Semantic Visual Query Answering on Heterogeneous Territorial Data

Loris Bozzato, Giulia Bruscagin, Gaetano Calabrese, Luciano Serafini, Cesare Furlanello

Fondazione Bruno Kessler,
Trento, Italy
Email: `<surname>@fbk.eu`

***Abstract*—With the growing availability of administrative data in digital form, one of the current challenges of local municipalities is to manage and integrate information coming from different sources and formats with the aim of recognizing irregularities derived from the combination of such data. In this paper, we present our experience in the context of a local project for the semantic integration of real estate and associated tributes data. The integrated access to these data should support semantic visual analytic, and query formulation and geo-visualization. The data integration is based on an ontology that has been specifically developed to represent the main concepts and relations on this domain. This ontology is populated with the data available in different repositories, such as the cadastre, municipal registry, and household utilities. The ontology has been populated with the available data, which need to be normalized, completed and, in some cases, disambiguated. Finally, we developed a system for the visual formulation and execution of queries over the integrated knowledge base.**

Keywords–Cadastral Information System; Semantic Knowledge Integration; Visual Query Formulation.

I. Introduction

With the growing availability of administrative data in digital form, the local municipalities are nowadays faced with the problem of managing this data coming from different sources and putting it to good use. In particular, one of the advantages of having such data is the possibility to link the different sources and reason on the resulting combined knowledge in order to recognize irregularities and analyze the overall trends of the merged data.

Solving these problems was the aim of the *Geo@Reporter project*, a local project developed in the Trentino region (Italy). The project proposed an integration of real estate and associated tributes data in a common semantic repository, with the support of visual data analytics on cadastrial maps and guided semantic query formulation over the integrated data. The goal of the project was thus to provide municipalities with a comprehensive system for the analysis of the integration of the local cadastral and tax-related information. The choice of defining our system on semantic technologies allowed us to integrate data coming from heterogeneous sources and provide an unified model for the formulation of complex queries. It is interesting to present our experience on this project because, other than the interest in the developed knowledge and software resources, it provides an insight on what are the challenges and possible solutions for the management of geo-indexed semantic data and the development of a visual system for *SPARQL Protocol and Query Language (SPARQL)* query formulation and execution. The contributions of this paper can be summarized as follows:

- We present (in Section III) the structure and development of the Geo@Reporter base ontology. The ontology represents the main objects of the project (cadastral units, subjects and their relations) and their related information (e.g., energy tributes, rents, etc.). The structure of the ontology is derived from the analysis of the different sources to be merged.
- In Section IV, we present the system we developed for the integration and cleaning of the information coming from different sources. One interesting aspect is the management of geographical entities recognition, where an external geographical service is used as reference.
- We describe (in Section V) the system for visual analysis and query for the understanding of the integrated data. The interface for geographical analysis integrates different visual tools for the representation of data over the cadastral maps. The visual query system provides an intuitive way of composing a semantic query that is then executed as a SPARQL query over the semantic knowledge base.

II. Related Work

As we introduced in previous section, in the Geo@Reporter project we aimed at developing a comprehensive system for the analysis of integrated data on cadastral and local tribute data based on semantic technologies. This implies that the system has to provide a combination of different solutions for the semantic representation of (cadastral) data, data import and integration, data analysis and formulation of semantic based queries. As we briefly discuss in the following, to the best or our knowledge, no one of the currently available solutions for these aspects could cover all of the features we needed for the realization of our system.

With respect to the model for representation of cadastral data and related information, different local initiatives in Italy have explored the possibility to use ontologies for its representation, but there is still no agreement on a reference model. For example, in [1] ontology matching is used to map different schemas related to tax and revenues in the Province of Trento in order to integrate new data sources in the local knowledge bases. In [2], semantic technologies and ontologies are used to publish the contents of a territorial information system as Linked Open Data. However, these works do not aim at using their semantic representations to provide a general system for the management of cadastral and tributes data. Moreover, we highlight that, in the case of the Trentino province, the cadastre records have a slightly different structure from the Italian records, historically inherited from the Austrian cadastre. We note that a current effort to standardize ontologies and vocabularies for the Italian public administration is ongoing under the *OntoPiA* initiative [3].

In order to populate ontology based knowledge bases from different data sources, there are currently solutions to convert (mostly tabular) data in *Resource Description Framework (RDF)* format. A relevant example is *OpenRefine* [4], a Java-based tool to define and apply transformations on data files which is commonly used to publish (tabular) data in RDF format. Related approaches include mapping languages like *RDB to RDF Mapping Language (R2RML)* [5], which allow for the definition of transformations of data stored in a relational database for their exposition as RDF data. In our case, however, these approaches are not enough: first of all, the format of some data sources (specially in the case of cadastral data) is not strictly in a tabular format (i.e., records are positional and have multiple record formats depending on some of the fields) and thus the manipulation of such sources needs more freedom. The type of data formats we need to import is also different, thus our need is to provide a single (but extensible) tool to integrate all of the sources.

Different solutions for implementing web based dashboards for data analysis exist. Among the currently available open source solutions, two major proposals are *Superset* [6] and *Kibana* [7]. The objective of these environments is to facilitate the realization of web interfaces with views and graphical indicators over a given database, in order to provide users with insights on their datasets of interest. With respect to query composition, there exist libraries that can be promptly included in web interfaces, as for example *jQuery Builder* [8]. An advantage of such solutions is that they facilitate the definition of complex queries to non expert users. On the other hand, in our project we need to adapt and combine the current available tools to the specific case of representation of cadastral maps and building structures. Also, the current web libraries for query composition are not intended to work with SPARQL queries, so further work is needed to compose such queries, interact with the RDF knowledge base and represent the returned results.

III. Ontology Structure and Development

In this section, we detail the modelling activity and resulting structure of the *Geo@Reporter ontology*, which defines the base schema for the representation of the semantic information of the project. The ontology has been developed following the common guidelines and methods for ontology engineering (see, e.g., [9]). In particular, given the goal of integrating specific knowledge sources, the modelling has been guided by the form and contents of the initial data.

A. Ontology specification

The goal of the realization of the ontology is the formalization and schematization of the aggregated data in the Geo@Reporter Knowledge Base (KB). In particular, the ontology aim is to represent the central objects of the project knowledge base, i.e., buildings, their cadastral subdivisions together with their owners and their associated features. The ontology will be formalized and implemented in *Web Ontology Language (OWL 2)* [10], which allows us to take advantage of the available tools for modelling and reasoning in the standard ontology language. The formal definition of the ontological model is thus oriented towards the definition of the relations across the imported data sources, so to permit the successive population of the knowledge base, and the primitives for access to the main objects of the project, so to facilitate the formulation of relevant semantic queries on the integrated data.

B. Contents

The specification of the contents of the ontology, that is the extraction of the domain objects to be included in the ontology representation, has been driven by the need for the integration of the available data sources. The identification of the domain objects to be represented began with the analysis of the initial data sources and their structure. The interpretation of this data has been supported by meetings with domain experts (developers of cadastral related systems, knowledgeable about the details of data sources and interests of municipalities), who guided the modelling decisions in this developement phase by recognizing the most important objects and properties. The modelling of the ontology was driven by the initial data sources motivated by the heterogeneous nature of the sources and the need to recognize connections across the integrated information related to the represented objects. This analysis is also functional to the identification of the format of the different information pieces that need to be integrated in the final KB, thus it is preliminary to the design of the KB integration system.

The initial data over which we based our modelling consisted of an extraction relative to one year of two specific municipalities in the Trentino region. The following sources of data were identified:

Cadastral data: the source includes land and real estate cadastral information; the geographic information about boundaries and shape of the parcels are stored separately in the visualization interface system. This data has been provided in a proprietary format in form of pipe-separated text files (together with documentation for interpretation of fields). Cadastral data contains the principal objects of the discourse, thus will represent the central objects of the model. Principal objects from this data source are land and real estate units, owner subjects and their onwership relation with cadastral units.

Civil registry data: this source provides complete information about family units from the municipalities' registry data. This data is provided as *comma-separated values (CSV)* format files for individuals and families units. Main objects of this sources are resident individuals and family units.

Real estate tributes: this data includes all information from the municipality about real estate tributes for the specific year. The data is provided as a spreadsheet export of the administrative software of the municipality. The central content of this source is the information about municipalities taxes to be linked to local real estate units.

Utility and rent contracts: similarly, data about rents and utility contracts for waste collection, gas, electricity and water were provided by the municipality. These data is extracted as spreadsheets from the *Sistema Interscambio Anagrafe Tributarie Enti Locali (SIATEL)* system for exchange of tributary data in the Italian public administration. The main objects of interest are contracts (and their related information) for the different utilities linked to specific real estate units.

As it can be noted, the sources are heterogeneous for format and structure of the data, while also overlapping on some of the represented objects.

C. Application

Parallel to this activity of knowledge sources analysis, we identified the intended users and uses of the model. The

 ISBN: 978-1-61208-765-8

ontology will be used mostly by two "users" (i.e., components of the system):

(1). Data analysis interface: the visualization interface has to allow the user to visualize data in the KB and permit different kinds of aggregations and query over the integrated data, with the goal of analyzing the connections and possible anomalies across the different sources. The interface will not modify the KB, but allow both simple retrieval of objects in the KB (e.g., for their visualization on a map) and complex queries that require interaction of objects from different data sources.

(2). Population and integration system: the system has the objective of inserting (or updating) in the KB the objects derived form the different data sources, with the creation (or update) of new instance data in the form specified by the ontology. The system is typically to be used at the "setup" of a new instance of the system with the import of data relative to a specific municipality to be analyzed or, in a later time, with the addition of new updated data. The integration system has the goal of uniforming the different sources of data to the RDF format used in the KB; during insertions, the system can use query services exposed by the KB to retrieve references to existing objects that need to be linked to the new objects.

D. Conceptual modelling

From the analysis of the presented data sources and use cases, we then proceeded to the conceptual modelling of the ontology schema. In Figure 1, we show an overview of the resulting conceptual schema. The conceptual modelling started from the analysis of the structure of cadastral data, given that it contains the central objects to be represented, and then completed by developing the sub-models for each of the related data sources. Figure 1 displays the main classes of the model (where the larger labels are the main objects of the model) and the main object properties across them. Gray boxes mark the classes of the sub-schemas for *cadastral*, *contracts*, *tributes and utilities* and *civil registry* information. Note that, by the integration of such schemas, many of the shared concepts like e.g., *Indirizzo* (address) or *IdentificativoCatastale* (cadastral identifier) become hubs for the access to information extracted from different sources.

E. Implementation

The resulting schema has been realized as an OWL ontology implementing the conceptual model. This schema will constitute the schema at the base of the project KB for the semantic representation of the integrated data. Intuitively, the physical modelling of the conceptual schemas is realized by translating to OWL class inclusion axioms the hierarchy of classes described in the conceptual model; similarly relations across classes are modelled as object properties while data fields associated to each object are modelled as a distinct data property. In the implementation of the ontology further features of the properties (e.g., functionality, transitivity) can be specified and hierarchies across properties have been defined. We report in Table I some metrics about the final version of the Geo@Reporter ontology. The rather small number of classes is due to the limited number of object types to be represented, while the larger number of datatype properties is related to the quite extended set of (mostly numerical and textual) information associated to such objects. Property axioms are mainly composed of domain and range assertions on object and data properties.

TABLE I. GEO@REPORTER ONTOLOGY METRICS

DL expressivity	$\mathcal{ALUHF}(D)$
Classes	29
Object / datatype properties	28 / 230
Annotations	284
SubClass axioms	12
Object / datatype p. axioms	63 / 462

The current version of the ontology is available for download in [11].

IV. KNOWLEDGE BASE INTEGRATION SYSTEM

After the realization of the ontological schema, we developed a system for importing the data from the different sources and integrating this information as semantic objects of the KB. Figure 2 shows the architecture of the integration system. The architecture has been designed with the goal of providing independence between the management of the format of input data and their information content to be integrated in the KB.

Intuitively, the import process is divided in two steps. In the first, the *input wrapper* component works on the format of the input files to apply data cleaning operations and to recognize relations across entities. In the second step, different *update services*, implemented as *Representational State Transfer (REST)* interfaces, allow to interact with the KB in order to add to the KB (or update) the information derived from input files in terms of the structure of the ontology. In the first step, the data cleaning and entity definition operation are kept separated in the input wrapper implementation so to facilitate their maintainability in case of the integration of new data formats.

In the first phase of the import, every input wrapper (designed for a specific data source) takes as input the data files containing the objects to be integrated in the KB together with a *mapping file* and *headers file* if required by the input data format. The mapping files, implemented as *JavaScript Object Notation (JSON)* textual files, define the correspondence between the fields in the input records and the ontology properties of the object to be created in the KB. By representing this information as a file external to the implementation, the mapping files ensure a better adaptability of the import procedure to the changes in the format of input files. For some of the input formats (in particular, for the cadastral data), the fields' headers are not specified in the input files but are documented externally. For such data formats, the input wrapper also needs a further support headers file containing the fields' interpretation.

During the data import, the system implements some base operations for cleaning and uniforming the data (e.g., by uniforming the format of dates, numbers and person names). These data refinement procedures have been developed during the analysis of the data sources, contextually to the development of the wrappers, with the aim of uniforming the data contents of the KB and thus facilitate the query operations on the integrated data. In particular, in order to facilitate the identification of elements on the map visualization of data, the data related to objects of type *Indirizzo* (address) has been normalized and completed with an external service. On the base of the data available on an address, the external service

 ISBN: 978-1-61208-765-8

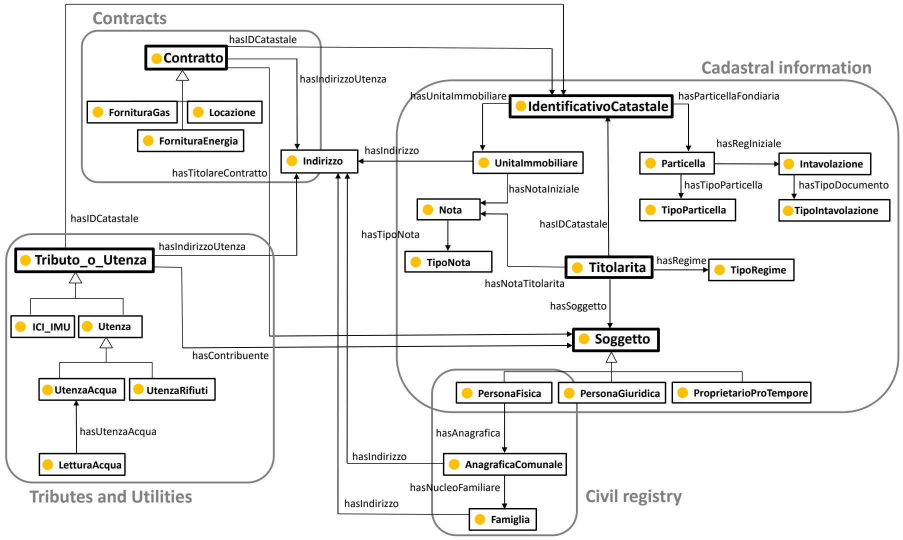

Figure 1. General schema of the Geo@Reporter ontology.

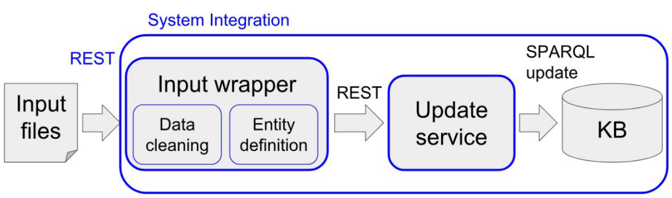

Figure 2. Integration system architecture.

Google GeoCode [12] (based on GoogleMaps API) is used to retrieve a normalized address string and its geographic coordinates. This new data is added (in post processing) to the address objects in the KB by distinct data properties. Similarly, to facilitate the recognition of links across elements of the KB, a post processing procedure has been developed that recognizes the possible relation across utility contracts and their associated cadastral units. The (one-to-many) association is derived by computing the proximity between the address specified for a contract and the addresses of cadastral identifiers.

Once the objects to be added to the KB have been defined by the first step, the actual creation of the objects in the KB is executed in the second step of the architecture (*Update service*). This is realized as calls to REST operations (for each kind of object) providing the interface for SPARQL based insertion queries on the RDF store implementing the knowledge base (in our case, *Eclipse RDF4J* [13]). In this phase, REST interfaces implementing the query services are used to recognize and retrieve entities that need to be linked to the new objects. Actually, both the *update services* and *query services* on the KB are implemented as REST interfaces. They define the access interface to the KB by allowing to abstract from the details of the (SPARQL based) interaction on the RDF store.

The integration system implementation has been completed with respect to the initial data sources, by realizing the necessary wrappers and mapping files for the import and cleaning of the initial data files. To simplify the import process, a simple Web interface has been realized to facilitate the access to the import services.

To verify and validate the work on the integration system and the ontology implementation (and for assessing the visualization components), the system has been tested on the initial data from the Trentino municipalities. An integrated KB for each of the two municipalities has been successfully obtained.

V. VISUAL QUERY ANSWERING SYSTEM FOR DATA ANALYSIS

The main purpose of the research activity of this project is to provide a Web-based instrument that can make affordable the analysis of heterogeneous data through one visual interface. In public administrations, there are usually various pieces of software dedicated to managing different data sources. The analysis between these datasets is difficult and problematic and is vulnerable to ambiguities, inaccuracies, and inconsistencies. In order to solve this problem, we developed an innovative visual system that can guide the user to explore and combine different data sources with flexibility, consistency, and intuitivity. We developed three main visualizations tools for Geo@Reporter:

- *Smart Query Builder*: a visual-based system for complex semantic query building on the KB;
- *Logic Tree*: a visual representation of the query results, highlighting the existing relationship between the different ontology elements;
- *Hybrid 2D/3D Geographic Information System (GIS)*: a multi-level visualization that combines 2D and 3D environments to describe and inspect geo-referenced entities from a high level to detailed features.

These tools are developed so that they can dynamically display the ontology model and, in case of changes to entities or tables, they can automatically adapt with no additional code modification.

A. Smart Query Builder

Public Administration identifies discrepancies in tax reports or cadastral documents through the comparison of various data sources. This activity is often very demanding in terms of time and resources, since most of the time these analyses are done manually. The chance to overlook some details or to commit some mistakes is very high. Furthermore, the dataset is often incomplete, and the user cannot easily find a way to correlate between two entities.

For this reason, we decided to create a graphic instrument that can guide the user easily and with high flexibility in the search process. We developed *Smart Query Builder* (SQB), a graphical interface that allows the user to easily compose complex queries and display the aggregated results in a table with optional geo-referenced data represented on the cadastral map. This system is highly interactive, with user-friendly user experience and allows to: (i). retrieve all the information from the ontology in a series of easy steps; (ii). query the ontology from different starting points, depending on the objective of the analysis. Smart Query Builder is a *Decision Support System (DSS)*: the tool does not identify problems or inconsistencies, but it is the user, after retrieving the results from the ontology, that can easily judge the investigated situation and decide an action strategy. The graphic design and the logical functionality of Smart Query Builder resemble the standard of *Visual Programming Languages (VPL)* where the user creates a query manipulating the elements graphically rather than specifying it with a procedural language (like, e.g., SPARQL). Therefore, queries and ontology are accessible even by less expert users, with no specific knowledge in programming.

Our Smart Query Builder is built as a customization of the Open Source library *jQuery Builder* [8] and it is developed in JavaScript. jQuery Builder provides a visual interface in order to build complex interrogations and it was the starting point for our research and development activities for this part of the project. The most important features we added were: (i). interoperability with SPARQL (not originally included within the specification list); (ii). query consistency controls (AND/OR/NOT functions are permitted only between entities with effective relations). We developed a specific workflow to enable the following steps:

1) Dynamic and flexible query-building process: the user chooses the first entity, and the SQB progressively suggests only the entities directly or inversely correlated to the previously chosen entity.
2) The set of rules is translated to JSON format and sent to the knowledge base;
3) A dedicated service of the KB translates the JSON dictionary into a SPARQL command, that is sent to the RDF store;
4) The extraction from the KB is returned to the front-end and results are displayed in tables and on the map.

Figure 3 shows the graphical interface developed for Smart Query Builder. After choosing the first entity (i.e., *Contratto Locazione*, rent contract), the list of connected entities is available in a drop-down menu. The user can choose mathematical and logical operators in a dedicated drop-down menu and insert free text in the empty box to complete the query with custom parameters. As long as the Smart Query Builder finds relations with other entities, the user can keep adding new boxes, creating a more complex query. The results can be consulted in tables and on a specific layer on the cadastral map. A donut pie chart is placed on the centroids of the corresponding cadastral parcel on the map, showing the number of real estates present on the parcel, highlighting the ones resulting from the query (Figure 4).

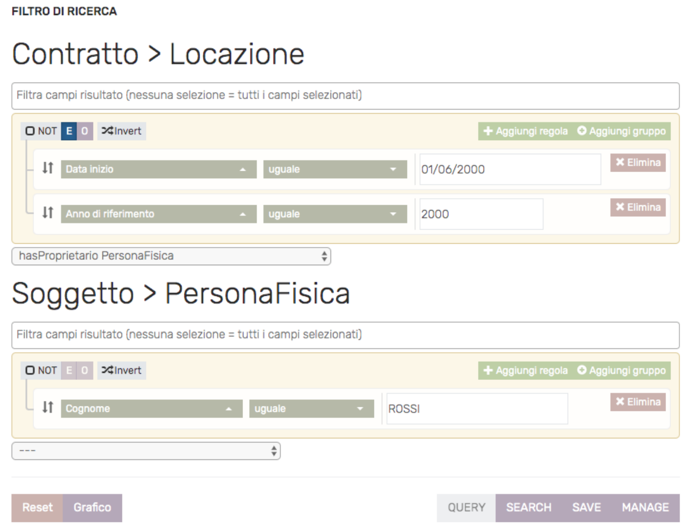

Figure 3. Example of use of Smart Query Builder.

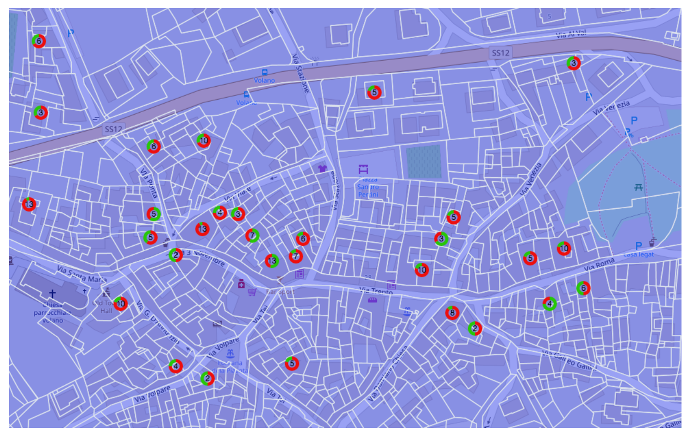

Figure 4. Example of graphical and geo-referenced representation of the results of the query (Map data © OpenStreetMap contributors, CC BY-SA).

B. *Logic Tree*

In order to highlight the association between cadastral parcel, real estate identifier (*Unità Immobiliare*) and personal/services/tax-related data in the ontology, we defined a graphical logic structure to represent the relations between the various entities. For this purpose, we choose a tree representation that we called Logic Tree. This structure can visually represent the connection between cadastral parcel and the various properties of the parcel itself, with a strong hierarchical mark. The structure of the Logic Tree intuitively represents the existing connection between the cadastral elements and the entities referred to them (i.e., tax-related data, rent contracts, services, etc.). The selection of a subset of cadastral parcels with Smart Query Builder triggers the automatic generation of the Logic Tree where the user can interact on its leaves, querying the entities related to the parcel under investigation. The data are organized in the following levels (Figure 5) of the tree:

- *Level 0:* cadastral parcel;
- *Level 1:* real estate units;
- *Level 2:* personal and tributes data, services and contracts.

The graphical structure of Logic Tree is particularly helpful

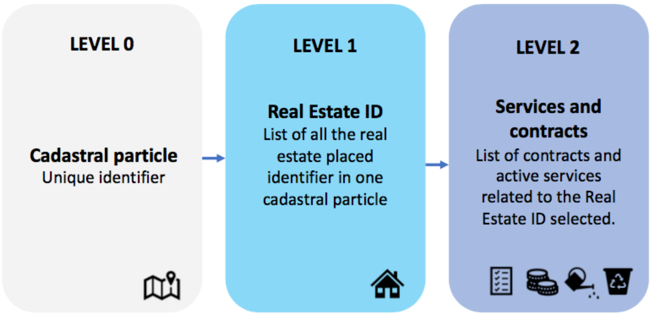

Figure 5. Logic scheme of Logic Tree.

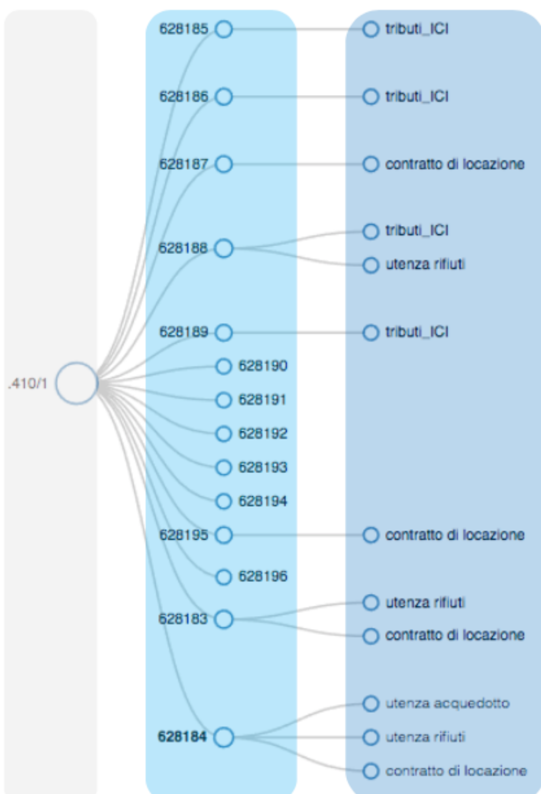

Figure 6. Example of Logic Tree: colors reflect the levels of the structure.

to understand the dependencies between the entities of these kinds of datasets (Figure 6). In particular, Logic Tree highlights the central position of the cadastral data and its structure reflects the ontology structure thanks to the visual aggregation of different data sources through simple graphical connections.

C. Hybrid 2D/3D GIS System

One of the most important data sources of the Geo@Reporter Project is the geo-referenced and geometric descriptions of the real estate properties, that are collected from two different sources: the cadastral data and the documentation inventory report, called *Documenti Catasto Fabbricati (DOCFA)*, officially registered to the *Ufficio del Catasto* (Municipal Building Department). A dedicated GIS module equipped with specific functions was developed for Geo@Reporter software. The GIS module, allows the user to geo-reference and navigate the consulted data in a spatial context.

To empower the user with the best possible data exploitation, we explored and tested different visualization solutions. We took into consideration different visualizations systems ranging from 2D to 3D environments. Considering the type of data involved, we were strongly interested in the use of the 3D representation, in particular, to model real estate data available in 3D for the single cadastral parcel. There are various approaches that allow three-dimensional modeling and that can be more or less suitable depending on the available data. We focused our research activity on two main components of these systems: (i). Modeling and exchanging formats of semantic enriched 3D data; (ii). Web libraries for Dynamic 3D visualization. We evaluated and tested all the main functionalities of several software and formats, considering the benefits and disadvantages, given the requirements of Geo@Reporter. The most suitable stack of software selected for the project was:

- *Leaflet* [14]: an open-source JavaScript library for 2D interactive maps;
- *Three.js* [15]: an open-source JavaScript library used to create and display 3D models in a Web browser;
- *Cesium* [16]: an open-source JavaScript library for world-class 3D globes and maps.

The DOCFA documentation provides most of the information required to reconstruct the three-dimensional representation of the building, but it does not specify the geographical coordinates. Thus, it is possible to generate a 3D reconstruction of the entire building or even just a part of it, but the 3D model cannot always be exactly placed in a geo-referenced map. We conducted several usability tests with Cesium, to check the performances of the library and the representation quality. While very immersive, the overall experience in using Cesium showed several usability concerns. The navigation and the consultation of 3D objects can be confusing and difficult due to the free camera and location movements that, in some cases, the user can find difficult to control. Furthermore, Cesium hides the representation of objects that are located under the road surface (i.e., cellars, basements, parking lots, etc.). For these reasons, we decided to place side by side the bidimensional cadastral map and the three-dimensional building in two separate modules. The cadastral map can be navigated with Leaflet and a connected module is dedicated to the 3D reconstruction of a selected building with Three.js. In this way, the user can navigate the bidimensional map (published with Leaflet) and, when interested in consulting the features of a particular cadastral particle, the 3D model (rendered with Three.js) shows the reconstruction of the whole building placed on the selected particle (Figure 7).

The usability test (performed with the domain experts) showed that this hybrid representation is easier to explore rather than a full 3D environment. In Three.js, the point of view of the visualization is always fixed towards the center of the building and the library allows the reconstruction of those parts that are under the road surface, allowing a complete exploration of the structure.

VI. Conclusion and Future Work

In this paper, we presented our experiences in developing a semantic-based platform for the Geo@Reporter project for

 ISBN: 978-1-61208-765-8

Figure 7. Example of Hybrid 2D/3D GIS Representation (Map data © OpenStreetMap contributors, CC BY-SA).

the representation and analysis of local administrative data. We first detailed the structure and modelling of the project ontology, by showing how the different data sources guided the schema definition. Then, we detailed the integration system of the platform, which allows to clean and map different data formats to the semantic based representation of the project KB. Finally, we presented our solutions for data visualization, providing different tools for data analysis and 2D/3D map navigation.

The project has been currently tested and applied to data for municipalities in the Trentino region. As a future direction, it can be interesting to extend the knowledge model and integration system to allow for additional information and data formats. In particular, a KB currently represents the data relative to a municipality for a specific period of time. It would be interesting to add a structured concept of *context* to sets of such data (see, e.g., [17][18]) to represent "snapshots" of such information and their relations. Similarly, the system can be extended to be applied to other Italian municipalities (but also to local administrations outside of Italy) by generalizing the ontological model and integration system to new sources of data. This can be useful to perform usability tests on the platform and further extend the system data-analysis capabilities.

ACKNOWLEDGMENT

This work has been supported by the *Geo@Reporter* project (*Progetto Di Ricerca Applicata*), funded by the Province of Trento, Italy under L.P. 13.12.1999, n. 6, art. 5.

REFERENCES

[1] S. Brida, M. Combetto, S. Frasson, and P. Giorgini, "Tax and revenue service scenario for ontology matching," in OM, ser. CEUR Workshop Proceedings, vol. 551. CEUR-WS.org, 2009, pp. 242–243.

[2] S. Consoli et al., "D4.1.2. Linked Open Data Arricchimento e pubblicazione dei dati," CNR, PRISMA Project Deliverable D4.1.2, Jan. 2013.

[3] "OntoPiA - la rete di ontologie e vocabolari controllati della Pubblica Amministrazione," 2019, URL: https://github.com/italia/daf-ontologie-vocabolari-controllati [accessed: 2020-03-19].

[4] "OpenRefine," 2019, URL: https://github.com/OpenRefine/OpenRefine [accessed: 2020-03-19].

[5] S. Das, S. Sundara, and R. Cyganiak, "R2RML: RDB to RDF Mapping Language," W3C, W3C Recommendation, Sep. 2012.

[6] "Apache Superset," 2019, URL: https://github.com/apache/incubator-superset [accessed: 2020-03-19].

[7] "Kibana," 2019, URL: https://www.elastic.co/kibana [accessed: 2020-03-19].

[8] "jQuery Builder," 2019, URL: https://querybuilder.js.org/ [accessed: 2020-01-28].

[9] Y. Sure, S. Staab, and R. Studer, "Ontology engineering methodology," in Handbook on Ontologies, ser. International Handbooks on Information Systems. Springer, 2009, pp. 135–152.

[10] W3C, OWL 2 Web Ontology Language Document Overview. W3C Recommendation, 2009.

[11] "Geo@Reporter ontology," 2019, URL: https://github.com/dkmfbk/GeoreporterOntology [accessed: 2020-01-28].

[12] "Google GeoCode," 2019, URL: https://developers.google.com/maps/documentation/geocoding/intro [accessed: 2020-01-28].

[13] "Eclipse RDF4J," 2019, URL: https://rdf4j.org/ [accessed: 2020-01-28].

[14] "Leaflet," 2019, URL: https://leafletjs.com/ [accessed: 2020-01-28].

[15] "Three.js," 2019, URL: https://threejs.org/ [accessed: 2020-01-28].

[16] "Cesium," 2019, URL: https://cesium.com/cesiumjs/ [accessed: 2020-01-28].

[17] L. Bozzato, C. Ghidini, and L. Serafini, "Comparing contextual and flat representations of knowledge: a concrete case about football data," in K-CAP 2013. ACM, 2013, pp. 9–16.

[18] L. Bozzato, L. Serafini, and T. Eiter, "Reasoning with justifiable exceptions in contextual hierarchies," in KR 2018. AAAI Press, 2018, pp. 329–338.