, always designed by following the LiveData node architecture (see section 5.3) but with the objective of creating, integrating and distributing data about a particular Purpose, that of a single person. The individual nodes are rarer than the LDE nodes, within the LiveData network, for this reason, they do not appear in figure 5.9. However, they represent the most granular support to the generation and distribution of diversity-aware data, that the LiveData network can provide. The individual nodes allow for the sharing of data in which the diversity level is higher with respect to the data handled by LDE nodes. As a consequence, is higher also the information and knowledge derived from such data. In other words, the individual nodes are the representation of the diversity of individual people within the LiveData network. This allows each person to consider herself as an active member of the data community supported by the iTelos methodology. To provide a concrete example of a LiveData individual node, let's assume that a student of the University of Trento (Italy) decides to join the data community supported by the iTelos methodology, thus having her own Individual node, to collect and distribute the data about her own daily life, thus for example:

- The university facilities she is used to frequent: the computer science department, the university library, and the university canteen.
- The transport services she uses: the bus number 5 route and timeline, the transport to and from the train station.
- The food facilities she is used to frequent during lunch time: supermarkets, bars and restaurants.
- And others.

[10] Tunisian Arabic vocabulary

All the above information is focused on the student. The student's individual node does not contain the data of the entire transportation service database of the city of Trento, nor the data about all the university facilities in the city. It presents only the data that is valuable for the student's daily life. The student can decide what and how to add, update and distribute, by using her own LiveData individual node. A simple example of an individual catalogue has been developed for Simone Bocca (PhD candidate), called *LiveData Simone Bocca* [11].

LiveData gateway node - The last type of node present in the LiveData network is the LiveData gateway node. It is actually a single node, which acts as the main entry point of the whole LiveData network. The main characteristics of these nodes are two. The first is that the architecture of this node is different from all the other types of nodes in the LiveData network. In fact, this node is composed of a catalogue only, which is actually a catalogue of catalogues. The resources distributed by the LiveData gateway node catalogue are actually the catalogues of the other nodes in the network. In other words, in the LiveData gateway node catalogue, it is possible to navigate and explore the metadata describing the ecosystems (also the individual ecosystems if the relative owners allow to be accessed from the gateway node) of the other nodes in the LiveData network. It is also possible to access the catalogues of the other nodes directly from the catalogue of the gateway node, by navigating through the links available as metadata values within the pages of the catalogue of the gateway node itself (blue arrows connecting the LiveData gateway node's catalogue with the other network's catalogues, in figure 5.9). The current version of the LiveData gateway node is accessible through its catalogue, called simply *LiveData*[12].

5.5 The iTleos data intermediary process

This section aims to detail the last process of the iTelos methodology, namely the iTelos DI process. Being the three iTelos methodology's processes encapsulated each other, this process includes both the iTelos data representation and the iTelos KGC processes, with the objective of providing a complete solution for the reuse of data. This process is strongly based on the definition of the iTelos DI, as well as on the definition of the iTelos DI infrastructure, implemented by the LiveData network. Figure 5.10 depicts a top-level representation of the process. Like the other two processes, already defined in chapters 3 and 4, the iTelos DI process is composed of four phases, described here below.

- *Data retrieval*: the retrieval of existing data, based on a given Purpose

[11] LiveData Simone Bocca
[12] LiveData gateway node catalogue

to be satisfied, by considering both the data produced by the iTelos KGC process, thus following the iTelos data representation process (EB data model), as well as the existing data provided by other data sources which need to be transformed into diversity-aware data.

- *Data production*: the generation of new diversity-aware data starting from resources provided by data sources which do not adopt the iTelos methodology.
- *Data integration*: the integration (or composition) of already created diversity-aware data to satisfy a given Purpose. This phase aims to be a solution to reduce the cost of data reuse, by exploiting the interoperability of the diversity-aware KGs produced by the iTelos KGC process.
- *Data distribution*: the sharing of the diversity-aware data produced in the data production and integration phase. This phase plays a crucial role since there is no reuse without data sharing. For this reason, this phase is actually connected to the first phase of the iTelos data intermediary process, thus creating a loop defining a sustainable solution for the reuse of data.

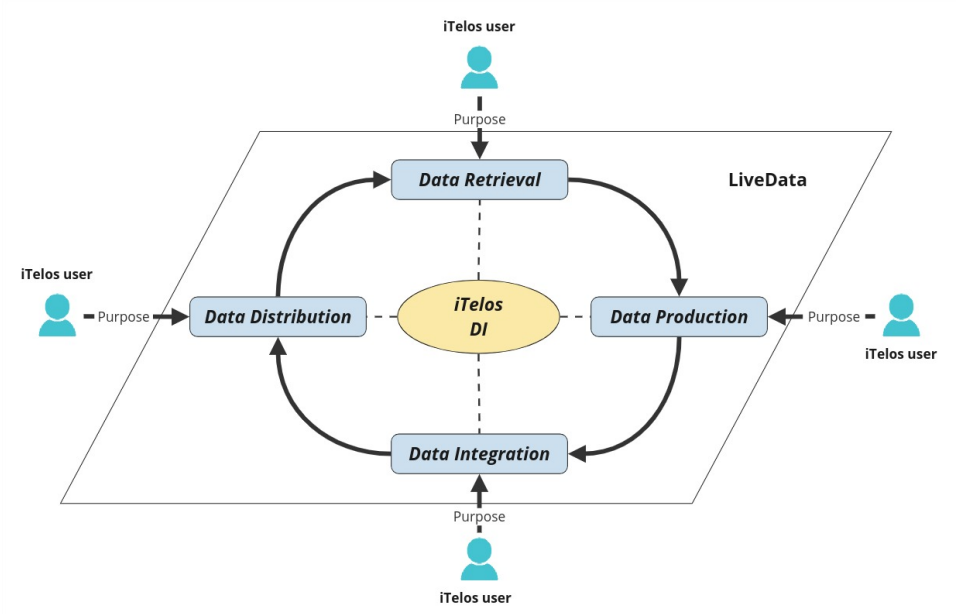


Figure 5.10: The iTelos DI process.

The four phases of the process can be executed by different iTelos users individually (i.e., executing only one phase, or a portion of the whole process)

or sequentially, following the entire process workflow. The four phases of the process define how to reuse existing data to create, integrate and share diversity-aware data. This means that different users, each one with his or her own Purpose, can interact by performing different phases of the process. The iTelos DI is the provider of the iTelos DI process, meaning that she supports the iTelos methodology users along the execution of the different phases of the process. Moreover, as already described in section 5.1, the iTelos DI acts as a coordinator between the different users who are executing different phases of the process. For example, the iTelos DI coordinates the interaction between a user that is acting as a data producer performing the data production phase, and another user acting as a data consumer in the data integration phase. Another example of coordination provided by the iTelos DI is between a user that is distributing diversity-aware data, and another one that is instead retrieving it, for a different Purpose. In the latter case, the iTelos DI is responsible for establishing the appropriate connection between who is searching for the data and who is distributing the most appropriate data to satisfy the Purpose of the first user, which is performing the data retrieval phase. The iTelos DI process defines a continuous loop of data reuse and new diversity-aware data generation. Each phase of the process can be a start or an end point of the process. Moreover, the process is defined to support the cooperation between different users, under the coordination of the iTelos DI. For this reason, the iTelos methodology can be defined as a cooperative methodology for diversity-aware data (KGs) generation and reuse. The following subsections aim to provide more details for each phase of the iTelos DI process. The process described in this section has been elaborated in the last months of the research periods interested in this thesis. For this reason, this section and the corresponding subsections provide a top-level view of the process that needs to be refined in order to provide the users of the iTelos methodology with precise methodological and technological support for the generation and reuse of the data they are interested in.

5.5.1 Data retrieval

The data retrieval is the first phase of the iTelos DI process. It is important to mention that, this is the first phase we choose to describe, but actually, being the iTelos DI process a loop, it is meaningless to define a general start and end point for the whole process. In this phase, the objective of the iTelos DI, as a provider of the process, is to support the user in the identification and collection of the data she is looking for satisfying her Purpose. To this end, the telos DI uses two main instruments, the first one is the LiveData network, where the different LDE nodes (see section 5.4) act as entry points for data ecosystems in which the user can search the data she is interested in. The second instrument is the iTelos KGC process, which is accessed through

the architecture of the LiveData node (see section 5.3) the user interacts with. In fact, the iTelos KGC process, in its first phase (Purpose Definition phase, described in section 4.2), allows the user to formalise her Purpose and to collect the data needed to fulfil it. As the aim of the iTelos DI process data retrieval phase is to search and collect data, there are two cases in which this phase is performed by the user:

- in the first case, the data required by the user is fully available from the LiveData network. In other words, the data that the users are looking for has been completely processed by the iTelos KGC process, thus transformed into diversity-aware data, and distributed to one or more nodes in the LiveData network. In this case, the iTelos DI first makes the iTelos KGC process available to the user, with the objective of formalising her Purpose. Then, during the execution of the Purpose Definition phase of the iTelos KGC process, the user is driven through the LiveData network with the objective of matching the formalized Purpose, with one or more of the LDE nodes available in the network. The key idea is that the matched nodes should be those distributing the diversity-aware data the user is looking for. Note that more iterations over this source identification activity (see iTelos KGC process Purpose Definition phase activities in section 4.2.1) could be required to find the interested data. Once the right LDE nodes have been identified, the user can use the LiveData node services to search for the specific data she is looking for, as well as to download such diversity-aware data through the LiveData node's catalogue.

- In the second case, the data required by the user is not fully available from the LiveData network. In other words, a portion of the data that the users are looking for has not yet been processed by the iTelos KGC process, thus transformed into diversity-aware data, and distributed to a node in the LiveData network. Also in this case, the iTelos DI makes the iTelos KGC process available to the user, with the objective of formalising her Purpose, and the LiveData network is explored to match the nodes distributing the portion of the data available in the LiveData network. However, unlike the previous case, an additional step is required. Once the interested LiveData nodes have been identified, the iTelos DI drives the user into the next phase of the iTelos DI process, namely the data production phase (described below in this section), with the objective of collecting the portion of data not available in the LiveData network, from others ("external") data sources. Note that the data collected from data sources outside the LiveData network is not yet transformed into diversity-aware data according to the iTelos data transformation model, and therefore it is not compliant with the EB data model.

It is important to note that in this phase of the iTelos DI process, the iTelos DI has to find a proper match between different Purposes. One is the Purpose of the user searching for data, while the other Purposes are those of the users who created the diversity-aware data distributed within the LiveData network (actually also the data outside the LiveData network is defined based on a different Purpose). As already mentioned in chapter 1 the Purpose is always present in the data, and it is used to implicitly or explicitly shape the data. The diversity-aware data generated by the iTelos methodology explicitly represents the Purpose as a component of the real-world entities represented by the data itself, following the EB data model. This enables a simpler (semi-automatic matching) and more cost-effective alignment between different Purposes.

5.5.2 Data production

As already mentioned in the data retrieval phase, the data production is the phase of the iTelos DI process where the DI supports the users in the generation of new diversity-aware data. The key instrument used by the iTelos DI in this phase is the iTelos KGC process, provided to the users through the relative component of each LiveData node. As described in the previous phase, when the user needs to retrieve data to satisfy her Purpose, she is driven by the iTelos DI to the LDE nodes which handle the data she requires. It may happen that the diversity-aware data distributed by such LiveData nodes is not sufficient to satisfy the user's Purpose. Nevertheless, each LDE node is managed by a domain expert actor who has the necessary knowledge to help the user find other data sources within the same LDE (the one the domain expert lives in and interacts with) that can be transformed into diversity-aware data and then used to satisfy the user Purpose. In other words, the user plays the role of data consumer, as intended by the iTelos methodology, while the domain expert plays the role of data producer. The goal of the iTelos DI in the data production phase of the iTelos DI process is to bring the data producer and the data consumer into contact with each other in order to foster their cooperation. Once the LDE node's domain expert and the user establish cooperation, the domain expert exploits the iTelos KGC process to collect and transform already existing data, extracted from other data sources into the relative LDE, into diversity-aware data required by the user to satisfy her Purpose. The benefits of the data production phase are threefold. The first is focused on the user, who is supported by the iTelos DI and the local (LDE) domain expert in the fulfilment of her Purpose. The second one instead is focused on the domain expert whose work and expertise were requested by the user. In other words, the iTelos methodology, through the coordination of its DI, enhances the status of the domain expert job, increasing its importance and the economic value of the services it provides. This is particularly important in social contexts where people

with knowledge and expertise find it difficult to find employment. The third benefit relates to the LiveData network, which is enriched by the generation of new diversity-aware data, thus actually improving its ability to support data reuse by providing data that has already been transformed into more interoperable diversity-aware data.

5.5.3 Data integration

The data integration is the third phase of the iTelos DI process, following the order we are adopting to describe the whole process. As the name suggests, the objective of this phase is to integrate data to satisfy a given user's Purpose. There are two key intuitions that allow this phase to be more cost-effective in terms of time and effort. The first is to use the iTelos KGC process to execute the data integration, while the second is to focus the integration over diversity-aware data only, thus considering only the data produced by the data production phase (described above in this section). The diversity-aware data is more interoperable, thanks to the standard language and knowledge resources adopted for its construction by adopting the iTelos KGC process (data producer side). Therefore, by using the iTelos KGC process (data consumer side) to integrate such data, the integration cost is less costly in terms of pre-processing and time required. It is important to note that, the iTelos KGC process is designed to take care of each layer in which the diversity-aware data is structured, namely, Purpose, language, knowledge and data values. Based on the above described intuition, the data integration phase is executed by a user having a specific Purpose. The iTelos DI, as provided by the DI process makes the iTelos KGC process available to the user, which in this case acts as a data consumer. By executing the process phases, the user is able to first formalize her Purpose, which is then used as a parameter for the navigation of the LiveData network, looking for LDE nodes handling the data able to satisfy that. Once such LDE nodes have been identified, if the data required to fulfil the user's Purpose is already distributed by the selected LDE nodes, the user downloads such diversity-aware data and proceeds with the execution of the remaining iTelos KGC process phases with the aim of integrating such data into a unique EG capable of fulfilling its Purpose. Instead, if the data required to fulfil the user's Purpose is not fully provided by the selected LDE nodes, the user must restart the iTelos DI process from the data production phase to build the diversity-aware data required for her Purpose.

5.5.4 Data distribution

The data distribution is the last phase of the iTelos DI process to be described. As mentioned above, there is no data reuse without data sharing, so the objective of this phase is to support the distribution of the diversity-

aware data produced by the data production and data integration phases since the composition of existing diversity-aware data actually represents a new diversity-aware data (based on a new user's Purpose). Therefore, diversity-aware data is generated both during the data production phase, when the domain expert acts as the data producer, and during the data integration phase, when the user acts as the data consumer. The key instrument provided, by the iTelos DI, to support the user during the execution of this phase, is the LiveData network (see section 5.4) and its LDE nodes (see LDE nodes architecture in section 5.3). On the data producer side, the iTelos DI provides the domain expert with the ability to set up his own LDE node with the ability to manage and distribute the diversity-aware data for the respective LDE. On the data consumer side, the iTelos DI provides the user with the possibility to share the diversity-aware data generated after the execution of the data integration phase. In the latter case, the iTelos DI connects the user, as the owner of the newly generated data, with the domain expert who manages the LDE node appropriate for the data to be distributed. The domain expert and the user then jointly define the best data distribution policy to ensure the right level of protection and security for the data to be distributed. It is important to note how such cooperation between the user and domain expert brings benefits to both actors. The domain expert receives new data to distribute through the relevant LDE node, thus actually increasing the amount of information and knowledge she can distribute. The user could decide on the best policy for the distribution of her own data, which may include payment in the form of specific data services in return for the data she decides to share.

6

iTelos educational Use Case - KGE

Contents

This section describes the first of the two use cases in which we applied the iTelos methodology, with the objective of testing and evaluating its characteristics. The current use case is applied in an educational context. More in detail, we have extensively evaluated and revised the first version of the iTelos methodology during the years 2018, 2019 and 2020 as part of the Knowledge and Data Integration (KDI) class, a six-credit course for Master's Degree in Computer Science of the University of Trento[1]. Thanks to the feedback received from the students of the KDI course, over the years, it has been possible to elaborate a different version of the iTelos methodology. More in detail, the first version of the methodology was mostly focused on the integration of data and knowledge resources. While the new version of the iTelos methodology is focused on the whole process of data reuse, thus involving data retrieval, the representation of existing data as diversity-aware data, as well as data integration and distribution. The new iTelos version, which is actually the version described in the previous chapters of this thesis, has been evaluated in the past three years (2022, 2023, 2024) as part of the Knowledge Graph Engineering (KGE) class, a six-credit course of the Master Degree in Computer Science of the University of Trento. See the web page of the 2024 edition of the KGE course [2] for more details. In this thesis we report the result of the evaluation performed over the version of the methodology applied in the KGE course, being the subject of this research work. For the details about the evaluation conducted over the KDI course during the years 2018, 2019 and 2020, it is possible to see the works [37, 11]. The following sections of this chapter provide a description of the KGE course and the projects carried out by the KGE students in which they applied the

[1] KDI-2020 web site
[2] KGE24 webpage

iTelos methodology over the years. The last section reports the results of the evaluation carried out by the students of the KGE course by means of dedicated questionnaires.

6.1 Knowledge Graph Engineering (KGE) course

In order to provide a clear context for the iTelos evaluation performed in this use case, this section aims to describe the KGE course. As already mentioned above, KGE is a 6-credit course, taught in English in the Master Degree in Computer Science of the University of Trento (Italy), having the following objectives.

- To explain the problem of data reuse by describing not only the processing of the data but also its heterogeneous representation at different levels. Furthermore, the course aims to explain data reuse by considering the effort required for data retrieval and data distribution activities.
- To teach what are Knowledge Graphs, which are their possible usages and features, and what it means to build a KG. During the curse, the students will discover the different issues to be addressed when a KG is built, learning which are the activities to be executed to that end, as per the current state of the art on Knowledge Graph Engineering.
- To teach the iTelos methodology as a solution for the data reuse problem, applied for the construction of KGs composed into EBs. Moreover, such a process will be supported, within the course, by the usage of specific tools and libraries that the students will learn in order to solve the issues encountered.

KGE is a 14-week, 48-hour, advanced course on how to develop a Knowledge Graph (KG) starting from data – to be cleaned and adapted – which are already available. This is a hands-on course. After a few introductory classes, students are given a problem to solve and they will build a knowledge graph to solve this problem. A limited set of teaching materials is available, mainly in the form of slides. The student will learn mainly by doing the actual work and by interacting with the teacher and tutors. More in detail the lectures of the course are organized as follows.

- The first 4 weeks of the course consist of theoretical lectures, where the lecturers explain the problem, the state of the art, as well as a top-level view of the solution (iTelos methodology) applied with its basic principles. The lectures in this first course's phase are supported by slides and reference material, discussed in class and provided to the students.

- Starting from the 5th week, when students are assigned a project to carry out (more details about the KGE projects are available in section 6.2), the KGE lectures become both theoretical and practical, and mainly focused on teaching the iTelos KGC process. Therefore, for each theoretical lecture describing a phase of the iTelos KGC process, a practical lecture explains how to concretely execute the phase over concrete examples (demo of the tools and running examples are used to teach practical lectures).

- Additionally, after the explanation of each phase of the iTelos KGC process (after both the theoretical and practical lecture), the lectures provide the students with a Questions & Answers (Q&A) lecture, where the students can ask the lecturers any kind of questions or doubts they have about the issues encountered during the development of their projects, or regarding the theoretical contents of the course. The Q&A lectures have the objective of supporting the students in the development of their projects, by addressing specific issues relative to their projects.

After the completion of each iTelos methodology phase (both concerning theory and practice), the students will have to provide an intermediate report of the work done so far, over their projects, which will be checked and evaluated by lecturers. This intermediate evaluation will allow the lecturers to lead the teams towards the right direction by correcting possible errors during the methodology implementation. The final exam consists of a presentation of the KGE projects developed along the course and finalized achieving the output required by the initial project's Purpose. In addition, some questions can be asked, at the end of the project presentation, to evaluate the students over the key aspects of the iTelos methodology.

The KGE course is a very important generator of data for educational Purposes. The students of the course, by developing their projects, collect, formalize, integrate and distribute new data that has been elaborated by the iTelos methodology thus capable of reducing the overall cost of data reuse. For this reason, at the end of each edition of the course, all the projects developed by the KGE students are collected and published on a dedicated LDE node of the LiveData network, accessible through its own data catalogue, called LiveData KGE [3]. The LiveData KGE catalogue is used as an input data source for the KGE projects developed over the years by the KGE students, thus allowing the evaluation of the data distribution (more in general the data reuse) approach pushed by the iTelos methodology.

The KGE course was taught for the first time at the National University of

[3] LiveData KGE

Mongolia, in Ulaanbaatar (Mongolia), during the spring of 2024. A dedicated course website has been developed for this edition of the KGE course [4]. The students experienced the development of diversity-aware data generation projects designed over the Mongolian context, thus involving the collection of data regarding the Ulaanbaatar (Capital city of Mongolia) geography, transportation systems, and daily life. Moreover, the Mongolian edition of the KGE course was the first university course of the computer science department to be taught in English. The projects developed during the Mongolian edition of the KGE course are available in the KGE catalogue, together with the projects developed by the UniTN students.

6.2 KGE projects

In this section, we provide a more detailed overview of the projects assigned to the students of the KGE course. The key idea behind the assignment of the KGE projects is first of all to teach the students how to build a KG able to solve the problem identified by a project's Purpose. While a second objective is to produce high-quality data, over specific domains of interest, which can be reused in the following edition of the course, thus supporting the iTelos data reuse approach. Each project is assigned to a team composed of two (or eventually three) students. The idea is that the components of the team should cover the roles required by the iTelos methodology (see section 4.1.2). At the beginning of the KGE course, the lecturers present different project proposals, which the students have to choose and develop during the course. Each project proposal is defined by three components:

- *The informal Purpose*: It is provided as a natural language (English) text and represents the objective to be achieved by developing the project. Here below we report an example of informal Purpose extracted from the KGE project proposals defined for the edition 2024 of the course.

  The Purpose of this project is to engineer a Knowledge Graph able to support applications and services providing information about the main Tourism Facilities (like Hotels, Restaurants, Museums, Natural parks, and others) on the Trentino Province territory.

- *The data resources*: it is a list of data sources from which the students can collect the data to be used to fulfil the relative Purpose. The data resources provided with the informal Purpose example described above are the following:

[4] KGE 2024 NUM

  - Trentino OSM - collect your data : this is a link to the Open Street Map (OSM) data portal from which the students can collect the geographical data they need.
  - KGE22 - Trentino Tourist Facilities : this is a link to a project from the 2022 edition of the KGE course, from which the students can reuse useful data, since the two Purposes are focused on the same domain of interest.
  - ISTAT Turismo : this is a link to the ISTAT data portal from which the students can collect data they need.
  - OPEN DATA TRENTINO : this is a link to the Open Data portal of the Trentino Province (Italy) from which the students can collect the geographical data they need.

- The knowledge resources: it is a list of knowledge sources from which the students can collect the data to be used to fulfil the relative Purpose. The knowledge resources provided with the informal Purpose example described above are the following:

  - Open Street Map - Trentino Territory Teleontology.
  - SCHEMA.ORG
  - SCHEMA.ORG LOV
  - GTFS STATIC
  - GTFS LOV
  - TIME ONTOLOGY

  The knowledge resources are data sources to be considered for the download of standardized knowledge models which could be reused to represent the ETypes and properties of the KG to be developed by the students, according to the relative Purpose.

The data and knowledge resources provided for each project proposal can be enriched by the students, if new data sources, distributing required data, are discovered during the first phase of the iTelos KGC process. The three components of each project proposal, represent the initial input of an execution of the iTelos KGC process (see section 4.2). The students of the KGE course, along the course duration, develop a KG (more in detail an EG) able to satisfy the Purpose given by the project proposal they chose, by following the different phases of the iTelos KGC process. Each project is supported by a dedicated GitHub repository where students upload intermediate results for each completed phase. At the end of the course, the results of each project are stored in these repositories and described by a dedicated webpage (project webpage). Thanks to the metadata created by each student team, each project and each output produced can be published in the LiveData KGE catalogue, ready to be reused in other iTelos projects.

6.3 iTelos methodology evaluation

The choice of the educational context and the set-up described above is twofold. First, we were able to observe the extent to which the iTelos methodology was appropriate for driving a group of skilled computer scientists, having limited background in knowledge engineering, through the entire KGC process. This is an important aspect if we consider that the iTelos methodology aims at supporting the creation of high-quality KGs by people who could have little or no expertise in KGC techniques (like many domain experts). Second, the context of the course allows to define a proper schedule for applying the entire methodology and for observing the timing aspects related to the execution of each phase of the iTelos KGC process. Notice that preliminary versions of the iTelos methodology were adopted and validated within the context of European projects. Below, we report the evaluation of the methodology related to a period of three years (2022, 2023 and 2024) [5]. Throughout this period, the methodology has been refined after each course cycle by addressing the feedback provided by the students and the experts. Table 6.1 shows the demographic distribution of the students engaged in validating the iTelos methodology from both the quantitative and qualitative perspectives.

	2022	**2023**	**2024**	**Tot**
# **Students**	25	11	28	64
# **Project teams**	15	6	13	34
% **Male**	64%	66.7%	64.3%	-
% **Female**	36%	33.3%	35.7%	-
% **Undergraduate**	24%	33.3%	21.4%	-
% **Post graduate**	76%	66.7%	78.6%	-

Table 6.1: Evaluation's subjects in KGE class - 2022, 2023, 2024.

The results of the first quantitative evaluation are reported in figure 6.1. We show three heat maps containing the answers to a set of quantitative questions submitted to users. The full set of questions can be found in the questionnaires used to collect the relative answers (we link the version of the questionnaire used in the edition 2024 of the KGE course [6]). We have selected 5 criteria, extracted from the questionnaire's answers, to evaluate

[5]The methodology was also tested in 2019, 2020 and 2021, when it was taught in a previous version of the KGE course, called Knowledge and Data Integration (KDI). However, significant updates changed the methodology and led to the definition of a new course (KGE) with a more appropriate evaluation. For this reason, the KDI evaluation is not reported in this work.

[6]KGE-2024 methodology questionnaire

the user's perception and the usability of the iTelos methodology. Such criteria, described here below, are evaluated by the iTelos users within the range $[1, 5]$ where 1 represents the most negative score and 5 is the most positive.

- *Efficiency*: this criterion refers to the extent to which the iTelos Methodology, as applied to the user's project, is a good solution for solving the assigned problem, thus satisfying the initial Purpose. Low-efficiency scores indicate that the methodology is not a good tool to achieve the Purpose. On the other hand, high values indicate that the iTelos methodology is an efficient solution and is likely to be used again to solve similar problems.

- *Structure*: the second criterion refers to the extent to which the iTelos Methodology has a well-organized structure or not. Low scores for this criterion indicate that the methodology was perceived as confusing, cluttered, and therefore more difficult to use. On the other hand, high scores indicate that the methodology results are well organized and more easy to use.

- *Support*: this criterion refers to the extent to which the iTelos Methodology is able to support the user in the execution of the different phases through its internal activities as well as through the tools provided. Low support scores indicate that the methodology is too broadly defined, resulting in more work for the user to interpret and implement through her own tools. On the contrary, high scores indicate a good level of support received by the user thanks to the activities and framework of the iTelos methodology.

- *Innovation*: this criterion refers to the level of innovation recognised in the iTelos methodology. This criterion indicates how much the methodology is perceived by the users as a novel solution to the data reuse, and data integration problems. Low innovation scores indicate that the methodology is perceived as a more conservative solution, and therefore more similar to existing solutions that may be preferable to this one. On the other hand, high scores indicate the novelty of the proposed solution and therefore a greater future use of the iTelos methodology.

- *Understandability*: the last criterion relates to the extent to which the iTelos methodology is easy to understand and thus learned by new users. Low scores for this criterion indicate difficulties in learning the methodology and therefore less use of it. On the other hand, high scores indicate a clear understanding of the methodology and therefore an easier solution to learn for new users.

The aim was to observe whether the overall evaluation of the methodology over the entire three-year period showed an increasing quality of the methodology from the users' point of view in relation to the selected criteria. This point is validated by observing the heat maps related to the years 2022, 2023 and 2024. Indeed, the reader can appreciate the scores shifted to the right part of the heat map, in particular for what concerns the *Support* and *Understandability* criteria. The latest version of the methodology, taught in the 2024 edition of the KGE course, received good feedback, as shown in the heat map (c) in Figure 6.1, where scores for all evaluation criteria are higher on the positive side of the heat map (right).

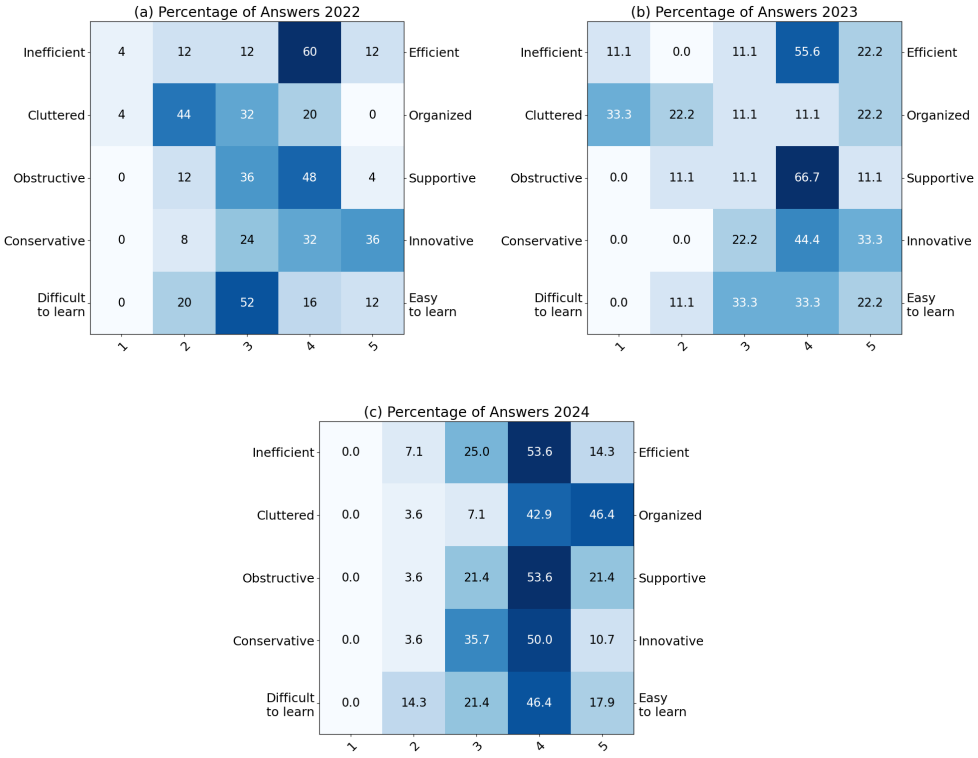


Figure 6.1: Usability of the methodology: (a) 2022, (b) 2023, (c) 2024

iTelos users were also asked to answer a number of qualitative questions as part of the questionnaire to gather more detailed feedback. Below we report some of the responses we received. The weaknesses and negative feedback about the iTelos methodology are reported first, by indicating the edition of the KGE course from which they were extracted. The strengths and positive feedback are then reported to show how the updates to the iTelos methodology improve on previous versions.

Weaknesses:

- KGE 2022 edition: "I found some steps unnecessary and only to induce confusion. I would also make the approach a bit more data-centered, meaning taking available data into account before defining the Purpose."
- KGE 2022 edition: "There were some blurred boundaries between the phases."
- KGE 2022 edition: "I'd say that a negative aspect of the methodology is that it's not straightforward and unless you do a concrete project, it's very blurry."
- KGE 2022 edition: "If you find out that something does not work you have to change things for all the previous phases, this require time for non experts."
- KGE 2022 edition: "Lack of documentation/tutorials."
- KGE 2023 edition: "A bit confusing in what each phases produces."
- KGE 2023 edition: "Some parts are really confusing to execute and need more time than others."
- KGE 2023 edition: "The methodology often requires backtracking to previous phases to fix thing and it can be annoying at times."
- KGE 2024 edition: "One negative aspect of the methodology is that certain steps, such as data collection, are highly time-consuming. Additionally, the dependency on accurate and sufficient data may pose challenges in maintaining efficiency and consistency."
- KGE 2024 edition: "Certain aspects of the methodology, such as data collection, are often highly time-consuming and, due to the nature of the project, may not yield high-quality results. This can be attributed to the insufficiency of existing resources or the challenges in extracting specific data."

The above negative feedback from KGE students over the three-year evaluation period highlights some important lessons. Firstly, it was clear that the previous versions of the methodology (2022 and 2023) were still confusing and not well structured, resulting in an inability to properly support new users who lacked the necessary expertise in KGC. Even though the methodology was designed to allow users to go back to previous phases and run them again for refinement Purposes, the structure and execution flow of the methodology were not able to support such a process, thus resulting in

annoying and time-consuming. The 2024 edition of the course, instead, revealed an aspect that has been already investigated in this research. Indeed, more the one student reported difficulties, and considerable effort to be spent in the data collection activity, by linking that to "insufficiency of existing resources". This important feedback confirms the high cost of building a KGC when reusable data is lacking, highlighting once again the importance of providing high-quality data.

Strengths

- KGE 2022 edition: "Helps to order the steps to create a complex project."
- KGE 2022 edition: "It clarifies the necessary steps for developing the KG and avoids the out-of-scope implementations."
- KGE 2022 edition: "It forces you to follow the necessary steps that you might have missed in other case without it."
- KGE 2022 edition: "Step by step progression on the project that helps with time (and possibly cost) management."
- KGE 2022 edition: "Lots of chances to correct the mistakes made."
- KGE 2022 edition: "Step by step. Some steps are easy. Reusability. Alignment."
- KGE 2023 edition: "In theory it is very innovative and important in order to improve the quality of the data used in projects, so that further reusability can be ensured."
- KGE 2023 edition: "Easier to be sure you respond at the good answer."
- KGE 2023 edition: "Having distinct phases helps with organization and use of the Purpose helps in clearly stating what the final product should provide."
- KGE 2023 edition: "Kind of clear relation between different steps of the methodology. I mean you can find why you should do some steps before more precise and prepare them for the next steps."
- KGE 2023 edition: "Learning and enriching the lexical terms."
- KGE 2023 edition: "It defines precisely the different phases required to build a kg."
- KGE 2023 edition: "The definition of the Purpose at the beginning of the methodology is a good idea which allows data engineer to do not loose focus on which will be the final objective of the Knowledge Graph."

- KGE 2023 edition: "Reusability of what you produce is one of the main request you have to consider."
- KGE 2024 edition: "Understandable without previous knowledge."
- KGE 2024 edition: "The iTelos methodology offers several positive aspects: it facilitates the creation of reusable knowledge graphs by developing independent data and schema levels, improving the reusing of prior work. Furthermore, iTelos ensures that the resulting applications are both cost-effective and reusable."
- KGE 2024 edition: "Allows simple adjustments or changes to earlier phases if needed."
- KGE 2024 edition: "Data Reuse."

The positive feedback listed above reveals some important lessons to consider. The first is that in 2022 and 2023, even if the methodology was not yet structurally suitable (as indicated by the evaluation weaknesses reported above), it was nevertheless a useful tool for developing complex projects where the order of the activities to be carried out is a crucial aspect. The backtracking process, which some of the KGE students found time-consuming, was associated with the useful possibility of correcting mistakes and thus improving the final result. Some students highlighted as a positive aspect the possibility of "learning and enriching lexical terms", thus stating the importance of the concept and terms disambiguation activity. Another important positive feedback received was about the focus given by the methodology over the Purpose. More the one student states that this approach is useful to "respond to the good answer", thus clearly stating what the final product should provide. In the 2024 edition of the KGE course, the students reported positive feedback about the understandability of the methodology for users without previous knowledge (in KGC). Last but not least, the evaluation of the methodology last year (2024) reported several positive feedbacks about the data reuse, which was recognised by the students as a crucial aspect of developing a KGC project. We think that this important result could be related to the increasing amount of data and examples (project examples, documentation, code scripts) produced by the previous course editions and distributed through the LiveData KGE catalogue. This latter result of the iTelos evaluation suggests that the methodology's approach to reducing the costs of the KGC process through data reuse is indeed working in the right direction.

7

LiveData Use Case - UniTn & NUM

Contents

This section describes the second use case related to the iTleos methodology, focused on the LiveData network (see section 5.2). More in detail, this use case describes the deployment, setup and exploitation of two LDE nodes, developed for the University of Trento (Italy) (UniTN) and the National University of Mongolia (NUM), respectively [1]. The objective of this collaboration, reported in [12], between UniTN and NUM, was to define the first two LDE nodes, as part of the LiveData network, which can be used to distribute diversity-aware data. To this end, the use case has been defined by considering the same domain of interest, namely the university research product and internal organisation. However, such a domain is considered differently in two different geographical and social contexts, an Italian and a Mongolian university. The implementation of the two LDE nodes follows the specification indicated in section 5.3 (see also Appendix A for more details). The first was to understand the effort and time required to develop new LDE nodes and connect them to the LiveData network. The second objective was to assess the extent to which the LDE nodes created were able to support the development of data-based applications and services. The following sections provide a description of the two LDE nodes that were created, as well as a description of how the current nodes of the LiveData network have been used over the last 7 months.

[1] The work described in this sections has been conducted by PhD candidate Simone Bocca, together with the supervision and support of the Professor Amarsanaa Ganbold, during a four months research period spent at the National University of Mongolia (NUM) in Ulaanbaatar (Mongolia).

7.1 LiveData UniTN

The LDE nodes created for the University of Trento (Italy), have been defined to collect and distribute diversity-aware data about:

- the UniTN courses;
- the UniTN organizations;
- the UniTN research products (like conference and journal papers);
- the UniTN staff;

To this end, we began by collecting existing data on such a domain of interest and transforming it into diversity-aware data using the iTelos KGC process. The data source considered was the digital university portal [2] described by Maltese in [72]. The data collected from such a data source was shaped as high-quality JSON datasets, but it did not meet the diversity-aware requirements indicated by the iTelos methodology. More specifically, the terms used in the JSON datasets were not disambiguated by the definition of a specific concept, as well as there was no definition of the knowledge model used to represent the entity involved in the JSON datasets. The iTelos KGC process has been adopted to transform the JSON datasets into diversity-aware data. After the execution of the process, the following diversity-aware data have been created:

- Digital University Concepts: a language dataset collecting and describing the terms used in the Digital University data of the University of Trento (Italy).
- Digital University Ontology: a knowledge model, representing the structure of the ETypes and properties considered.
- Digital University Data values datasets: the set of original JSON datasets distributed as data values datasets connected with the other diversity-aware data.
- Digital University KGs datasets: a set of KGs created for the different ETypes involved. The execution of the iTelos KGC process generated KGs for the following ETypes: UniTN courses, UniTN organizations, UniTN research papers, and UniTN staff.

After the diversity-aware data generation, the infrastructural components of the LDE node have been set up, namely the data repository and the data catalogue. The LDE node data repository has been deployed as a GitHub repository [3], and internally partitioned as indicated in section 5.3.1. The

[2] UniTN digital university
[3] LiveDataUNITN-Repository

LDE node data catalogue, called LiveData UNITN [4], has been deployed and set up according to the instructions in section 5.3.3 and in appendix A. Moreover, appropriate metadata has been defined to describe the data published in the catalogue, according to the metadata definition indicated in 5.3.3 and in appendix B. The UniTN LDE node has been included in the list of the LDE nodes linked by the LiveData gateway node (see section 5.4) in order to be accessible by the LiveData network users.

7.2 LiveData NUM

The LDE nodes created for the National University of Mongolia (Mongolia), has been defined to collect and distribute diversity-aware data about:

- the NUM courses;
- the research conferences participated by NUM;
- the curriculum & programs offered by NUM;
- the NUM research products (like conference and journal papers);
- the NUM organizations;
- the projects participated by NUM;
- NUM buildings rooms;
- NUM staff;

The key idea was to collect and transform data about the same domain of interest considered for the UniTN LDE node, with the objective of enabling the sharing of information about the university environment and, possibly, to produce data-based applications and services supporting the cooperation between the two universities.

To this end, as per the UniTN LDE node's data, we began by collecting existing data that was already provided by the Department of computer science of NUM, as JSON datasets. However, also in this case, such data did not meet the diversity-aware requirements indicated by the iTelos methodology, thus we proceeded to execute the iTelos KGC process to generate the relative diversity-aware data. The first difference we immediately recognised, with respect to the UniTN context, was the language adopted by the data within the JSON datasets. In fact, 90% of the terms were written in the Mongolian language (which uses a version of the Russian Cyrillic alphabet with the addition of two more letters). This linguistic aspect represents concretely

[4] LiveData UNITN catalogue

the diversity of the Mongolian context in which a different language must be adopted to properly describe the information entities carried by the data. In order to address this issue of linguistic heterogeneity by preserving the diversity of the linguistic layers of the Mongolian context, a domain expert is required to create a context-specific language dataset. In such a linguistic dataset, Mongolian terms are correctly associated with unique concepts and provided with a description in both Mongolian and English, thus supporting data reuse in both local and international contexts. In this case, the domain expert who supported the execution of this activity was Professor Amarsanaa Ganbold. After the execution of the iTelos KGC process, the following diversity-aware data have been created:

- Digital University Concepts: a language dataset collecting and describing the terms used in the Digital University data of the NUM. This dataset contains the terms extracted from the JSON datasets, which have been linked to specific concepts for which words and word descriptions have been provided in both Mongolian and English.
- Digital University Ontology: a knowledge model, representing the structure of the ETypes and properties considered. The knowledge model used in the Mongolian context is the same as the one used for the UniTN LDE node's data, plus some details added to represent specific information to be included in the NUM LDE node's data. Some examples of such differences are the ETypes "Room" and "Curriculum" that are present in the NUM knowledge model but not in the UniTN one.
- Digital University Data values datasets: the set of original JSON datasets distributed as data values datasets connected with the other diversity-aware data.
- Digital University KGs datasets: a set of KGs created for the different ETypes involved. The execution of the iTelos KGC process generated KGs for the following ETypes: NUM courses, NUM organizations, NUM conferences, NUM staff, NUM curriculum & programs, NUM journals, NUM research papers, NUM projects, NUM rooms.

Also for the NUM LDE node the infrastructural components of the LDE node have been set up, namely the data repository and the data catalogue. The LDE node data repository has been deployed as a GitHub repository [5], and internally partitioned with as indicated in section 5.3.1. The LDE node data catalogue, called LiveData NUM [6], has been deployed and set up according to the instructions in section 5.3.3 and in appendix A. Moreover,

[5] LiveDataNUM-Repository
[6] LiveData NUM catalogue

appropriate metadata has been defined to describe the data published in the catalogue, according to the metadata definition indicated in 5.3.3 and in appendix B. The NUM LDE node has been included in the list of the LDE nodes linked by the LiveData gateway node (see section 5.4) in order to be accessible by the LiveData network users.

7.3 LiveData usages & Lesson learned

The set up of the first two nodes of the LiveData network was crucial to understanding the criticalities and the features of the infrastructure as it as been thought. This use case elaborated across two different contexts, the UniTN one and the NUM one, allows for the first evaluation of the LiveData network infrastructure. More in detail the two LDE nodes have been used in two different moments described here below.

- *Mongolian Hackathon*: the NUM LDE node has been used as a data source for the hackathon organized by the Hackum Students Club [7]. They are NUM students and ex-students interested in the development of new digital ideas to support university life, as well as the research and innovation community composed of ex-students of NUM. The hackathon, organized in May 2024, involved different groups of students who worked for two days on the development of digital web services for the support of the NUM students. During the initial hackathon introduction, the LiveData network has been presented with the objective of promoting the diversity-aware approach of the iTelos methodology, for data generation, reuse and sharing.

- *KGE 2024 course project*: one of the project proposed to the students of the 2024 edition of the KGE course was described by the following Purpose:

  > *The Purpose of this project is to engineer a Knowledge Graph able to support applications and services providing information about research collaboration (published paper, project or others) between the University of Trento and the National University of Mongolia, describing also the different locations of the two university facilities.*

  Both the UniTN and the NUM LDE nodes have been provided as input data sources for the development of this project.

Another use of the LiveData infrastructure worth mentioning is the catalogue that collects the KGE projects along the different editions of the course. Different students, in the evaluation questionnaire for the 2024 edition of

[7] Hackum Students Club

the course, mentioned the support provided by the KGE catalogue for the collection of data required for their projects. The data distributed by such a catalogue has already been processed by the students of the previous edition of the course, following the iTelos methodology, thus considerably simplifying the work of the students of the 2024 edition of the course.
From the exploitation of the LDE nodes available in the LiveData network in the cases reported above, we learnt very important insights, allowing us to evaluate the LiveData infrastructure, and more, in general, the iTelos data distribution approach. Below we report on the key findings of this evaluation.

- The first, and maybe most important, aspect we discovered while developing the LiveData network, was the need to be focused on the local context, or LDE. This focus does not only involve the data to be generated or exploited but also the social and geographical aspects of the context itself. In order to produce useful data in a specific context, it is necessary to understand which are the needs of the people living in such a context, who have to interact with the applications and digital services supported by the data to be produced. Only in this way it will be possible to generate data suitable to represent the (diversity of) context, and the people who live there, data that can be a concrete solution for the problem of those people. To this end, it is important to remember that the LiveData infrastructure, as well as the entire iTelos methodology, are instruments to be put in the hands of local people, who are experts in their own geographical and social context. The iTelos methodology must not be used in a centralised way, by people without the knowledge of the context for which they intend to produce applications and digital services. Such a situation will only lead to a manipulation of the information about the relative context, thus distorting the reality and losing the diversity that characterises the context itself.

- As a direct consequence of the above aspect, we noticed how a different effort could be considered for the data generation, reuse and distribution when the iTelos methodology is applied in different contexts. For example, the Mongolian case reported above describes a situation where the language diversity of the data needs to be preserved by using a natural language (Mongolian Cyrillic) that is not so commonly adopted to represent interoperable data. This situation required more effort to create a language dataset in which the terms are described in both Mongolian and English so that the diversity-aware data using such terminology would be more interoperable at an international level.

- During the events in which the LiveData network and its nodes have been used, we recognized a very important aspect that has been taken

into account for the definition of the overall iTelos methodology. We refer to the role of the iTelos DI, as a key role to be considered both locally and as a central role. By discussing with local domain experts, we recognized how the local action of the iTelos DI is fundamental to filling the gap between the data producer and the data consumers. In particular, during the research period spent in Ulaanbaatar (Mongolia), we discovered that the local government is strongly seeking a digitisation process of social services to support citizens through mobile and web applications based on data about people, integrated with social and geographical data representing the urban context of Ulaanbaatar. However, the local government is currently missing experts in data management and application development, thus conveying such efforts to few data producers and digital service providers. The local experts told us about the need for an infrastructure for the distribution of data about the local context, together with the presence of an intermediary actor who is able to dialogue with both the producer of such data and those who wish to exploit it, by protecting sensible data and preserving diversity without distorting information. On the other hand, centralised action by the iTelos DI is essential to ensure connectivity between the various LDE nodes of the LiveData network and to enhance interoperability and reuse of the diversity-aware data shared between LiveData users.

- Moreover, during the LiveData network testing we recognized an important feature of the iTelos data distribution network. In fact, each node of the LiveData network share information regarding its relative LDE, by distributing diversity-aware data. Therefore, by navigating the LiveData network it is possible to explore the diversity of each LDE. Such exploration highlights both the differences, which constitute the uniqueness of each LDE, but also the common aspects between two or more LDEs. The LiveData network plays a key role in discovering the unity between different contexts, as well as between the people living in those contexts, by defining their local diversity. Such unity is the fundamental ingredient for the development of interaction and cooperation among people living all over the world. An example of this unity is the Purpose behind the digital university project of the University of Trento (Italy), which has been recognized as a need to be implemented also in the National University of Mongolia, thanks to the data distributed by the Mongolian LDE node.

8 Conclusion

Contents

The research work of this thesis examined the evolution of data representation and use over the past years, in order to better understand the current needs in the use of data and the information it contains. We have studied how the use of data has changed over time, becoming more subjective thanks to its availability to an increasing number of people. If in the past, the ability to use data was in the hands of a few actors (big companies), nowadays it is in everyone's hands, thanks to the Web, smartphones, mobile applications, AI services, and other technologies. This data spread changed the way in which the data is represented and perceived. In the past, the data was produced to satisfy a unique Purpose, within a restricted context. Instead, data reuse plays a central role in the development of new applications and services, with the result that the primary purpose for which the data was created is not the only one it must serve. Secondary purposes need to be satisfied by data collected from and created for, different geographical, social or even personal (individual) contexts. To this end, data reuse involves an adaptation of the data for different purposes. Although such reuse reduces the cost of application development by saving the effort, time and money involved in the process of creating data from scratch, on the other hand, the diversity of information carried by the original data, expressing the specificity of the local context for which the data was created, is lost in the process of adapting the data to new purposes. The impact of such an information loss is even more dangerous when the data is reused to train AI models. Several AI solutions (applications and services) are based on the adaptation of the data, collected from different local contexts to a unique, usually Western and business-oriented, data model. This leads to distortion of the reality as it is represented by such AI applications, due to the loss of that key information diversity which was able, in the original data, to represent a specific context. The risk is that we will be (or perhaps already are) supported by tools that

have a single interpretation of reality, based on a centralised knowledge that cannot properly represent the culture, society, and geographical aspects of a foreign context, as well as the daily life of people living in such a context. It is not difficult to think about the enormous inequalities that this approach will produce. Inevitably, the richest capital-intensive models, privately owned (e.g. Google's PaLM, DeepMind's Gopher, OpenAI's DALLE-2), will be the only ones able to handle and exploit the knowledge produced by the data reuse. As a consequence, the knowledge representation will be in their hand, under the unique perspective of their "point of view", most of the time focused on business, or in any case centralized into a unique culture (usually the Western one). Moreover, such a situation will lead to an economic concentration in favour of the above mentioned actors.

We think that to avoid such a dangerous situation, the solution is based on a decentralized approach, where the data and the relative information are handled by the actors having the appropriate knowledge about the local contexts for which the data is created. However, this solution is achievable only if such local actors are supported by instruments allowing them to produce and share data that can concretely represent their information diversity. With this objective we propose, in this research work, iTelos, a methodology for data reuse based on the production, integration and distribution of diversity-aware Knowledge Graphs (KGs). iTelos aims to support the generation, integration, and distribution of local diversity-aware data, with the major objective of improving the quality and reducing the cost of data reuse. To this end, iTelos is composed of three processes. The data representation process is the first iTelos process, and it provides the base principles for the creation of diversity-aware data. The novelty introduced by this process is the possibility of using it to transform existing data into a novel type of graph-based data where the information diversity is concretely represented as three different KGs connected to each other. More in detail, the three KGs represent the information diversity at three different layers: language, knowledge (or data schema), and data values. Additionally, the three KG layers are associated with a fourth one representing concretely the Purpose for which the data has been created. Such a new type of data allows for more interoperability, being the information diversity clearly understandable from the data itself, without implicitly defined details. The second iTelos process is a KG Completion (KGC) process. This process is focused on the integration of data both diversity-aware and non-diversity-aware. The key idea is that the iTelos KGC process supports both the users interested in creating new diversity-aware data, by integrating existing non-diversity-ware data, as well as users interested in integrating already created diversity-aware data to fulfil Purpose-specific application requirements. To this end, the iTelos KGC process encapsulates the iTelos data representation process providing the guidelines for the data transformation. The last iTelos

process is a data distribution process which, in turn, encapsulates the iTelos KGC process. The iTelos data distribution process aims to enable the reuse of the diversity-aware data produced by the previous two processes. It is based on a data distribution network called LiveData. Each LiveData node is created to support a specific local context, and allows the users navigating the network, to discover the diversity-aware data produced for such a context. Moreover, each LiveData node allows the users to access and use the iTelos KGC process, to eventually create new diversity-aware data or integrate the diversity-aware data already available. A key feature introduced by the iTelos data distribution process, thanks to the LiveData network, is the possibility of creating new collaboration between the users who have the expertise of creating data (data producers) and the users having a specific Purpose for the development of data-based applications and services (data consumers), but often lack of the required expertise to get the data required to implement it.

The main difference with respect to the centralised approach adopted by large AI companies today is that the data is produced and remains in the hands of those who have the necessary knowledge to handle it. In this case, the local actors are responsible for the use of the data, and therefore for collaborating with external actors (concerning their context), to increase innovation and research purposes. In this approach, what is required is that the tools for generating and distributing the data are freely available to the local actors, who not only have the right knowledge of the local context but also a clear understanding of the needs to be satisfied locally so that they can lead the production of data and the development of new services for their own society.

8.1 Limitations

The research presented in this thesis has some limitations that could not be covered during the study period. In this section we have reported on such limitations with the aim of stimulating further work on this broad research topic.

The first, and perhaps most significant, limitation is that, as iTelos is a methodology based on data reuse and sharing, its impact can only be more accurately assessed by considering a large-scale application of the methodology itself. As demonstrated in chapter 6, students in the KGE course benefited from the reuse of diversity-aware data previously produced by students in previous years of the course. However, the iTelos methodology has only been tested within the KGE context and no evaluation has been carried out outside the University of Trento (Italy). For this reason, it is not

possible to declare that iTelos can be applied in practice, based only on the current research. A more comprehensive evaluation is needed, including the adoption of iTelos in real use cases outside the University of Trento (Italy).

The second limitation is related to the iTelos approach to solving the problem of data reuse. In fact, the proposed methodology is based on the key aspect of data sharing. The more data produced with iTelos is shared, the more the LiveData network will be able to support its users in collecting diversity-aware data and, consequently, to support iTelos users who can benefit from the cost reduction of producing new data by reusing the high-quality KGs previously produced. This means that, especially in the initial bootstrap period when iTelos is being adopted, users need to trust the process and agree to share their data to support future re-use. Such confidence in the process is not a detail, as it can only come from a clear understanding of the motivations underlying the iTelos methodology, which is to promote the use of quality data capable of representing the diversity of local information, rather than distortions of reality produced by users who do not have the necessary knowledge to represent it. For this reason, we believe that education about data representation for AI, where iTelos is proposed as a novel methodology, is fundamental to providing people with the above-mentioned understanding of the iTelos principles.

The third limitation is related to the framework that supports the use of the iTelos methodology. In fact, the current version of the methodology integrates the use of existing tools to implement the activities required in each phase (for more details see chapter 4). However, the methodology, in its current version, lacks a dedicated framework to support users along the execution of all phases. This is a limitation for two main reasons; (i) the first is that the tools currently used are not designed to work with the EB data model proposed by the iTelos methodology. As a consequence, users have to use the tools provided in a specific way with the aim of linking the different components of the EB they are building. This may result in a decomposed version of the EB, where the different information layers are more difficult to explore. (ii) The second limitation, due to the lack of a specific iTelos framework, is the support for non-expert users. In fact, the use of the tool currently provided by the methodology requires the expertise of knowledge engineers and data scientists to support a domain expert who (often) does not have such technical skills.

8.2 Future work

The research work reported in this thesis paves the way for several future works towards several objectives concerning all the three processes compos-

ing the iTleos methodology. We report here below, more details about the planned future work towards the next version of the iTelos methodology.

- *Collection of standard reference knowledge.* The iTelos data representation process defines how to transform existing data into diversity-aware data by building an EB. For this purpose, as described in section 3.3.3, reference domain knowledge (ontologies and data schema) is considered for modelling the ETG (EB knowledge component). Such domain-specific knowledge is fundamental to ensure the reusability of the diversity-aware data produced by iTelos. If the structure of such data has been modelled by considering well-known knowledge standards in the respective domain, the data itself will be more reusable in this specific domain. As a consequence, the higher the number of domains covered by at least one reference knowledge model provided by iTelos, the more possibilities there will be for iTelos to be applied in different domains, thus producing a large variety of diversity-aware data. To this end, our next step in this direction is to work on the identification and modelling of new reference domain knowledge, for other different domains, to be provided as support of the data representation process. This will require a detailed analysis of the ontologies and data schemas modelled for different domains, with the aim of finding those that are widely adopted and well-modelled. The reference knowledge models collected will be then accessible through the LiveKnowledge data catalogue as part of the LiveData distribution network.

- *iTelos KGC tools and automation.* The current version of the iTelos KGC process is already supported by a framework that includes tools to support the different activities involved in the process. However, the goal for the future version of the methodology is to increase the level of automation of the entire KGC process to support non-expert users in performing all phases of the process. During the KGE course, in which the methodology was tested by the Master's students, we provided the students with a second questionnaire, to be completed at the end of each edition of the course, to provide feedback on the tools used throughout the process. The answers collected showed that some instruments are not good enough to properly support the activity to be executed. For example, the list of concepts defined in the second phase (see section 4.3), which will then be aligned with the UKC, is currently defined as a spreadsheet. One of the future works planned for iTelos is a dedicated tool that would directly integrate access to the UKC, which would speed up the execution of such an activity. In addition, several students reported difficulties in modelling the ER diagram of the Purpose in the first phase (see section 4.2), due to the fact that the ER model is a modelling paradigm designed for relational DB and not

for graph-based data. In the next version of the iTelos methodology, we plan to explore and test new models to shape the purpose knowledge in the first phase of the iTelos KGC process. Last but not least, many students in different editions of the course reported that the use of the Karmalinker tool in the final phase of the KGC process (see section 4.5) is not trivial and intuitive. As such a tool is a de facto standard for data mapping activities, we do not intend to replace it, but as part of the future work planned for iTelos, we intend to update the Karmalinker tool to make it easier to use.

- *LiveData nodes update.* Regarding the iTelos data distribution process, future work includes several updates both at the individual node level and at the network level. The aim is to implement each node as a local platform capable of supporting all the services provided by the iTleos methodology. A key role in the LiveData node architecture is played by the data catalogues. For this reason, we intend to update the catalogues by adding new search functionalities based on specific criteria to allow users to easily access the data they are looking for. We also plan to update the LiveData catalogues with dedicated API layers for automatic uploading, downloading and publishing of diversity-aware data and its descriptive metadata. Dedicated web pages describing the data and the project that produced it will be added as part of the next version of the catalogue design. At the network level, the next version of the LiveData architecture will include a dedicated protocol for communication between nodes, allowing for more automated data retrieval and distribution.

- *Educational goals.* The KGE course, described in chapter 6, is not only a way to test the methodology, but it is also an important generator of educational data that can be used to provide real-word-like experiences to the students, thus allowing them to learn better the topics addressed by the iTelos methodology. Moreover, the course is fundamental not only for the students of the University of Trento but also for each local user who intends to create and exploit diversity-aware data for her own Purpose, in her local context, to satisfy local needs. The teaching of the iTleos methodology is the first step towards the migration from the centralized approach in data reuse for the development of applications and services, to a decentralized approach able to preserve local information diversity and to promote a fairer and more distributed economy.

- *iTelos glossary.* As the iTelos methodology is a complex research topic involving different research areas, we are working on a dedicated glossary to collect the most important terms used to describe the methodology. A preliminary version of the iTelos glossary was initially attached

at the end of this thesis. However, we decided to not include that, because it needs to be refined by defining properly the terminology used to describe the methodology and by improving the quality of the definition of each term. We are currently working on such updates with the aim of providing the first stable version of the iTelos glossary as soon as possible.

Bibliography

[1] Renzo Angles and Claudio Gutierrez. "Survey of graph database models". In: *ACM Computing Surveys (CSUR)* 40.1 (2008), pp. 1–39 (cit. on p. 25).

[2] Majid Asgari-Bidhendi, Ali Hadian, and Behrouz Minaei-Bidgoli. "Farsbase: The persian knowledge graph". In: *Semantic Web* 10.6 (2019), pp. 1169–1196 (cit. on p. 28).

[3] Sören Auer, Christian Bizer, et al. "Dbpedia: A nucleus for a web of open data". In: *international semantic web conference*. Springer. 2007, pp. 722–735 (cit. on p. 25).

[4] G. Mihaila B. Golshan A. Halevy and W. Tan. "Data Integration: After the Teenage Years". In: *36th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database SystemsMay 2017 Pages 101–106h.* 2017 (cit. on p. 28).

[5] Mayukh Bagchi et al. "Language and Knowledge Representation A Stratified Approach". In: (2025) (cit. on p. 45).

[6] Gábor Bella, Linda Gremes, and Fausto Giunchiglia. "Exploring the language of data". In: *Proceedings of the 28th International Conference on Computational Linguistics*. 2020, pp. 6638–6648 (cit. on p. 19).

[7] Tim Berners-Lee et al. *Semantic web road map*. 1998 (cit. on pp. 4, 21).

[8] Ceri Binding and Douglas Tudhope. "Improving interoperability using vocabulary linked data". In: *International Journal on Digital Libraries* 17.1 (2016), pp. 5–21 (cit. on p. 20).

[9] Libby Bishop. "Using archived qualitative data for teaching: Practical and ethical considerations". In: *International Journal of Social Research Methodology* 15.4 (2012), pp. 341–350 (cit. on p. 17).

[10] Simone Bocca, Gábor Bella, and Fausto Giunchiglia. "Interoperating Vocabularies in Multilingual Domain Data". In: *2022 IEEE 24th Conference on Business Informatics (CBI)*. Vol. 2. IEEE. 2022, pp. 73–79 (cit. on pp. 19, 50).

[11] Simone Bocca, Mauro Dragoni, and Fausto Giunchiglia. "iTelos - Case studies in building domain specific knowledge graphs". In: *ISIC-22 International Semantic Intelligence Conference*. 2022 (cit. on pp. 11, 159).

[12] Simone Bocca, Amarsanaa Ganbold, and Tsolmon Zundui. "LiveData– A Worldwide Data Mesh for Stratified Data". In: *arXiv preprint arXiv:2407.00036* (2024) (cit. on pp. 13, 142, 170).

[13] Simone Bocca, Alessio Zamboni, et al. "Building Interoperable Electronic Health Records as Purpose Driven Knowledge Graphs". In: *Data Science and Artificial Intelligence for Digital Healthcare: Communications Technologies for Epidemic Models*. Springer, 2024, pp. 237–254 (cit. on pp. 8, 11).

[14] Kurt Bollacker, Colin Evans, et al. "Freebase: a collaboratively created graph database for structuring human knowledge". In: *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. 2008, pp. 1247–1250 (cit. on p. 25).

[15] Matteo Busso et al. "The iLog methodology for fostering valid and reliable Big Thick Data". In: (2024) (cit. on p. 71).

[16] Michel Callon and John Law. "After the individual in society: Lessons on collectivity from science, technology and society". In: *Canadian Journal of Sociology/Cahiers canadiens de sociologie* (1997), pp. 165–182 (cit. on p. 125).

[17] Chen Chen, Behzad Golshan, et al. "BigGorilla: An Open-Source Ecosystem for Data Preparation and Integration." In: *IEEE Data Eng. Bull.* 41.2 (2018), pp. 10–22 (cit. on p. 28).

[18] Peter Pin-Shan Chen. "The entity-relationship model—toward a unified view of data". In: *ACM transactions on database systems (TODS)* 1.1 (1976), pp. 9–36 (cit. on p. 39).

[19] Edgar F Codd. "A relational model of data for large shared data banks". In: *Communications of the ACM* 13.6 (1970), pp. 377–387 (cit. on p. 38).

[20] European Commission and PwC EU Services. *Cost-benefit analysis for FAIR research data: Cost of not having FAIR research data*. 2018 (cit. on p. 24).

[21] Randall Davis, Howard Shrobe, and Peter Szolovits. "What is a knowledge representation?" In: *AI magazine* 14.1 (1993), pp. 17–17 (cit. on pp. 4, 21).

[22] Umeshwar Dayal, Malu Castellanos, Alkis Simitsis, and Kevin Wilkinson. "Data integration flows for business intelligence". In: *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*. 2009, pp. 1–11 (cit. on p. 18).

[23] Zhamak Dehghani. *Data Mesh*. Marcombo, 2022 (cit. on p. 125).

[24] James M DuBois, Michelle Strait, and Heidi Walsh. "Is it time to share qualitative research data?" In: *Qualitative psychology* 5.3 (2018), p. 380 (cit. on p. 17).

[25] Lisa Ehrlinger and Wolfram Wöß. "Towards a definition of knowledge graphs." In: *SEMANTiCS (Posters, Demos, SuCCESS)* 48.1-4 (2016), p. 2 (cit. on pp. 25, 26).

[26] Fajar Ekaputra, Marta Sabou, et al. "Ontology-based data integration in multi-disciplinary engineering environments: A review". In: *Open Journal of Information Systems* 4.1 (2017), pp. 1–26 (cit. on p. 27).

[27] Eimear Farrell, Marco Minghini, et al. *European Data Spaces-Scientific Insights into Data Sharing and Utilisation at Scale*. Tech. rep. Joint Research Centre (Seville site), 2023 (cit. on p. 33).

[28] Dieter Fensel, Umutcan Şimşek, et al. "Introduction: what is a knowledge graph?" In: *Knowledge Graphs*. Springer, 2020, pp. 1–10 (cit. on p. 29).

[29] Mariano Fernández-López. "Overview of methodologies for building ontologies". In: (1999) (cit. on p. 23).

[30] Mariano Fernández-López, Asunción Gómez-Pérez, and Natalia Juristo. "Methontology: from ontological art towards ontological engineering". In: (1997) (cit. on p. 23).

[31] Tim Finin, Richard Fritzson, Don McKay, and Robin McEntire. "KQML as an agent communication language". In: *Proceedings of the third international conference on Information and knowledge management*. 1994, pp. 456–463 (cit. on p. 151).

[32] Mattia Fumagalli, Daqian Shi, and Fausto Giunchiglia. "Ranking schemas by focus: A cognitively-inspired approach". In: *Graph-Based Representation and Reasoning: 26th International Conference on Conceptual Structures, ICCS 2021, Virtual Event, September 20–22, 2021, Proceedings 26*. Springer. 2021, pp. 73–88 (cit. on p. 72).

[33] Aldo Gangemi, Nicola Guarino, Claudio Masolo, and Alessandro Oltramari. "Understanding top-level ontological distinctions." In: *OIS@ IJCAI* 47 (2001) (cit. on p. 22).

[34] Fausto Giunchiglia et al. "Contextual reasoning". In: *Epistemologia, special issue on I Linguaggi e le Macchine* 16 (1993), pp. 345–364 (cit. on p. 23).

[35] Fausto Giunchiglia, Khuyagbaatar Batsuren, and Gabor Bella. "Understanding and Exploiting Language Diversity". In: *Proceedings of the 26th Int. Joint Conference on Artificial Intelligence (IJCAI-17)*. 2017, pp. 4009–4017 (cit. on p. 21).

[36] Fausto Giunchiglia, Khuyagbaatar Batsuren, and Abed Freihat. "One World - Seven Thousand Languages". In: *19th International Conference on Computational Linguistics and Intelligent Text Processing*. Hanoi, Vietnam, 2018 (cit. on p. 21).

[37] Fausto Giunchiglia, Simone Bocca, et al. "iTelos–Purpose Driven Knowledge Graph Generation". In: *arXiv preprint arXiv:2105.09418* (2021) (cit. on pp. 11, 159).

[38] Fausto Giunchiglia, Simone Bocca, et al. "iTelos-Building reusable knowledge graphs". In: *arXiv preprint arXiv:2105.09418* (2021) (cit. on pp. 11, 29).

[39] Fausto Giunchiglia, Simone Bocca, et al. "Popularity driven data integration". In: *Iberoamerican Knowledge Graphs and Semantic Web Conference*. Springer. 2022, pp. 277–284 (cit. on p. 11).

[40] Fausto Giunchiglia and Biswanath Dutta. "DERA: A faceted knowledge organization framework". In: (2011) (cit. on p. 69).

[41] Fausto Giunchiglia, Mattia Fumagalli, et al. "Entity Type Recognition-Dealing with the Diversity of Knowledge." In: *KR*. 2020, pp. 414–423 (cit. on pp. 52, 109).

[42] Fausto Giunchiglia and Mattia Fumagalli. "Teleologies: Objects, actions and functions". In: *International conference on conceptual modeling*. Springer. 2017, pp. 520–534 (cit. on p. 23).

[43] Fausto Giunchiglia and Luciano Serafini. "Multilanguage hierarchical logics, or: how we can do without modal logics". In: *Artificial intelligence* 65.1 (1994), pp. 29–70 (cit. on p. 23).

[44] Fausto Giunchiglia and Daqian Shi. "Property-based entity type graph matching". In: *arXiv preprint arXiv:2109.09140* (2021) (cit. on p. 7).

[45] Fausto Giunchiglia, Alessio Zamboni, Mayukh Bagchi, and Simone Bocca. "Stratified data integration". In: *arXiv preprint arXiv:2105.09432* (2021) (cit. on pp. 12, 19, 28, 142).

[46] Barney G Glaser. "Retreading research materials: The use of secondary analysis by the independent researcher". In: *American Behavioral Scientist* 6.10 (1963), pp. 11–14 (cit. on p. 17).

[47] Barney G Glaser. "Secondary analysis: A strategy for the use of knowledge from research elsewhere". In: *Soc. Probs.* 10 (1962), p. 70 (cit. on p. 17).

[48] Michael Grüninger and Mark S Fox. "The role of competency questions in enterprise engineering". In: *Benchmarking—Theory and practice*. Springer, 1995, pp. 22–31 (cit. on p. 68).

[49] Nicola Guarino, Daniel Oberle, and Steffen Staab. "What is an ontology?" In: *Handbook on ontologies* (2009), pp. 1–17 (cit. on p. 22).

[50] Shubham Gupta, Pedro Szekely, et al. "Karma: A system for mapping structured sources into the semantic web". In: *Extended Semantic Web Conference*. Springer. 2012, pp. 430–434 (cit. on p. 56).

[51] Stephen C Guptill. "Metadata and data catalogues". In: *Geographical information systems* 2 (1999), pp. 677–692 (cit. on p. 33).

[52] Mordechai Haklay and Patrick Weber. "Openstreetmap: User-generated street maps". In: *IEEE Pervasive computing* 7.4 (2008), pp. 12–18 (cit. on p. 24).

[53] Pamela S Hinds, Doris E Chaves, and Sandra M Cypess. "Context as a source of meaning and understanding". In: *Qualitative health research* 2.1 (1992), pp. 61–74 (cit. on p. 17).

[54] Mohammad Alamgir Hossain, Yogesh K Dwivedi, and Nripendra P Rana. "State-of-the-art in open data research: Insights from existing literature and a research agenda". In: *Journal of organizational computing and electronic commerce* 26.1-2 (2016), pp. 14–40 (cit. on p. 30).

[55] Saffron Huang and Divya Siddarth. "Generative AI and the digital commons". In: *arXiv preprint arXiv:2303.11074* (2023) (cit. on pp. 10, 17, 18).

[56] Richard Hull. "Managing semantic heterogeneity in databases: a theoretical prospective". In: *Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*. 1997, pp. 51–61 (cit. on p. 38).

[57] David Huynh and Stefano Mazzocchi. *OpenRefine*. 2012 (cit. on p. 29).

[58] WH Inmon. "Building the Data Warehouse. A Wiley QED publication". In: *WIley* 8 (1993) (cit. on p. 18).

[59] Marijn Janssen, Yannis Charalabidis, and Anneke Zuiderwijk. "Benefits, adoption barriers and myths of open data and open government". In: *Information systems management* 29.4 (2012), pp. 258–268 (cit. on pp. 30, 31).

[60] Marijn Janssen and Anneke Zuiderwijk. "Infomediary business models for connecting open data providers and users". In: *Social Science Computer Review* 32.5 (2014), pp. 694–711 (cit. on p. 32).

[61] Thorhildur Jetzek, Michel Avital, and Niels Bjorn-Andersen. "Data-driven innovation through open government data". In: *Journal of theoretical and applied electronic commerce research* 9.2 (2014), pp. 100–120 (cit. on p. 30).

[62] William Jones, Jesse David Dinneen, et al. "Personal information management (PIM)". In: *Encyclopedia of library and information science* (2017), pp. 3584–605 (cit. on p. 31).

[63] Petar Jovanovic, Sergi Nadal, et al. "Quarry: a user-centered big data integration platform". In: *Information Systems Frontiers* 23.1 (2021), pp. 9–33 (cit. on p. 29).

[64] Tahir Emre Kalaycı, Bor Bricelj, et al. "A knowledge graph-based data integration framework applied to battery data management". In: *Sustainability* 13.3 (2021), p. 1583 (cit. on p. 28).

[65] Athanasios Kiourtis, Argyro Mavrogiorgou, et al. "Electronic health records at People's hands across Europe: The InteropEHRate protocols". In: *pHealth 2022*. IOS Press, 2022, pp. 145–150 (cit. on p. 8).

[66] Craig A Knoblock and Pedro Szekely. "Exploiting semantics for big data integration". In: *Ai Magazine* 36.1 (2015), pp. 25–38 (cit. on pp. 29, 104).

[67] Sebastian Krause, Leonhard Hennig, et al. "Sar-graphs: A language resource connecting linguistic knowledge with semantic relations from knowledge graphs". In: *Journal of Web Semantics* 37 (2016), pp. 112–131 (cit. on p. 26).

[68] Erik Lakomaa and Jan Kallberg. "Open data as a foundation for innovation: The enabling effect of free public sector information for entrepreneurs". In: *IEEE Access* 1 (2013), pp. 558–563 (cit. on p. 30).

[69] Deryle Lonsdale, David W Embley, et al. "Reusing ontologies and language components for ontology generation". In: *Data & Knowledge Engineering* 69.4 (2010), pp. 318–330 (cit. on p. 7).

[70] Shane Loughlin. "Industry 3.0 to Industry 4.0: Exploring the transition". In: *New Trends in Industrial Automation*. IntechOpen, 2018 (cit. on p. 5).

[71] Inês Araújo Machado, Carlos Costa, and Maribel Yasmina Santos. "Data mesh: concepts and principles of a paradigm shift in data architectures". In: *Procedia Computer Science* 196 (2022), pp. 263–271 (cit. on p. 125).

[72] Vincenzo Maltese. "Digital transformation challenges for universities: Ensuring information consistency across digital services". In: *Cataloging & classification quarterly* 56.7 (2018), pp. 592–606 (cit. on pp. 30, 171).

[73] Sara Mannheimer. *Scaling Up: How Data Curation Can Help Address Key Issues in Qualitative Data Reuse and Big Social Research*. Springer, 2024 (cit. on p. 17).

[74] John McCarthy and Sasa Buvac. "Formalizing context". In: *AAAI Fall symposium on context in knowledge representation*. Citeseer. 1994, pp. 99–135 (cit. on p. 23).

[75] Connor McMahon, Isaac Johnson, and Brent Hecht. "The substantial interdependence of Wikipedia and Google: A case study on the relationship between peer production communities and information technologies". In: *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 11. 1. 2017, pp. 142–151 (cit. on p. 18).

[76] Marina Micheli, Eimear Farrell, et al. *Mapping the landscape of data intermediaries*. 2023 (cit. on p. 13).

[77] Peter Mika, Abraham Bernstein, et al. *The Semantic Web–ISWC 2014: 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part II*. Vol. 8797. Springer, 2014 (cit. on p. 26).

[78] George A Miller, Richard Beckwith, et al. "Introduction to WordNet: An on-line lexical database". In: *International journal of lexicography* 3.4 (1990), pp. 235–244 (cit. on p. 20).

[79] Ruth Garrett Millikan. *Language, thought, and other biological categories: New foundations for realism*. MIT press, 1987 (cit. on p. 20).

[80] Ian Niles and Adam Pease. "Towards a standard upper ontology". In: *Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001*. 2001, pp. 2–9 (cit. on p. 22).

[81] Marcelo Iury S. Oliveira and Bernadette Farias Lóscio. "What is a data ecosystem?" In: *Proceedings of the 19th Annual International Conference on Digital Government Research: Governance in the Data Age*. dg.o '18. Delft, The Netherlands: Association for Computing Machinery, 2018. ISBN: 9781450365260. DOI: 10.1145/3209281.3209335. URL: https://doi.org/10.1145/3209281.3209335 (cit. on p. 32).

[82] Irene V Pasquetto, Christine L Borgman, and Morgan F Wofford. "Uses and reuses of scientific data: The data creators' advantage". In: (2019) (cit. on p. 17).

[83] Irene V Pasquetto, Bernadette M Randles, and Christine L Borgman. "On the reuse of scientific data". In: (2017) (cit. on p. 18).

[84] Heiko Paulheim. "Knowledge graph refinement: A survey of approaches and evaluation methods". In: *Semantic web* 8.3 (2017), pp. 489–508 (cit. on pp. 25, 26).

[85] Liubov Pilshchikova, Anneke Zuiderwijk, and Marijn Janssen. "Non-profit organisations' capabilities to reduce open government data barriers: a conceptual framework". In: (2024) (cit. on p. 32).

[86] A Poikola, P Kola, and KA Hintikka. "An introduction to opening information sources, Ministry of transport and communication". In: *Helsinki, Finland* (2011) (cit. on p. 125).

[87] Hemant Purohit, Rajaraman Kanagasabai, and Nikhil Deshpande. "Towards next generation knowledge graphs for disaster management". In: *2019 IEEE 13th international conference on semantic computing (ICSC)*. IEEE. 2019, pp. 474–477 (cit. on p. 28).

[88] Marcelo Iury S. Oliveira, Glória de Fátima Barros Lima, and Bernadette Farias Lóscio. "Investigations into data ecosystems: a systematic mapping study". In: *Knowledge and information systems* 61 (2019), pp. 589–630 (cit. on p. 31).

[89] Julia Christina Schweihoff, Ilka Jussen, and Frederik Möller. "Trust me, I'm an intermediary! Exploring data intermediation services". In: (2023) (cit. on p. 31).

[90] Rolf Schwitter. "Controlled natural languages for knowledge representation". In: *Coling 2010: Posters*. 2010, pp. 1113–1121 (cit. on pp. 20, 48).

[91] Md Hanif Seddiqui, Feroz Farazi, et al. "A Material Passport Ontology for a Circular Economy". In: *Available at SSRN 4993406* () (cit. on p. 23).

[92] Ashraf Shaharudin, Bastiaan van Loenen, and Marijn Janssen. "Exploring the contributions of open data intermediaries for a sustainable open data ecosystem". In: *Data & Policy* 6 (2024), e56 (cit. on pp. 32, 125).

[93] Andrea Tagarelli, Mario Longo, and Sergio Greco. "Word sense disambiguation for XML structure feature generation". In: *European Semantic Web Conference*. Springer. 2009, pp. 143–157 (cit. on p. 19).

[94] Joe Tekli. "An overview on xml semantic disambiguation from unstructured text to semi-structured data: Background, applications, and ongoing challenges". In: *IEEE Transactions on Knowledge and Data Engineering* 28.6 (2016), pp. 1383–1407 (cit. on p. 19).

[95] Denny Vrandečić and Markus Krötzsch. "Wikidata: a free collaborative knowledgebase". In: *Communications of the ACM* 57.10 (2014), pp. 78–85 (cit. on p. 26).

[96] Holger Wache, Thomas Voegele, et al. "Ontology-based integration of information-a survey of existing approaches". In: *Ois@ ijcai*. 2001 (cit. on p. 27).

[97] Xiaoguang Wang, Qingyu Duan, and Mengli Liang. "Understanding the process of data reuse: An extensive review". In: *Journal of the Association for Information Science and Technology* 72.9 (2021), pp. 1161–1182 (cit. on p. 18).

[98] Terry Winograd. "Understanding natural language". In: *Cognitive psychology* 3.1 (1972), pp. 1–191 (cit. on p. 19).

[99] Wayne Xin Zhao, Kun Zhou, et al. "A survey of large language models". In: *arXiv preprint arXiv:2303.18223* (2023) (cit. on p. 10).

[100] Yuanwei Zhao, Lan Huang, et al. "Is your Schema Good Enough to Answer my Query?" In: *arXiv preprint arXiv:2103.07703* (2021) (cit. on p. 109).

[101] Shuangjia Zheng, Jiahua Rao, et al. "PharmKG: a dedicated knowledge graph benchmark for bomedical data mining". In: *Briefings in bioinformatics* 22.4 (2021), bbaa344 (cit. on p. 28).

Appendices

A

LiveData catalogue set up

A.1 Setup of LiveData JKAN framework

This section reports the instructions that a LiveData model user (domain expert eventually supported by a technical role) has to follow in order to create a new base LiveData catalogue based on a specific template provided by the LiveData model. The LiveData model user will be referred to in this section as the "catalogue administrator" (or simply "the administrator").

Requirements:

- The administrator has to be a GitHub collaborator of the GitHub organization in charge of managing, and making connections with, the different LiveData nodes created in different geographical and cultural contexts around the world. Such an organization, called DataScientia (DS) [1], maintains, under the same GitHub organization, all the LiveData catalogue's repositories. In other words, the DS GitHub organization is the concrete implementation of the LiveData data distribution network connecting all the different catalogues together. More details about such a network are provided in section 4.2. If the administrator has no GitHub account added as a collaborator of the DS organization, the administrator can make a request for that to the DS technical support (simone.bocca@unitn.it).
- (Required for catalogue local development) Install locally "node js" and "npm" (documentation). You can verify the installation and version of them by running in terminal the following commands:
  - node –version
  - npm –version

Instructions:

[1] DataScientia

1. The administrator has to provide, the DS tech support, with the information for the creation of the new LiveData catalogue repository, the one which will serve the new catalogue. Such information must contain, as a minimal requirement, the name of the repository, which will also be the name of the catalogue webpage that will be accessible online.

2. With the information provided in the step above the DS tech support will create the new repository, for which the administrator will have the rights for any modification she needs to do. The new LiveData catalogue's repository is created starting from the DS catalogue template. The catalogue template UI can be visualized at this link.

3. The administrator has to do some preliminary modifications in the $\_config.yml$ file available in the repository created in the step above. In the $\_config.yml$ file the following fields need to be modified:

   - *title*: the title of the catalogue webpage.
   - *greeting*: the name of the catalogue, that will appear in the catalogue landing page.
   - *description*: the description of the catalogue, that will appear in the catalogue landing page.
   - *baseurl*: the base catalogue URL, that must be composed as follow "/<name-of-the-repository>

A.2 Structure of LiveData JKAN github repository

In this section, we describe the structure of the LiveData template repository that the DS org. Provides to set up a new catalogue. The information provided in this section will help the catalogue administrator during the configuration and customization of the catalogue. Below we describe the most important elements that can be found in the template repository.

- *_data*: this directory contains the yml configuration files used to define the metadata of the resources handled by the catalogue (within the "schemas" directory), as well as the categories that can be used to categorize the resources in the catalogues, and the custom services which can be accessed through the catalogue.

- *_datasets*: this directory stores the markdown file describing (following a precise metadata schema) each dataset uploaded in the catalogue. The markdown files are used by the catalogue to display the metadata for each resource published in the catalogue.

- *_includes*: this directory contains the website pages (HTML files) that are shown in the catalogue web page.
- *_organizations*: similar to the *_datasets* directory, this one contains the markdown files defining the organizations which upload contents to the catalogue. Each organization is described by its own markdown file that is displayed in the catalogue while the name of the organization is clicked into one of the catalogue's web pages (see below for more details about the catalogue's web page navigation).
- *css*: this directory contains the catalogue main.css style file.
- *img*: this directory contains the images used in the catalogue website.
- *scripts*: this directory contains the business logic of the JKAN catalogue. More in detail, in the *src* subdirectory we can find the javascript code divided into different files, containing different specific functions. While in the *dist* subdirectory we can find the bundle.js file which is the file created after building the code in the *src* subdirectory. The bundle.js file is the only file that is actually executed in order to apply the JKAN business logic to the catalogue. This means that all the modification performed in the source code (*src* subdirectory) needs to be reported in the bundle.js file too, this happens automatically when running the build process (see the next section for the instructions about how to make a new build locally).
- *_config.yml* : this is the main *config* file of the JAKN catalogue.
- *datasets.json* : This file defines the structure of each dataset that can be searched within the catalogue. This file is used for the search functionality mostly, this means that, in order to be searchable, the datasets in the catalogue need to respect the structure in this file, which can be updated in order to add new search criteria.
- *index.html* : this is the main HTML file of the catalogue webpage. The visualization of the catalogue website, as well as its navigation, starts from this file.

The remaining files in the repository are not described because they are out of the scope of this document. Nevertheless, you can contact the DS tech support to get more detailed information about the JKAN catalogue repository.

A.3 LiveData catalogue customization

In this section, we report the instructions to be followed in order to customize an already existing catalogue (or even a newly created one). The

idea is to provide the administrator with the possibility to develop locally (in development mode) new features for her catalogue, and then push such modifications into the catalogue's repository in order to be available online (in production). The catalogue web application (website + business logic), as already anticipated, is developed exploiting the JKAN framework that is based on node js, therefore, the administrator needs to be sure to have node and npm properly installed in her local machine (see the Requirements listed above). Then, to do updates and add features to a LiveData catalogue, the administrator should follow the instructions below.

1. Clone the JKAN LiveData catalogue repository created and provided by the DS tech support for your new catalogue (or clone the existing catalogue repository, if that is not a new one).

2. Through the Webpack integrated tool, install the JKAN framework components required to build the catalogue project locally. To this end, open a terminal in the root directory of the LiveData catalogue repository (cloned in the previous step) and execute the following commands:

   - npm install webpack
   - npm run build

   This last command will verify that you are able to build the catalogue source code locally.

3. Install the Jekyll web app server that will execute your catalogue locally.

   (a) Inside the root directory open a terminal and run the following:

   - sudo gem install jekyll bundler

   Some conflicts may appear during the execution of this command. Follow the instructions you will receive from the terminal directly, to solve such conflicts.

   (b) Inside the root directory open a terminal and run the following commands:

   - sudo bundle install

   (c) Then run the server executing, within a terminal, in the root directory the following command:

   - sudo bundle exec jekyll serve

   (d) If everything worked well the Jekyll server is running at (localhost) http://127.0.0.1:4000/jkan/. The localhost URL can change after the port number, where the catalogue name (the same as the repository name) appears.

Once the instructions above have been properly executed, the administrator can execute the following step every time she wants to make modifications locally, and then push such modifications in the relative catalogue repository, to be published online directly.

4. Do the modifications/changes/updates needed in the catalogue source code.
5. Make a new build (thus generating a new bundle.js file that will be executed online) by running the following command, in a terminal, inside the cloned repository root directory:
   - npm run build
6. Run the Jekyll server (see the step 3.c), or close (Ctrl+c in the running terminal) and restart it if it was already running.
7. Open the http://127.0.0.1:4000/jkan/ where the catalogue will be shown with the last updates performed.
8. Verify that the expected behaviour is correct, then (git) commit and push the modification on the catalogue repository by the ordinary git commit-push process.

The DS technical support is available to address any issue related to the above instructions.

A.4 Metadata and search functionality

One of the most important base LiveData catalogue functionalities (if not the most important) is the research of resources published, based on the metadata defined for each dataset. In this section, we describe the instructions to be followed in order to update the metadata structure of the LiveData JKAN catalogue, and how to update the catalogue base search functionality with the objective of enabling the search over a new (custom) metadata schema. To this end, the administrator has to execute locally the instructions below.

1. Add the new metadata by adding new element in the file: $\_data/schemas/default.yml$
2. Update the file datasets.json (see above the definition of this file, in the description of the LiveData repository structure) in order to upgrade the dataset schema used by the search functionality.
3. Update, as follows, the function $\_createSearchFunction(datasets)$ in the file: $scripts/src/components/datasets-list.js$.

- Modify the second line of the function, by adding the name of the new metadata (those to be used in the search functionality) in the $constkeys = ['title','notes','maintainer','tags']$

4. Make a new build of the catalogue source code, in order to produce a new bundle.js file.

5. Push the last build process outcome in the catalogue's repository in order to put the update online.

B
Metadata

This appendix aims at listing the metadata considered by the iTelos methodology to describe the diversity-aware data distributed through the LDE nodes catalogues. The list of metadata reported here below represents the metadata indicated by the iTelos methodology, however, the iTelos users are free to add custom and more context-specific metadata to describe the data they handle. More in detail, the list reported below defines the metadata for three metadata categories, namely People, Project and Datasets.

B.1 People metadata

This section concentrates on the metadata specification for describing different roles covered by people using the iTelos methodology, or involved in a project where the methodology is adopted.

- *ds:member*: The metadata properties assigned for the concept Member are generic in nature and can be used for any of the specific roles which are involved in iTelos projects. The public metadata properties for members are as follows:
  - (a) *ds:comIdentifier*: this attribute encodes the identifier uniquely identifying a person within its LDE.
  - (b) *ds:firstName*: this attribute encodes the first name of the person in a natural language.
  - (c) *ds:lastName*: this attribute encodes the last name of the person in a natural language.
  - (d) *ds:email*: this attribute encodes the email id of the person.
  - (e) *ds:nationality*: this attribute encodes the nationality of the person.
  - (f) *ds:gender*: this attribute encodes the gender of the person.
  - (g) *ds:affiliation*: this attribute encodes the organization to which the person is affiliated in a natural language.

  - (h) *ds:personalWebpage*: this attribute encodes the URL of the personal webpage of the person.

- *ds:prjResearcher*: Includes all members of the iTelos methodology who are recognized as researchers by any university organization. This role inherits all the metadata properties for *ds:member*, plus, the following metadata properties:
  - (a) *ds:researcherProfile*: this attribute encodes the URL of a research profile of the re searcher (e.g., Google Scholar).
  - (b) *ds:areasOfInterest*: this attributes lists the areas of research interest of the researcher.
  - (c) *ds:nameOfPrjsCoordinating*: this attributes encodes the names of the projects which is being coordinated by the researcher.
- *ds:prjParticipant*: Includes all members of the iTelos methodology who are additionally a participant within a particular project. This role inherits all the metadata properties for *ds:member*, plus, the following metadata property:
  - (a) *ds:nameOfPrjsPaticipatingIn*: this attributes encodes the names of the projects in which the participant is contributing to.

B.2 Project metadata

This section concentrates on the metadata specification for describing different projects in which the iTelos methodology has been adopted.

- *ds:prjTitle*: this attribute encodes the name of the project in a natural language as a string.
- *ds:prjURL*: this attribute encodes the dereferenciable URL of the project.
- *ds:prjKeywords*: this attribute encodes the various keywords in a natural language that can be utilized to quickly understand the theme of the project.
- *ds:prjType*: this attribute encodes the type of the project. e.g., Knowledge Resource Generation, Language Resource Generation, KG Generation etc.
- *ds:prjDescription*: this attribute can be used to provide a description of the project in a natural language.
- *ds:prjStartDate*: this attribute encodes the date of the commencement of a project.

- *ds:prjEndDate*: this attribute encodes the date of conclusion of a project. NR
- *ds:prjFundingAgency*: this attribute encodes the name of the agency or institution funding a project.
- *ds:prjInput*: this (repeatable) attribute encodes the various inputs (e.g., datasets, specific cations, etc.) with respect to a project.
- *ds:prjOutput*: this (repeatable) attribute encodes the various outputs (e.g., datasets, domain languages, etc.) with respect to a project.
- *ds:prjCoordinator*: this attribute encodes the name of the research coordinator in charge of a project.
- *ds:prjObservations*: this attribute can be used to record any observations about a project in a natural language.

B.3 Dataset metadata

This section concentrates on the metadata specification for describing the diversity-aware datasets produced by using the iTelos methodology.

- *ds:DatLicense*: this attribute encodes the license of the dataset, e.g., CC-BY-SA-4.0.
- *ds:DatURL*: this attribute encodes the dereferenceable URL of the dataset.
- *ds:DatKeyword*: this attribute encodes the keywords which can quickly convey the topic of the dataset.
- *ds:DatPublisher*: this attribute encodes the publisher of the dataset.
- *ds:DatCreator*: this attribute encodes the creator of the dataset.
- *ds:DatOwner*: this attribute encodes the owner of the dataset.
- *ds:DatLanguage*: this attribute encodes the natural language(s) in which the dataset information is represented.
- *ds:DatLevel*: this attribute encodes the diversity layer of the dataset (i.e., Language, Knoweldge, Data values).
- *ds:DatSize*: this attribute encodes the byte size of the dataset.
- *ds:DatName*: this attribute encodes the name of the dataset in a natural language.

- *ds:DatPublicationTimestamp*: this attribute encodes the timestamp of the publication of the dataset in the respective catalogue.
- *ds:DatDescription*: this attribute encodes the description about the dataset in a natural language.
- *ds:DatVersion*: this attribute encodes the version of the dataset.
- *ds:DatDomain*: this attribute encodes the domain to which the dataset belongs, e.g., society and territory.
- *ds:DatFileFormat*: this attribute encodes the file format of the dataset, e.g., OWL RDF/XML, Excel.