

Received September 26, 2019, accepted October 14, 2019, date of publication October 17, 2019, date of current version November 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2948115

Intuitive Ontology-Based SPARQL Queries for RDF Data Exploration

ALEJANDRO RODRÍGUEZ DÍAZ¹, ALEJANDRO BENITO-SANTOS¹, AMELIE DORN²,
YALEMISEW ABGAZ³, EVELINE WANDL-VOGT², AND ROBERTO THERÓN¹

¹Visual Analytics and Information Visualization Group, Department of Computer Science and Automation, University of Salamanca, 37002 Salamanca, Spain

²Austrian Centre for Digital Humanities, Austrian Academy of Sciences, 1010 Vienna, Austria

³Adapt Centre, Trinity College Dublin, O'Reilly Institute, Dublin 2, Ireland

Corresponding author: Alejandro Rodríguez Díaz (jancho@usal.es)

This work was supported in part by the Nationalstiftung of the Austrian Academy of Sciences through the Funding Scheme Digitales kulturelles Erbe under Grant DH2014/22, in part by the ExploreAT! Project, carried out in collaboration with the VisUSAL Group, Universidad de Salamanca, Spain, and the ADAPT “Centre for Digital Content Technology”, Dublin City University, Ireland, through the Science Foundation Ireland Research Centres Programme under Grant 13/RC/2106, in part by the European Regional Development Fund, and in part by the CHIST-ERA Programme through the National Ministry of Economy (MINECO), Spain, under Grant PCIN-2017-064.

ABSTRACT This paper introduces a strategy for both the retrieval and analysis of linked open data (LOD) based on the use of visual tools. Retrieving and understanding data from triplestores (such as SPARQL) requires technical knowledge and proves to be challenging with large datasets, which result in an increase in the mental overload when unknown ontologies are involved in the creation of complex queries. These two problems benefit greatly from visual techniques that allow for executing them in an easier and more intuitive manner. These techniques have already been applied to each problem separately; however, we propose combining them to lower the complexity of triple-store data retrieval and empower its exploration and analysis through specific data visualizations. To demonstrate the suitability of this strategy, a web-based tool was implemented. It allows for the creation of interactive queries using node-link ontology representations, and a subsequent filtering and analysis through a configurable dashboard with different data visualizations. This paper also presents a user-centred design process and evaluation of the proposed tool.

INDEX TERMS Semantic Web, visualization, software tools, linked open data.

I. INTRODUCTION

Visualization is a key human capability for understanding information [1]. Visual representations are aimed at leveraging the user's understanding of datasets. The rapid understanding of online databases is a well-known problem in the visualization of Semantic Web resources. While the majority of approaches have focussed on either the visualization of ontologies or the underlying linked open data (LOD), methods offering a holistic view of semantically-enabled datasets by combining these two aspects are scarce. In particular, combining of several views that offer different perspectives on the same dataset is supported by the use of linked views and dashboards [2], [3]. Previous work has differentiated two different tasks in the exploration of data using the Resource Description Framework¹ (RDF) standard—the

retrieval and exploration of data.. These are two separate tasks, requiring the use of different types of data; thus, they are performed with different tools.

In our study, we focus on the exploration of data exposed through SPARQL² endpoints and the asserted ontology where the relationships are defined. Other tools use ontologies to reason about the information stored, applying semantics to the retrieved data and allowing for a more structured and less technologically coupled manner of accessing such data. Often, node-link graphs are used to represent the ontologies. This paper introduces a workflow for retrieving and exploring RDF data, where a task-specific view is designed to accommodate the user's needs. By performing the tasks in different views, it is possible to take into account task-specific needs and provide visual tools that best suit the mental models and techniques associated with the task. We implemented a web application to demonstrate this approach. This tool

The associate editor coordinating the review of this manuscript and approving it for publication was Feng Xia¹.

¹<https://www.w3.org/RDF/>

²<https://www.w3.org/2001/sw/wiki/SPARQL>

provides a three-step workflow where the sources are selected, the data is retrieved by means of visual queries, and a configurable dashboard exploring the retrieved data is provided. Dashboards can offer an extra degree of flexibility by allowing the user to select and combine the visualization that he/she considers most relevant to answer the questions at hand. In our case, the SPARQL query language might not be readily accessible for casual and/or inexperienced users due to its steep learning curve, a fact that calls for the creation of novel tools that circumvent this problem. More concretely, there are examples in the literature, such as the work done in [2], [4]–[7], which shows how this can be tackled by means of well-established visualization techniques. Thus, one main contribution of the work presented in this paper is the automatic generation of SPARQL queries employing visual tools that are accessible to a broad profile of end users.

Additionally, we conceive the filtering of data as an interactive task done once the user has retrieved it and is able to explore it, rather than a previous step done in the database query. This decision leads to a shorter and more intuitive initial phase where queries only require an entity and attribute selection.

The present article covers both the workflow and its implementation as an interactive tool in the following order: Section II discusses previous work in ontology visualization, RDF data visual exploration and dashboard usage in the field. This is followed by a presentation of the proposed workflow in Section III. To demonstrate this workflow an exploratory tool was implemented; thus, in Section IV, the iterative usability testing design process is explained, followed by an in-depth explanation of the tool implementation in Section V, and exemplified with two use cases in Section VI. In Section VII, we discuss the results of the final tool evaluation and how this tool compares with previous work on the subject feature wise. Finally, we present the conclusions in Section VIII.

II. PREVIOUS WORK

The use of visualization tools for Web Ontology Language³ (OWL) ontologies has been thoroughly addressed before [8]–[11]. One of the major contributions of our paper relates to bringing together three well-separated tasks of semantic visualization that can be found in the literature. These are: 1. ontology visualization, 2. visual query building and 3. interactive exploration of RDF data. Our initial motivation was to create a platform on which the user is able to create on-demand, configurable dashboards that allow the RDF data to be explored with a minimal informational trade-off.

The two main sources of information used by the platform are OWL-defined ontologies and RDF data. We have mentioned before that the ontologies defined in OWL represent interrelationships of entities within a specific knowledge domain. This knowledge is modelled by means of a) classes, b) sub-classes and c) individuals, along with their respective

specific properties. RDF is a standard modeling language that is based on a definition of resources given by subject - predicate - object triples that allow for data interchange.

We make use of the ontology to access the RDF database in a visual and user-friendly interface that does not require prior programming experience or knowledge of the underlying query language (SPARQL). In the data exploration process, however, both the task and the tools are different, and for that reason, we present a different interface, which makes use of proven techniques that assist with the exploration task of the data from different points of view. Both of the aforementioned tasks require the user to understand the different information and tasks involved in order to create a mental model based on the understanding and interpretation of it; a mental model will be then used to reason about what is presented and what can be done with the available information. This process of creating the mental model is called visualization and constitutes the most important step for successfully performing a task.

Usually by means of the visual channel, this problem has been present in the field of the Semantic Web for different tasks: providing visual representations and tools that help the user to understand the underlying semantics (usually provided in the form of an ontology) [12], [13], navigating through different datasets available in LOD repositories [14], or performing a more in-depth exploration and analysis of the data [2], [4]–[6], [15].

A. ONTOLOGY VISUALIZATION

The visual representation and manipulation of ontologies is solved commonly by means of node-link diagrams. This technique has proven successful in the correct representation of the hierarchical structure of an ontology over other potential alternatives. For example, treemaps [16] or circle packing [17] are other popular hierarchy visualizations on which nodes are given more relative importance and, thus, are more applicable to tasks such as finding files in a file system [18] or comparing hierarchical levels. Sunburst visualizations [19] are more suited to showing the different depth levels in a hierarchy. If the aim is to highlight the relationships between the parts, other visualizations such as chord diagrams, Sankey diagrams [20], and arc diagrams [21] have been documented to be more appropriate.

Different node-link diagrams are used depending on the user's needs: Suggested change: Unified Modelling Language (UML) representation for a better understanding of class hierarchies and restrictions [22], spring layouts with different links for representing each type of relationship [4], or combined multiple views, showing the class nodes with the attributes inside of them to easily understand the class structure along with the hierarchical trees and allow quick access to classes, properties, and scope. Another example of the use of multiple simultaneous views for on ontology visualization is the work introduced in [23], which shows different hierarchy-based visualizations (hyperbolic tree,

³<https://www.w3.org/TR/owl-ref/>

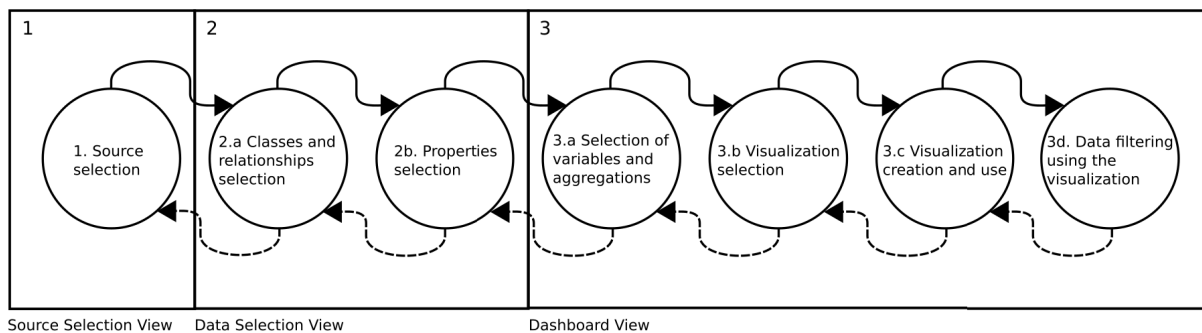


FIGURE 1. Proposed workflow. Diagram showing the user's natural workflow and the separation of concerns through the use of different views.

2.5D node-link graph, icicle tree) to provide both a global and a detailed view for specific classes.

B. INTERACTIVE VISUAL EXPLORATION OF RDF DATA

Although the literature is rich with examples of ontology visualizations, especially for ontology editors and navigators, less work has been done in visual navigation of RDF sources using these ontology visualizations.

Three main approaches can be distinguished, whereby the RDF data is navigated, explored, and analysed. A clear example of the first approach are lattices (proposed in [5]), where the RDF data is used to provide navigation space to reason about the relationships and properties among the entities. Other tools such as [4], [14] integrate RDF in the ontology visualization, providing an exploration-oriented interface. Examples of the third approach, which provides data access and analysis together in the same application, can be found in [15], where a node-link diagram with the structure of the retrieved data is shown after the SPARQL queries are performed, and a second view is used to select the visualization which best summarizes and provides insight.

III. WORKFLOW

We propose a workflow for retrieving and analysing LOD that focusses on aiding new users to perform the tasks and lower their cognitive load. One key aspect of the workflow is the separation of concerns among views in order to diminish its complexity, as opposed to other approaches that start with complex queries requiring:

- Previous in-depth understanding of the data;
- Knowledge of query languages such as SPARQL; and
- A well-defined need and question to be answered.

Similar to the strategy for separating task concerns to the one proposed in [2], where three main tasks were identified and a different view was provided for each of them, we present a three-step workflow in Figure 1, as detailed below. Each step requires different actions and are expected to provide different results; thus, using a dedicated view allows for making the best use of the visualizations for each task. This approach is more difficult with other single view applications. Thus, our workflow consists of:

- 1) Selecting the sources.
- 2) Selecting and retrieving the data:
 - a) Selecting classes and relationships; and
 - b) Selecting properties.
- 3) Visually exploring the data, which comprises:
 - a) Selecting a set of variables and aggregations;
 - b) Selecting a visualization;
 - c) Creating and using the visualization; and
 - d) Filtering data through interactive controls.

As illustrated in Figure 1, the workflow focusses on making data visualization, exploration, and filtering soon by reducing the steps in the first and second tasks. This reduces the overload of data retrieval and promotes spending more time in the later exploration and analysis.

This is often not encouraged by traditional approaches, where the entry point is usually a complete query that requires extensive knowledge of the stored data and the ontologies used. This approach is error-prone, challenging the researcher to have a clear question and provide a sophisticated query, including all necessary entities, properties, and filters.

Interactive approaches, such as the flowcharts used in [2], aid in the construction of a query. However, the user still needs to know what specific attributes and filters can be applied. We believe the retrieval phase should be simple enough to be used iteratively, abstracting the selection process to better show how the selection relates to the underlying ontology structure. This can help to reduce the number of decisions and input needed from the user. Additionally, the design and implementation for this interface should favour the three basic exploration tasks described in [24]:

- **Finding Gestalt:** Through projections on the variables, rearranging elements, and zooming (among others) to focus on a view and identify Gestalt features, such as outliers, linearities, groups, and discontinuities.
- **Posing queries:** Being able to answer visual originated (from the previous task) questions through visual means; such as highlighting, colouring or brushing information related to a subset of the data that is being queried.
- **Making comparisons:** Being able to create multiple views and arrange them in such a way that the layout is

TABLE 1. Correspondence between actions in the workflow and tasks described in the task by data type taxonomy.

Task	Description
Classes and relationship selection	
Overview	Providing an overview of the collection implies allowing to get a sense of how much information there is for a certain class at a glance.
Zoom	The functionality that allows focusing on a specific item of the ontology.
Filter	A task which allows the user to avoid getting distracted with classes, relationships, and properties that are not accessible or relevant to his/her question.
Details-on-demand	Showing the specific name of a class and relationship just when the user needs it, in order to reduce clutter.
Relate	See how different items of the query are related to each other.
Properties selection	
Overview	Provide an overview of the current query including classes, relationships, properties, and order of addition.
Zoom	Allow the user to focus on items of interest, providing more information upon interaction.
Filter	Allow the user to filter out information, such as properties that are not relevant to his/her question.
Details-on-demand	Show only those properties relevant to the selected entities.
Relate	Help the user relate the selection of data with the whole -the ontology schema- it belongs to.

meaningful for the user and aids in the task of comparing the data.

Moreover, the issue of providing multiple visualizations to perform several different tasks in the same view is addressed by Silva *et al.* in [25], where a taxonomy for classifying tasks and a model for visual analytics dashboards are proposed.

We designed our workflow to promote a top-bottom strategy for higher-level tasks, which translates into different views for each one of them, providing first the overview of the ontology, then attribute details as needed, and lastly, retrieving data for a specific subset of the data. Meanwhile, as described in [25], the exploration of the data, is a bottom-up approach that starts with a selection of data, its exploration involving the three fundamental tasks shown in [24] and mentioned previously, and a refining process that involves reconsidering the subset data chosen and the visualizations created until the research goal is fulfilled. The workflow is detailed in the following subsections.

A. SELECTION OF SOURCES

The sources involved in the analysis are: 1) the data endpoint, which exposes the triplets; and 2) the ontology schema, which contains the definition of the entities and the relationships present in the data. Either of these two resources may not be publicly accessible; this may happen due to privacy issues, security concerns, or even if it is a product in development. Moreover, it is common practice to develop a domain-specific ontology for tripletstore datasets. For this reason, it is important to provide a flexible alternative that allows the use of published, private, or local datasets and RDF schemas. The approach we propose allows the user to enter the URL for the endpoint, and uploading a local file with the ontology schema. This allows the endpoint to be local or external to the machine, and publicly accessible ontologies can be downloaded directly from the browser.

B. DATA SELECTION AND RETRIEVAL

Due to its complexity, the selection and retrieval of RDF data can be intimidating to users. The main cognitive load in this task is knowing in advance what information is available,

what data provides this information, and the specific syntax of the query language used. Such needs are often unfulfilled at the moment of consulting the database, for example, due to an exploratory motivation or a lack of understanding of the data structure (the ontology in this case).

The process of creating the queries benefits from a visual approach that abstracts query languages. However, in order to lower the complexity of this initial step even more, it is necessary to reduce of the number decisions to be made [26]. For example, traditional retrieval using query languages not only involves specifying graphs, entities, and properties, but also any aggregation or filter needed. However, these later aspects are better set when the user has had the opportunity to explore the data.

One key difference with other approaches (interactive or traditional query-based) is the less amount of understanding about the data needed, regarding what properties are available or the namespaces used. Selecting just unfiltered and unaggregated data lowers the overhead and simplifies this step, postponing filtering to a later phase when data can be seen and understood, and reducing the number of decisions to make.

This task benefits too from applying Shneiderman's mantra: 'overview first, zoom and filter, then details on demand' [26]. While having all the information available at once may seem to provide more informed decisions, ontologies can grow complex, and properties may be numerous enough to obscure the view, making the visual abstractions too difficult to be understood or used. Therefore, data selection can be divided into two different steps: class and relationship selection, and properties selection, each of which should take into account the advised task-domain information actions described in the task by data type taxonomy [26]. They represent the seven main abstract actions performed in an interface, and translate into different features depending on the main goal of the view. We provide a set of distinctions for each of the two steps in Table 1. Additionally, the implementation should allow for undoing selections made by the user, and extracting sub-collections that can be stored for later use and shared with other users; for data selection, this would be in the form of queries.

C. VISUAL EXPLORATION OF DATA

The different types of data retrieved through SPARQL require using different visualization techniques. Moreover, the combined use of these techniques is what helps most to gain insights into the retrieved data. Dashboards allow for such combined use with interactive spaces where different visualizations can be created, arranged, and manipulated.

They promote an exploratory approach that would not be possible with a traditional approach. Filters and aggregations can be created visually, rather than relying on the user’s understanding of the data to plan such selections in advance.

Examples of online data dashboards are Google Data Studio⁴, Visual.is⁵, and Grafana⁶. However, these tools do not allow for creating queries visually, a feature that helps with the effective use of RDF for analysis.

IV. ITERATIVE DESIGN

It is important to take into account how different target groups of users can introduce their own needs, influenced by their research objectives and mental models regarding data presentation and tool interaction. In our case, we used an iterative process which helped to introduce changes to meet user’s expectations and allowed for reducing friction in the user experience. It also provided information about the mental process followed by users when performing the tasks; this is often not expressed explicitly, but it can be identified by how they use the tool. An example of this insight is detecting an interest in the results of building visual queries. This leads to a feature for displaying the SPARQL query built interactively, an appreciated addition useful for understanding how interactions translate into a database query.

A. EVALUATION THROUGH THE DESIGN AND IMPLEMENTATION PROCESS

We used usability testing to drive the design and implementation process towards a more successful product, involving ‘representative users attempting representative tasks in representative environments’ [27]. Each of the two phases had test environments and looked towards obtaining different insights. Evaluation during the design was carried out using low-fidelity prototypes, which sought to adapt the workflow to the user’s needs. Additionally, we used cognitive walk-throughs for the ongoing evaluation in the implementation phase, with the intention of detecting usability issues regarding the tool’s ease of use, similar to the work done in [28]–[30].

1) DESIGN

Evaluation in the early phases of design and development (known as formative evaluation [31]) was done using low-fidelity interfaces; therefore, it did not influence the user response much and made it easier to make changes.

⁴<https://datastudio.google.com/>

⁵<https://visual.is>

⁶<https://grafana.com/>

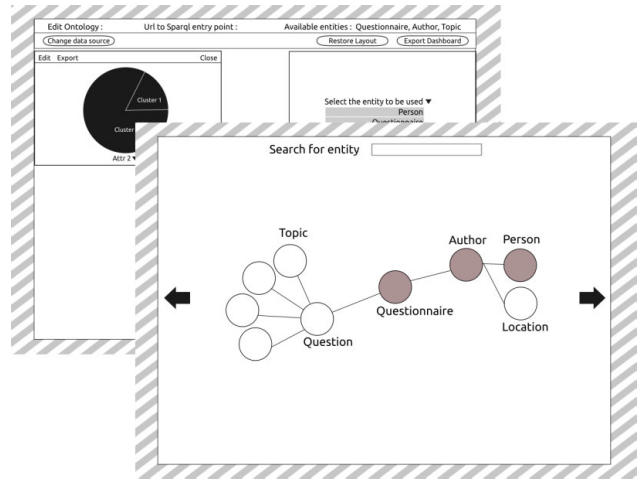


FIGURE 2. Screenshots from the Invision App platform showing two of the designed views or the screen mock-up.

TABLE 2. Tasks performed by users during the paper prototyping evaluations.

Task	Description
1	Select your sources for the data and the ontology.
2	Tell how many different entities there are in the ontology.
3	Select what data you want to explore.
4	You now have some data selected. Can you add a visualization for it?
5	How are entries of a set entity distributed by some attribute? How would you get that information using the tool?

We used paper prototyping during this early stages of the project as it is an UX design evaluation tool that helps iterating quickly, easily and inexpensively through ideas and concepts. This approach is effective if it meets certain conditions, such as having at least 3-4 people for evaluation. The first of the two prototypes was drawn with a graphic editor (Inkscape), printed to paper, and cut into small components which were moved according to the user’s interactions. Additional components were drawn on the fly to try new approaches for each task.

A higher-fidelity screen mock-up (see Figure 2) was done using the InvisionApp (<https://www.invisionapp.com>) tool. This second version of the paper prototype⁷ was then used on the second day with the same group of people, and it was provided asonline resource to a second (remote) group using the comment utility of the platform as a way of communicating issues.

Both versions of the prototype were designed to be used in sessions of less than 15 minutes and focussed on the set of tasks listed in Table 2. An additional set of questions (see Table 3) was made to promote the use of other functionalities.

⁷Publicly available at <https://invis.io/KTOTBEJ458S>.

TABLE 3. Additional questions made during the paper prototyping evaluation.

Question	Description
1	What entities are you going to start your exploration with?
2	Among all the data you have selected, which entity do you want to use for the visualization?
3	What is your desired visualization?
4	The pie chart may have given an overview of the distribution. Try to add another view to show the same information in another fashion.
5	Now that there are multiple visualizations, try redistributing them.

2) IMPLEMENTATION

Due to limitations on travel distances and schedule flexibility, remote usability testing was carried out after each subsequent iteration of the tool implementation. Each evaluation was carried out with a varying five-person group, with the focus on a specific set of tasks that involved using most of the tool’s available functionality.

In addition to information regarding the success of performing said tasks, more qualitative and subjective data regarding the ease of use and usability was obtained in the form of comments. This extra information not only allowed for evaluating evaluate how well the tool performed, but also understanding other issues related to the users’ own mental models and knowledge. An example of this kind of information was the need for a bubble graph because users found it easier to interpret the streamgraph.

B. ITERATIONS

The development cycle covered four different iterations where changes were made according to the issues identified in the intermediate evaluation. The collected feedback was analysed carefully to avoid observations turning into reports and key features changing without the needed impact assessment. The main usability issues and the changes that were done in each iteration can be seen in Table 4.

V. TOOL IMPLEMENTATION

An RDF data retrieval and visual analysis tool was implemented to demonstrate how the workflow could be applied, and the needs previously described could be approached. This tool can be accessed through the following URL: <https://exploreat.acdh-dev.oeaw.ac.at/exploreat-dh-dashboard/>. It consists of three main pages: the landing page, the application itself and a help page where the usage of the tool is described.

As previously discussed, the proposed workflow can be implemented for any kind of data source, retrieval strategy, and visual catalogue. In this specific implementation, data will be accessed from SPARQL endpoints, an ontology schema uploaded from the user’s machine will be used to query the data, and a set of nine different visualizations will allow the visual exploration. This tool has been implemented with web technologies, helping with its distribution and testing. The development of the tool is framed in the exploreAT! project, which assesses the exploration of digital data regarding the German language in Austria. Following the proposed workflow, the application comprises three screens, each of

TABLE 4. Usability issues identified and changes made in each of the implementation iterations.

Observed issue	Actions to improve the interface
First iteration	
The user expected data to follow a numerical or alphabetical order while the displayed data followed the retrieval order. Users found it harder to use visualizations with longer text in the labels. Relying just on interactive filters was cumbersome for some users. There was in interest on the effect of interactions in the resulting query.	Ascending numerical or alphabetical order was used by default with an option to change the sorting. Labels were truncated to a fixed length by default, and shown complete in tables and upon mouse hover. Textual filtering was implemented for easier searching by attribute and value. The generated SPARQL queries are shown through a control in the top bar of the application.
Second iteration	
Complex visualizations were less understood and, thus, not used frequently. Users found distracting having to restart the app for a new exploration. Less-experienced users found it difficult to deal with RDF-specific aspects, such as ontology prefixes and namespaces.	Textual and visual help was provided in the visualization selection control. All views were given controls for going back and forth in the workflow. The ontology prefix gets inferred from the ontology schema, and RDF-specific details were hidden to aid novel users with focussing more on the data.
Third iteration	
Users found text-based controls explanatory enough for complex tasks such as modifying a hierarchy in circle-packing graphs. More explanations were needed when first using the interface. Users expected URLs appearing in text results to be navigable.	More illustrative controls were implemented, showing their effect on the visualization. A help page was added, which explained the use of each visualization and showcased different use cases. URLs were made navigable by click interaction.
Fourth iteration	
Users tried refreshing the page to see changes when the ontology processing time was too long. Users found it confusing to have different available visualizations for each data selection.	A loading widget was added, showing the progress and the current task being done for the ontology load. Visualization descriptions now include a requirements section, showing unfulfilled ones for each unavailable visualization.

which addresses one of the three tasks identified earlier. The tool has a top bar with controls to look at the sources and data selected, and to undo actions taken while using of the application, and a central section which holds the view for the specific task.

A. INTUITIVE VISUAL QUERIES

Querying the database is an important first step, which determines the information that the user will be able to gain. To this extent, our goal in the present proposal is to help the user create this query easily and consciously. We start with the naive idea of what the user will get after the retrieval: a subset of the whole database which, indeed, will have its underlying relationships and properties given by the same ontology. Thus, making the query visually similar to the node-link diagram that shows the ontology allows the user to understand the relationships between the ontology, the query, and the retrieved data.

A common problem of force-directed node-link diagrams is how poorly they scale with larger amounts of information to be displayed. Therefore, the ontology only shows the classes and relationships, while class-specific properties are only shown when the user adds a class to the query. Both class names and properties (including relationships and attributes) are shown in a simplified form, without the namespace to make this large amount of information easier on the eye, especially for entry-level users.

The nodes' sizes in the node-link diagram are used to inform the user about the number of instances for the different classes, as proposed in [5]. In the process of creating the query, the user starts with an initial node selection, and subsequent nodes that are added by clicking in the available links (in blue). This approach helps the user make a query that is supported by the ontology without having to actively take into account link directionality.

In the lower part of the screen, a similar node-link visualization shows the subset of the classes selected, and the attributes that each of them have. By clicking the attribute names, they are added to the query, and highlighted in green. These two views (see the interface on Figure 3) allow to perform each of the seven actions described in [26], and provide an overview of the ontology that shows details upon hovering and allows to filter through relationships, adding classes for property selection and undoing previous actions. It allows too to relate the query made with the whole it belongs to by maintaining the same node-link representation which makes use of the mental model the user has for recognizing hierarchical data.

B. VISUALIZING DATA

The dashboard page provides an expandable work space, which is used for placing new visualizations. The used window system allows for dragging the frames by the top bar, and resizing them by clicking and dragging in the bottom-right corner. This allows the user to create the layout that best fits his/her research needs. An always present window can

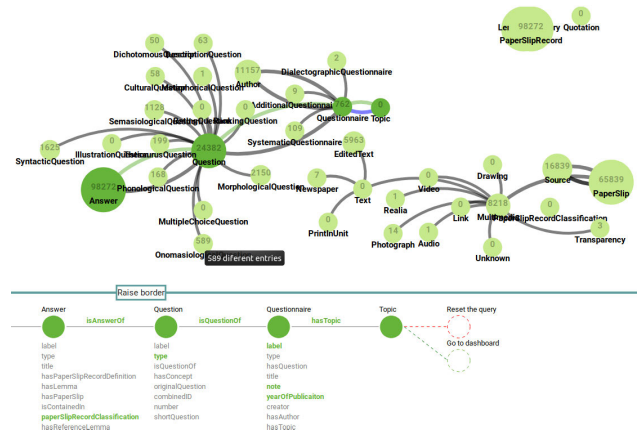


FIGURE 3. Data selection screen. Screenshot with the Oldcan ontology loaded (see use case in Section VI-A).

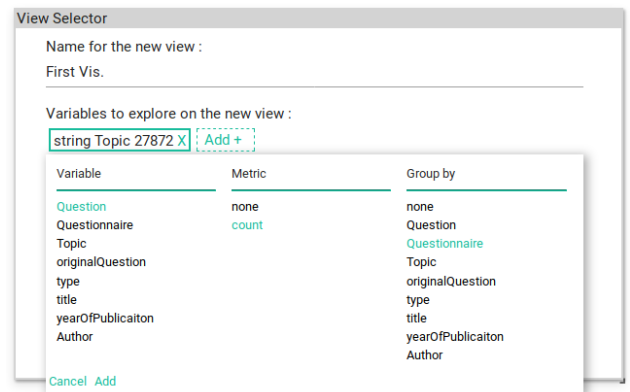


FIGURE 4. Variable selection for Vis creation. Screenshot from the tool.

be used to give a name to the visualization and select variables or aggregations (see Figure 4). After data is selected, a subset of visualizations is presented to the user; a different set of visualizations are proposed based on the type, cardinality and number of variables selected⁸. A total of nine different visualizations have been implemented in the tool: table, bar chart, pie chart, parallel coordinates, violin plot, jitter violin plot, circle pack diagram, streamgraph and bubblechart. An additional text field was also provided through which the user can filter by any combination of desired variables.

In order to facilitate the use of more complex visual techniques for novel users, the selection screen shows a representation of the resulting visualization and a textual description that displays upon hovering and explains the use, result, and possible configuration of the visualization.

VI. CASE STUDIES

This section presents two cases of study to demonstrate how the proposed workflow can facilitate both retrieving and filtering, and exploring the data. The first case of use will

⁸The reasons for a visualization not being eligible can be accessed at any moment, next to the visualization miniatures

focus on easily filtering through data without understanding or using complicated SPARQL queries, and the second one will focus on the use of an ontology to access the data through a specific point of view.

A. CASE STUDY: OLDCAN DATASET

For this particular use-case, the RDF-explorer is employed to explore data from the Ontology for Lexicographic Data Collection and ANalysis (OLDCAN) [32], [33]. This data is derived from a non-standard legacy language collection (Datenbank der Bairischen Mundarten in Österreich/-Database of Bavarian Dialects in Austria [DBÖ]) [34], [35] containing records for questionnaires, related answers on paper slips, excerpts of vernacular dictionaries and folklore literature, and multimedia content covering a period from the early 20th century until the present.

The OLDCAN ontology is organised around the questionnaires used to collect traditional, cultural and lexicographic data. Thus, it contains the questionnaire entity at the heart of the ontology and links the questionnaires to Questions, Authors, Collectors and Respondents. Each of these entities is linked to the questionnaires using object properties, including `hasQuestion/isQuestionOf`, `hasAuthor/isAuthorOf`, and other object relationships. Furthermore, questions are linked to different entities including Answers, PaperSlips, Collectors etc. Answers, in turn are linked to Lemmas. Most of these entities contain their original form in a multimedia format, such as images, which makes them interconnected with a specialized entity called Multimedia. The ontology links the core entities of the collection and further provides several data properties with the respective domains and ranges.

A detailed description of the ontology is available in [16], and the current version of the OLDCAN ontology can be accessed at <https://explorations4u.acdh.oeaw.ac.at/ontology/oldcan.owl>.

OLDCAN is used to semantically annotate the collection to create approximately 3 million of interconnected RDF triples. These triples are generated based on R2RML mapping of a relational database, which is stored in a MySQL database. This dataset is made available as a REST API, a SPARQL endpoint and a downloadable file⁹.

For this use case, four different questions will be answered with the use of the platform:

- What information related to the questionnaires and answers is available for exploration?
- What information can we retrieve from the collection on gender distribution?
- How are questionnaires distributed through time?
- How was the collection conceptualized in terms of cultural knowledge? Which aspects of cultural knowledge can be explored and analysed?

The approach to do this analysis follows the proposed workflow. An initial entity and properties selection is done guided by the visual interface, followed by the creation of

views, which can be used to filter the data and answer the questions. The above questions are needed for both the retrieval of both detailed information regarding location and topics for questionnaires, and aggregations of the data based on the publication date and the author's gender. This is not possible with a single query using plain SPARQL. Moreover, using a complete SPARQL query with filters for a specific question limits the analysis that can be done for the rest of them. However, the proposed approach delays filtering until details for the retrieved data are available, and data visualizations have been applied. This helps to assess which questions can be answered with the current data, thereby allowing for a better reasoning about which filters and aggregations are more useful. Another advantage of this technique is how it leverages the retrieved data, allowing the user to focus on different aspects depending on the current research question. This is not possible if multiple complex queries have to be done.

1) SOURCE SELECTION AND DATA RETRIEVAL

The very first step for using the dashboard is selecting a local RDF schema (OLDCAN) and the SPARQL endpoint <https://dboe-jena.acdh-dev.oeaw.ac.at/dataset.html?tab=query&ds=oldcan-2> that will be queried. These sources will be used when selecting both the entities and attributes. These sources allow performing the two basic tasks of data retrieval and analysis, which involve the selection of graphs, entities, properties, filters, and aggregations for the later detailed listing and information summary. This process is greatly benefitted from the strategy of making simple visual queries in the first place, and applying filters and aggregations through visualizations. Selecting these entities and attributes is done by clicking on a first *Answer* entity, and then proceeding by clicking in the available (blue) links in order to add the rest of the entities. Once added, the visual query in the lower part of the screen allows for adding attributes to the selected entities, and proceeding with the retrieval of the data by clicking on 'Go to dashboard'. Not only is the retrieval phase simplified using visual tools, but it allows for understanding how much information is available for each entity and the properties that it holds. Thus, it allows for answering the first questions. For instance, each questionnaire is related to the entities Author, Question, and Multimedia; it holds information regarding the year of publication, the type of questionnaire, its topics, and other information regarding its physical form, such as its label, notes on the paper, and notes that could have been written on top of it.

With other tools, the user needs to know this information in advance to retrieve the data. However, the provided interface already shows what can be queried to the endpoint and needs no further filtering. We believe this strategy makes the selection easier and shorter, lowering the entry barrier for novice SPARQL users and newer datasets. The example query shown in Figure 5 would translate into the following SPARQL query:

⁹A REST API is available at <https://dboe-jena.hephaistos.arz.oeaw.ac.at>

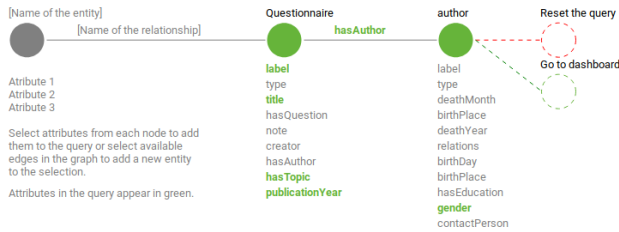


FIGURE 5. Visual query for retrieving the label, title, topic, and publication year of questionnaires, and the gender of the author.

```

1
2 PREFIX oldcan:
3   <https://explorations4u.acdh.oeaw.ac.at/ontology/oldcan#>
4 SELECT ?Answer ?Question ?Questionnaire
5   ?Author ?Answer ?type ?hasTopic
6   ?Questionnaire ?note ?label
7   ?paperSlipRecordClassification
8
9 WHERE{
10  GRAPH ?a {?Answer
11   <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
12   oldcan:Answer}.
13  GRAPH ?b {?Answer oldcan:isAnswerOf
14   ?Question}.
15  GRAPH ?c {?Question oldcan:isQuestionOf
16   ?Questionnaire}.
17  GRAPH ?d {?Questionnaire oldcan:hasAuthor
18   ?Author}.
19  GRAPH ?e {?Answer
20   oldcan:paperSlipRecordClassification
21   ?paperSlipRecordClassification}.
22  GRAPH ?f {?Question
23   <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> ?
24   type}.
25  GRAPH ?g {?Questionnaire oldcan:hasTopic
26   ?hasTopic}.
27  GRAPH ?h {?Questionnaire oldcan:note ?note}.
28  GRAPH ?i {?Questionnaire
29   <http://www.w3.org/2000/01/rdf-schema#label> ?label}.

```

2) INTERACTIVE EXPLORATION AND FILTERING OF DATA

The dashboard screen shows up after data has been retrieved from the SPARQL endpoint, providing the user with an initial view for selecting variables and creating visualizations. These variables are the retrieved classes, relationships, and properties, which, for the query in Figure 5, hold information regarding the questionnaire topics, types and classification, and author’s gender.

The first view used is a bar chart that allows to easily see how much data there is for each topic, and what cultural topics are present in the collection. We can also see which topics were deemed important at the time of conceptualization by having more than one questionnaire for a specific topic. All visualizations allow filtering by utilizing click interactions on their elements (such as a bar or a section in a bar chart or a pie chart, or an area in a streamgraph), applying a filter that selects just the elements that fall under that group. The created filters affect all the views, making them update and possibly enabling visualizations that were not eligible because of too much available data.

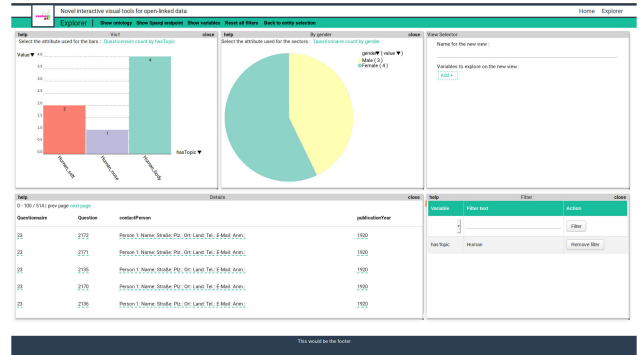


FIGURE 6. Dashboard screenshot showing the result of using a filter component for textual search.

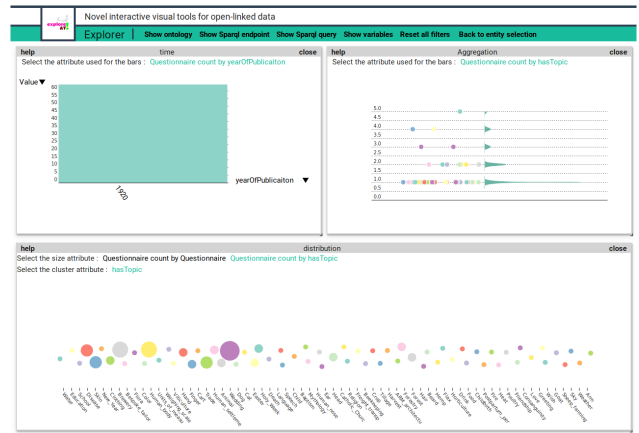


FIGURE 7. Dashboard screenshot showing the distribution of questionnaires according to topic and year.

We focus on *human*-related topics for this specific exploration. Questionnaires are selected by creating a filter view and searching for the term *human* in the *hasTopic* property. After a filter is created, views are automatically updated to use the resulting data subset. Figure 6 shows a workspace with a pie chart showing gender information, and a table with details about the author, year of publication and questions per questionnaire.

Further filtering can be done using the author’s gender, either by clicking the sectors of the pie chart or searching in the filter component. Using SPARQL would require adding the following lines to the query:

```

[language=Bash, label={vcfignore}]
2 filter contains(str(?topic), "string").
3 filter equals(str(?gender), "male/female")

```

For this specific example, the interface updates to show just questionnaires related to *human* topics and published by female authors and, thus, shows the 243 questions that belong to the four questionnaires published in 1920. These questionnaires and questions could be read and further analysed by clicking on the entry in the table.

As shown, the questionnaires have a topic assigned to them through the *hasTopic* property, which allows filtering through them in order to get relevant answers for said topic. However, it also allows to get a sense of how these questionnaires

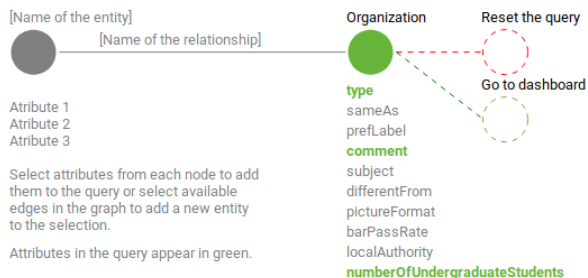


FIGURE 8. Visual query to retrieve data for exploring university related information.

were conceptualized. In Figure 7 a three-views dashboard layout was used to filter the year 1920 through a bar chart, the distribution of document count by the topic, and each specific group of questionnaires for the topic. A total of 64 questionnaires published in 1920 hold 67 different topics, some of which are shared across questionnaires; this is the least frequent, as the top-right jitter violin plot shows: each dot represents a topic, while the area to the right represents the number of topics that appear the corresponding amount of times. This second visualization shows how most of the topics are related to a single questionnaire, but the topics of *wedding*, *human body*, *bespoke tailoring*, *trade* and *disease* do appear in more questionnaires, indicating a major interest in these topics during 1920 in the corpus that these questionnaires and questions make up.

The third view is a bubble graph in which two aggregations have been selected. One aggregation allows weeing topics represented by bubbles whose size encodes the number of questionnaires where it appears, and a second aggregation that shows the questionnaires for each topic. This allows to filtering by a specific topic and switching to the second in order to see which questionnaires are involved; for instance, questionnaires from 7 to 11 have information related to the topic *wedding*.

B. CASE OF STUDY: BBC PROGRAMMES AND MUSIC

The exploration of SPARQL data endpoints is done with a reference RDF/OWL schema, which is used to create the previously discussed visual query. This second case of study demonstrates the use of the tool to explore endpoints for which no ontology has been specifically designed; rather, already existing ontologies are used to model the data.

Data regarding TV and radio programmes broadcast by the BBC are accessed with the endpoint provided at the datahub¹⁰. The data endpoint is published alongside with an RDF schema in turtle format which, in our proposal, is not supported. However, this ontology is partially based on the friend of a friend (FOAF) ontology¹¹, which can be used to query the endpoint. This shows how, depending on the ontology, the available questions change to give more flexibility.

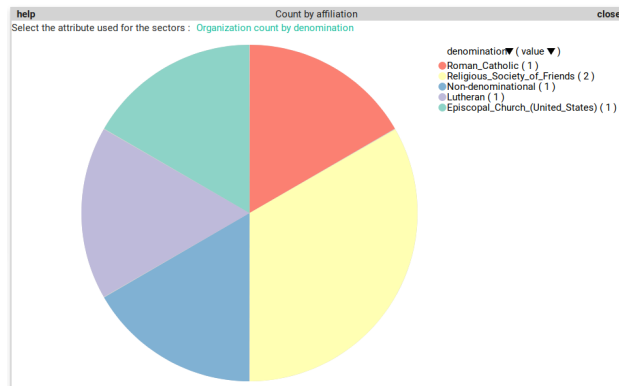


FIGURE 9. Religious affiliation for BBC programme-related organizations.



FIGURE 10. Distribution of organization founding by the year.

Next, the organizations available in the dataset are explored using the query shown in Figure 8. This class has a manageable amount of information and a wide range of attributes, which can provide information regarding organizations that were involved in the BBC programmes. The questions in this case study were:

- 1) There is information regarding affiliation. What are these and how much information is there regarding affiliation?
- 2) How many undergraduate students do BBC programme-related universities from America have? What are their distributions?
- 3) Explore how the organizations were founded.

1) RELIGIOUS AFFILIATION OF ORGANIZATIONS

The use of a pie chart (Figure 9) provides a summary of the organizations that have the information that was queried; a total of five organizations out of the 3,693 available in the dataset have a value for this attribute.

This information can be expanded by making use of a table that provides per entity information, leading to understanding that the majority are related to academia: Episcopal Collegiate School, Friends School of Minnesota, Brookling Friends School, Michigan Lutheran Seminary, and Avila College. Additionally, as data in the Semantic Web is defined

¹⁰<http://lod.openlinksw.com/sparql>. Visited on 1st April 2019.

¹¹<http://xmlns.com/foaf/spec/>. Visited on 1st April 2019

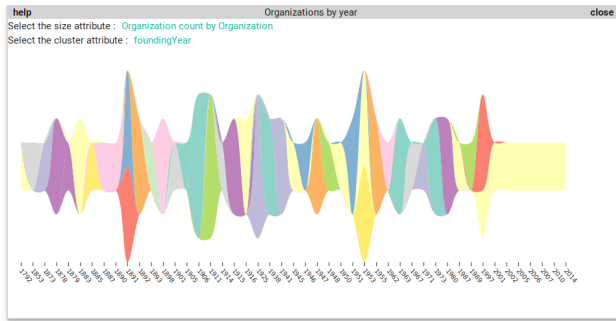


FIGURE 11. Companies distributed by their founding year on a horizontal timeline.

TABLE 5. Tasks performed by users during the evaluation.

Question
1. Tell how many different organizations there are in the database.
2. I found the system unnecessarily complex.
3. Use a visualization to represent the amount of web pages related to the lion's species.
4. Find the founding year for the religious university with the highest number of students.
5. Use a visualization to see how organization founding changed through time.

with the use of URIs, further analysis can be done by accessing the specific resource by clicking on it in the table.

2) AMERICAN UNIVERSITIES' SIZE

Interactive filtering can be hard when the attributes' cardinality is too high for the effective use of visualizations. This is the case when selecting universities by their type of organization. For this task, we can use a table as an auxiliary view to get a sense of the retrieved data and then search for entities containing 'American' in their description using the filter component.

A total of 21 universities from America are accessible from the SPARQL, with the number of undergraduate students ranging from 83 (American Public University System) to 18,656 (Empire State College). By looking at the visualization, two organizations with an unusual student count can be spotted: the American Public University System, an online learning institution offering associate, bachelor's and master's degrees, and other educational programmes. The second University is a non-American university present in the current set because it has 'America' in its description; this university, located in Quezon City, Philippines, was founded by the American colonial government on 18th June 1908, and has a total of 60,889 students: 43,927 undergraduates (the attribute selected in the visualization), 14,777 postgraduates, and 2,185 students with no information.

3) ORGANIZATION FOUNDING

When a SPARQL endpoint is queried for attributes, only entities that have values for such predicates or objects are

TABLE 6. Completion rate and execution times for each task done in the evaluations.

User	Task 1	Task 2	Task 3	Task 4	Task 5
User 1	8"	8'	7'30"	4'	4'
User 2	6"	7'20"	5'	5'2"	3'
User 3	3"	7'10"	5'	3'24"	3'23"
User 4	4"	9'10"	8'30"	5'30"	6'15"
User 5	2'	12'43"	6'10"	6'	5'40"
Mean	28"	8'53"	6'26"	4'53"	5'17"

TABLE 7. Questions used in the satisfaction questionnaire.

Question
1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think I would need the support of a technical person to be able to use the system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

TABLE 8. System-user evaluation results.

Question	Mean	SD
1.	3.8	0.75
2.	1.8	0.75
3.	3.4	0.45
4.	1.8	0.75
5.	4.6	0.49
6.	1.8	0.4
7.	4	0.63
8.	1.4	0.49
9.	3.2	0.4
10.	2.2	0.79

TABLE 9. SUS scale results per evaluation.

Scale score					Mean	SD
80	80	70	70	75	75	4.47

retrieved. In this third example, common attributes are used in order to get information regarding how these organizations are distributed and how they were founded.

The jitter violin plot in Figure 10 shows how the number of founded organizations range between 1 and 6 for the majority of years. Further information could be extracted if we made use of the 'foundingYear' variable to see its temporal evolution, a task that can easily be done with streamgraphs.

In this case, organizations are shown alongside their founding year in a visualization where coincident organizations will appear stacked. Two main time periods can be seen when a larger number of organizations were founded: from 1950 to 1978, and from 1990 to 2004. If we wanted to formulate the same question just for companies, we can use a textual filter with the 'type' variable. Figure 11 shows the

TABLE 10. Feature comparison between the proposed tool and previous work.

Feature	Our tool	VizLOD [4]	Payola [2]	KC-Viz [7]	VOWL [13]	LodLive [14]	AVOnEd [15]	Concept lattices [5]	Vis Wizard [6]	[23]	[25]
Ontology visualization and data retrieval											
Uses ontology	✓	✓	-	✓	- ✓	-	✓	-	-	-	✓
Uses RDF data	✓	✓	✓	-	-	✓	-	✓	✓	-	-
Ontology schema visualization	✓	✓	-	✓	- ✓	-	✓	-	-	-	✓
Ontology shown along with data	✓	✓	-	-	-	-	-	-	-	-	-
Handle equivalent classes	✓	✓	-	-	✓	-	✓	-	-	-	✓
Handle subclass properties	✓	✓	-	✓	✓	-	✓	✓	-	-	✓
Class selection	✓	✓	-	-	✓	-	✓	-	-	-	✓
Property selection	✓	-	✓	-	✓	-	✓	-	-	-	✓
The user can query data	✓	-	✓	-	-	✓	-	-	✓	✓	✓
Data exploration and visualization											
Raw data visualization	✓	✓	-	-	-	✓	-	✓	-	-	-
Multiple visualization options	✓	-	✓	-	-	✓	-	-	✓	✓	✓
Simultaneous views	✓	-	-	-	-	-	-	-	✓	✓	✓
Linked views functionality	✓	-	-	-	-	-	-	-	-	-	✓
Textual filters	✓	-	✓ ^a	-	✓ ^b	✓	-	-	-	-	✓
Interactive filters	✓	-	-	-	-	✓	-	✓	✓	✓	✓

a) Done in the process of querying.

b) Filtering is done on the ontology classes, not retrieved data.

result of creating this filter, allowing for completing the initial task of ‘exploring how organizations were founded’. No more than two companies appear stacked, being the second time in the year 1953; within the first range mentioned. During 1953, the two founded companies were *ABC-CLIO* and *Litton Industries*, whose information can be expanded if the ‘comment’ property is selected as part of the visual query or by clicking in such entity in a table view.

VII. EVALUATION

A. SUMMATIVE EVALUATION

Usability testing is a process that for obtaining a measurable and comparable score for the tool’s performance (in terms of effectiveness, efficiency and satisfaction). The evaluation protocol used in this project consisted of 50-minute sessions with different users where the time to perform basic tasks with the tool was measured, the users’ impressions while using the tool were gathered with a concurrent Think-Aloud Protocol, and the perceived satisfaction was quantified using a System Usability Scale (SUS),

The evaluations were carried out with five subjects, ranging from 22 to 50 years old, from different study areas: computer engineering students, humanist graduates, and management. All of the intervening users were familiar with some of the technical terms related to databases, although only one had previously worked with SPARQL.

We provided a brief introduction during the first 10 minutes, covering the tool’s usage, the dataset being used, and the tasks that they were going to perform. This was followed by 10 minutes of free time for exploration. After this period of time, they were given 30 minutes to do each of the tasks shown in Table 5, which were timed and can be seen in Table 6. These tasks covered understanding the ontology representation, and four different explorations that

involved filtering, aggregating data, and changing the data selection.

We observed that the first two tasks covering the whole workflow (task s#2 and #3) took the longest for users to complete (see Table 6), while the remaining tasks were done faster. Moreover, users changed the data selection more often in subsequent tasks, a sign indicating that perhaps an iterative exploration was more efficient for these tasks¹².

As expected, most of the users (four out of five) used simple visualizations to complete the third task. However, all the users tried to use more visualizations during the remaining exploration, involving the custom jitter violin plot and the bubblegraph. We associate this observation with an increase in the users’ confidence using the tool.

After the exploration, we asked users to answer a satisfaction questionnaire following the System Usability Scale [36] (see the questions on Table 7). Applying the SUS metric to the obtained results (see Table 8) provided us with a score that compares favourably with the standard average SUS score (68). We obtained a mean score of 75 (see Table 9), which corresponds to an adjective score of *good*; thus, scores between 68 and 80.3 were rated *good*, and scores higher than 80.3 were rated *excellent*.

Along with the questionnaires, users were asked about their experiences using the tool. A common feeling was that getting started with the tool was not immediate, although performing the tasks was easy enough to get used to it. They agreed on the practicality of the tool, although a common issue raised regarding the filter component was that the users found it misleading to locate it under the visualization selection view. They also expected the filter to show results instead

¹²Further research would be needed to decide if the increased performance could be just because of adaptation to the interface

of applying it to other views. However, they easily understood how it worked and applied it to the use cases.

B. COMPARISON WITH OTHER TOOLS

This subsection covers the comparison between the implemented tool and previous work done on the subject. Given the initial goal of combining ontology visualization, data retrieval, and data visualization, only a subset of the reviewed work covers a similar workflow to the proposed. Because of the inability to use these tools, due to unavailability or discontinuation (hence, they could not be used to replicate the tasks), a feature-wise comparison is provided in Table 10.

It can be seen how few of the reviewed works have covered most of the evaluated features. The main reason for this is the gap in the existing literature to tackle two different problems at the same time; although, much work has been done on ontology editing and visualization, such as [4], [5], [7], [15], and RDF data visualization [2], [6], [14]. However, fewer have covered the complete workflow, starting with ontology exploration, followed by query formulation based on the learned information, and a later exploration of the retrieved data, making use of different data visualizations and high-level tasks such as linked-view manipulation and interactive filtering (described in [26]). The approach proposed in [25] covers most of the features considered, and uses semantic information (described by an ontology) to help with the process of viewing a selection for data visualization. However, it starts from a data source that is ready to be visualized. This is in contrast to integrating the query phase in an assistive manner into the tool itself.

VIII. CONCLUSION

Here, a new approach for LOD retrieval and analysis has been presented. The approach moves the focus from the selection and retrieval to the exploration and analysis of the data. Part of the decisions made when creating queries with traditional tools are moved to the step where the user has access to the data and is able to make better decisions.

We successfully implemented a tool that demonstrates how relying just on visual tools, is sufficient to obtain information from data when retrieval would have needed of a complex query. The separation of concerns among views allowed for making use of techniques specific to each of the tasks: it enables seeing how an ontology's topology maps to the data stored in a SPARQL endpoint and making visual queries using that same mental model. It also provides a configurable dashboard interface to perform the next -exploratory- phase of the process. This strategy has been useful for the implementation of RDF explorer, and it can also be helpful too for future development of other exploratory tools that need different approaches in each step of the process. Finally, the tool was validated by a final user study that ensured that the proposed design aligned with the users' mental model and permitted a correct exploration of the test dataset.

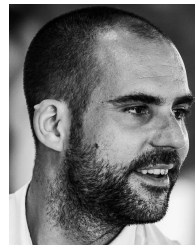
REFERENCES

- [1] A. Benito-Santos and R. T. Sánchez, "Cross-domain visual exploration of academic corpora via the latent meaning of user-authored keywords," *IEEE Access*, vol. 7, pp. 98144–98160, 2019.
- [2] J. Klímek, J. Helmich, and M. Necaský, "Payola: Collaborative linked data analysis and visualization framework," in *The Semantic Web: ESWC 2013 Satellite Events*. Berlin, Germany: Springer, 2013, pp. 147–151.
- [3] A. Vázquez-Ingelmo, F. J. García-Peñalvo, and R. Therón, "Information dashboards and tailoring capabilities—A systematic literature review," *IEEE Access*, vol. 7, pp. 109673–109688, 2019.
- [4] C. Anutariya and R. Dangol, "VizLOD: Schema extraction and visualization of linked open data," in *Proc. 15th Int. Joint Conf. Comput. Sci. Softw. Eng. (JCSSE)*, Jul. 2018, pp. 465–470.
- [5] M. Alam and A. Napoli, "Interactive exploration over RDF data using formal concept analysis," in *Proc. IEEE Int. Conf. Data Sci. Adv. Anal. (DSAA)*, Oct. 2015, pp. 529–538.
- [6] G. Tschinkel, E. Veas, B. Mutlu, and V. Sabol, "Using semantics for interactive visual analysis of linked open data," in *Proc. CEUR Workshop*, vol. 1272, 2014, pp. 133–136.
- [7] E. Motta, P. Mulholland, S. Peroni, M. d'Aquin, J. M. Gomez-Perez, V. Mendez, and F. Zablith, "A novel approach to visualizing and navigating ontologies," in *Proc. Semantic Web (ISWC)*, vol. 7031, 2011, pp. 470–486.
- [8] T. D. Wang and B. Parsia, "CropCircles: Topology sensitive visualization of owl class hierarchies," in *Proc. Int. Semantic Web Conf.* Berlin, Germany: Springer, 2006, pp. 695–708.
- [9] S. Krivov, R. Williams, and F. Villa, "GrOWL: A tool for visualization and editing of OWL ontologies," *Web Semantics, Sci., Services Agents World Wide Web*, vol. 5, no. 2, pp. 54–57, 2007.
- [10] S. Lohmann, V. Link, E. Marbach, and S. Negru, "WebVOWL: Web-based visualization of ontologies," in *Proc. Int. Conf. Knowl. Eng. Knowl. Manage.* Cham, Switzerland: Springer, 2014, pp. 154–158.
- [11] F. J. García-Peñalvo, P. O. de Pablos, J. García, and R. Therón, "Using OWL-VisMod through a decision-making process for reusing OWL ontologies," *Behav. Inf. Technol.*, vol. 33, no. 5, pp. 426–442, 2014.
- [12] M. Dudáš, S. Lohmann, V. Svátek, and D. Pavlov, "Ontology visualization methods and tools: A survey of the state of the art," *Knowl. Eng. Rev.*, vol. 33, p. e10, Jul. 2018.
- [13] S. Lohmann, S. Negru, F. Haag, and T. Ertl, "Visualizing ontologies with VOWL," *Semantic Web*, vol. 7, no. 4, pp. 399–419, 2016.
- [14] D. V. Camarda, S. Mazzini, and A. Antonuccio, "LodLive, exploring the Web of data," in *Proc. 8th Int. Conf. Semantic Syst.*, 2012, pp. 197–200.
- [15] S. Hartmann, F. Hallay, L. Brinkschulte, J. Rebstadt, L. Gesterkamp, A. Enders, N. Kewitz, and R. Mertens, "Aspect-oriented visual ontology editor (AVOnEd): Visual language, aspect-oriented editing concept and implementation," *Int. J. Semantic Comput.*, vol. 11, no. 2, pp. 229–274, 2017.
- [16] B. Johnson and B. Shneiderman, "Tree-maps: A space-filling approach to the visualization of hierarchical information structures," in *Proc. IEEE Vis.*, Oct. 1991, pp. 284–291.
- [17] B. Rodin and D. Sullivan, "The convergence of circle packings to the Riemann mapping," *J. Differ. Geometry*, vol. 26, no. 2, pp. 349–360, 1987.
- [18] B. B. Bederson, B. Shneiderman, and M. Wattenberg, "Ordered and quantum treemaps: Making effective use of 2D space to display hierarchies," in *The Craft of Information Visualization (Interactive Technologies)*, B. B. Bederson and B. Shneiderman, Eds. San Francisco, CA, USA: Morgan Kaufmann, 2003, pp. 257–278. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/B9781558609150500330>
- [19] J. Stasko and E. Zhang, "Focus+ context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations," in *Proc. IEEE Symp. Inf. Vis. (INFOVIS)*, Oct. 2000, pp. 57–65.
- [20] D. Willis, W. W. Braham, K. Muramoto, and D. A. Barber, *Energy Accounts: Architectural Representations of Energy, Climate, and the Future*. Evanston, IL, USA: Routledge, 2016.
- [21] T. L. Saaty, "The minimum number of intersections in complete graphs," *Proc. Nat. Acad. Sci. USA*, vol. 52, no. 2, p. 688, 1964.
- [22] Z. Cai, K. Shi, and H. Yang, "A novel visualization for ontologies of semantic Web representation," in *Proc. Int. Conf. Comput. Intell. Commun. Netw. (CICN)*, Dec. 2015, pp. 1371–1374.

- [23] K. Nazemi and D. Burkhardt, "Visual analytical dashboards for comparative analytical tasks—A case study on mobility and transportation," *Procedia Comput. Sci.*, vol. 149, pp. 138–150, Jan. 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1877050919301243>
- [24] A. Buja, D. Cook, and D. F. Swayne, "Interactive high-dimensional data visualization," *J. Comput. Graph. Statist.*, vol. 5, no. 1, pp. 78–99, 1996. [Online]. Available: <http://www.jstor.org/stable/1390754>
- [25] I. C. S. Silva, G. Santucci, and C. M. D. S. Freitas, "Visualization and analysis of schema and instances of ontologies for improving user tasks and knowledge discovery," *J. Comput. Lang.*, vol. 51, pp. 28–47, Apr. 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1045926X17302458>
- [26] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," presented at the IEEE Symp. Vis. Lang., Sep. 1996, pp. 336–343.
- [27] J. R. Lewis, "Sample sizes for usability studies: Additional considerations," *Hum. Factors*, vol. 36, pp. 78–368, Jul. 1994.
- [28] J. R. Tidal, "One site to rule them all, redux: The second round of usability testing of a responsively designed Web site," *J. Web Librarianship*, vol. 11, no. 1, pp. 16–34, 2017.
- [29] M. Jacobs, I. Henselmans, D. L. Arts, M. ten Koppel, S. S. Gisbertz, S. M. Lagarde, M. I. van Berge Henegouwen, M. A. G. Sprangers, H. C. J. M. de Haes, and E. M. A. Smets, "Development and feasibility of a Web-based question prompt sheet to support information provision of health-related quality of life topics after oesophageal cancer surgery," *Eur. J. Cancer Care*, vol. 27, no. 1, p. e12593, 2018. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/ecc.12593>
- [30] M. Stojmenović, T. Oyelowo, A. Tkaczyk, and R. Biddle, "Building Website certificate mental models," in *Persuasive Technology*, J. Ham, E. Karapanos, P. P. Morita, and C. M. Burns, Eds. Cham, Switzerland: Springer, 2018, pp. 242–254.
- [31] J. Lazar, J. H. Feng, and H. Hochheiser, *Research Methods in Human-Computer Interaction*. San Mateo, CA, USA: Morgan Kaufmann, 2017.
- [32] Y. Abgaz, A. Dorn, B. Piringer, E. Wandl-Vogt, and A. Way, "Semantic modelling and publishing of traditional data collection questionnaires and answers," *Information*, vol. 9, no. 12, p. 297, 2018. [Online]. Available: <https://www.mdpi.com/2078-2489/9/12/297>
- [33] Y. Abgaz, A. Dorn, B. Piringer, E. Wandl-Vogt, and A. Way, "A semantic model for traditional data collection questionnaires enabling cultural analysis," in *Proc. LREC Workshop Linked Data Linguistics (LDL)*, Miyazaki, Japan, 2018, pp. 1–9.
- [34] *Datenbank der Bairischen Mundarten in Österreich [Database of Bavarian Dialects in Austria] (DBÖ)*, Österreichische Akademie Wissenschaften, Vienna, Austria, 2018.
- [35] E. Wandl-Vogt. (2012). *Datenbank der Bairischen Mundarten in Oesterreich (DBÖ). Electronically Mapped dbo@ema*. Accessed: Jan. 17, 2018. [Online]. Available: <https://wboe.oeaw.ac.at/projekt/beschreibung/>
- [36] J. Brooke, "SUS—A quick and dirty usability scale," in *Usability Evaluation in Industry*. London, U.K.: Taylor and Francis, 1996.



ALEJANDRO RODRÍGUEZ DÍAZ is currently pursuing the master's degree in intelligent systems from the University of Salamanca. He is also a part of the Technical and Support Staff with the Visual Analytics Group VisUSAL (within the Recognized Research Group GRIAL), where he aids in the design of new visualization techniques for leveraging the understanding of complex data in digital humanities. His areas of interest are data visualization, process automation, and graphic design. He has special interest in the design of creative and interactive tools for multivariate data.



ALEJANDRO BENITO-SANTOS received the B.Sc. degree in computer engineering from the University of Salamanca and the M.Sc. degree in intelligent systems from the University of Salamanca, in 2016. He is currently pursuing the Ph.D. degree with the Visual Analytics Group VisUSAL, under the supervision of Dr. R. Therón. In 2016, he joined the University of Salamanca, Spain, where he is currently a Research Assistant and a Lecturer with the Department of Computer Science and Automation. He has taught HCI and Introduction to Python Programming for Statisticians at the Faculty of Sciences of Salamanca. He is a member of the Visual Analytics Group VisUSAL (within the Recognized Research Group GRIAL). In his thesis, he applies visual analytics in a broad range of interdisciplinary research contexts, such as the digital humanities, sports science, and linguistics. His research interests include the areas of human-computer interaction, design, statistics, and education.



AMELIE DORN received the B.A. degree in French, Spanish, and Portuguese and the M.A. degree in European studies from University College Dublin (UCD), the M.Phil. degree in linguistics from the Trinity College Dublin (TCD), and the Ph.D. degree in linguistics from the Trinity College Dublin, Ireland, in 2014, focusing on the analysis and description of the intonation of Donegal Irish, a Northern variety of Irish. Since 2015, she has been a Senior Researcher with the Austrian Centre for Digital Humanities of the Austrian Academy of Sciences (ACDH-OeAW). She is with the core unit of Methods and Innovation in exploration space at ACDH, discovering processes to open up cultural collections. She also acts as a Principal Investigator of the ChIA Project—accessing & analyzing cultural images with new technologies, from 2019 to 2021. She has experience and knowledge in open innovation methods and practices from being a selected member of the 2nd Lab for Open Innovation in Science (LOIS II) at the Ludwig Boltzmann Gesellschaft, Vienna, Austria, which is the First Worldwide Training Program in Open Innovation in Science.



YALEMISEW ABGAZ received the bachelor's degree in library and information science and the master's degree in information science from Addis Ababa University and the Ph.D. degree in computer science from Dublin City University, in 2013, focusing on ontology evolution and analysis of impacts of the changes. He is currently a Research Fellow in the area of Semantic Web technologies with the Adapt Centre, Dublin City University. He is also affiliated with the Austrian Centre for Digital Humanities, Austrian Academy of Sciences, and a Principal Investigator of the ChIA Project—accessing & analysing cultural images with new technologies. He is also an Associate Faculty with the School of Computing, National College of Ireland. Before joining Adapt, he was a Postdoctoral Researcher with Maynooth University, from 2014 to 2017, and a Lecturer with Addis Ababa University, from 2006 to 2008. He is mainly interested in topics at the intersection between semantic web technologies, knowledge representation, and natural language processing, including ontology development, ontology evolution, semantic search, semantic publishing, information retrieval, and computational creativity. He has interested in natural language processing and semantic modeling of amharic digital collections. He is serving as a reviewer of different conferences and workshops.



EVELINE WANDL-VOGT is currently an Experimentalist, a Network Facilitator, a Cultural Lexicographer with the Austrian Centre for Digital Humanities, Austrian Academy of Sciences, where she has also been the Founder and a Coordinator of exploration space, working group of methods and innovation, since 2017. The working group is an open space for innovation and experimentation, with the aim of stimulating, designing, enabling, and scientifically analyzing new

forms of knowledge production at the interface of Science, Technology, and Society. Beyond Humanities, methods of Open Innovation in Science, Geography, and Environmental studies as well as Arts and Design are applied. She has a multidisciplinary university background (German Philology, Geography, Informatics, Theater studies, Education Science, and Social Innovation) as well as dedicated experience in lexicography, data curation, and design thinking. She is also a Distinguished Member of the worldwide first Laboratory for Open Innovation in Science, from 2017 to 2018. She has been a Visiting Scholar at the Department for Romance Languages and Literatures, Harvard University, since 2018, as well as a Research Affiliate at the metaLAB, Harvard, and at Berkman Klein Center, Harvard, since 2019. She acts as an European Research Manager and Initiator on various international boards and serves as expert in various global initiatives, mainly in the area of technical and social infrastructures, such as ADHO, ALLEA, COST actions, DARIAH, ECSA, and standardization bodies. She is also a Network Facilitator, passionate to create cross-sectoral, cross-organizational value driven innovation networks of purpose.



ROBERTO THERÓN received the Diploma degree in computer science from the University of Salamanca, the B.S. degree from the University of A Coruña, the B.S. degree in communication studies and the B.A. degree in humanities from the University of Salamanca, and the Ph.D. degree from the Research Group Robotics, University of Salamanca. His Ph.D. thesis was on parallel calculation of the configuration space for redundant robots. He is currently the Manager of the

VisUSAL Group (within the Recognized Research Group GRIAL), University of Salamanca, which focusses on the combination of approaches from computer science, statistics, graphic design, and information visualization to obtain an adequate understanding of complex datasets. He has authored more than 100 articles in international journals and conferences. In recent years, he has been involved in developing advanced visualization tools for multidimensional data, such as genetics or paleo-climate data. In the field of visual analytics, he develops productive collaborations with groups and institutions internationally recognized as the Laboratory of Climate Sciences and the Environment, France, or the Austrian Academy of Sciences, Austria. He received the Extraordinary Doctoral Award for his Ph.D. thesis.

...