Linked Data Maps: Providing a Visual Entry Point for the Exploration of Datasets

Fabio Valsecchi, Matteo Abrate, Clara Bacciu,
Maurizio Tesconi, and Andrea Marchetti

Institute of Informatics and Telematics (IIT),
National Research Council (CNR), Pisa, Italy
{firstname.surname}@iit.cnr.it
http://wafi.iit.cnr.it

Abstract. Linked Data sets are an ever-growing, invaluable source of information and knowledge. However, the wide adoption of this large amount of interlinked structured data is still held back by some non-trivial obstacles. The one we tackle in this article is the difficulty users have in getting started with their work on Linked Data sources. In fact, querying, and in general dealing with such datasets, requires a deep knowledge about their specific classes, instances and properties. We believe that an *entry point* that eases the access to such information would significantly reduce the barriers around this technology and foster its promotion. Linked Data Maps is a method for representing RDF graphs as interactive, map-like visualizations, based on our previous work focused on the visual exploration of DBpedia. The approach is extended to deal with a wider range of Linked Data sets, and tested with a user evaluation study on two distinct RDF graphs.

Keywords: Linked Data, Dataset Exploration, Information Visualization

1 Introduction

The Linked Data project comprises a large, open and valuable collection of interlinked structured data over the World Wide Web. The potential of the numerous data sources within the Linked Data Cloud[1] is undeniable, and the enormous amount of knowledge they contain is valuable for both public and private organisations. However, a wide adoption of Linked Data and Semantic Web technologies is still prevented by some relevant obstacles. One of these impediments is that users are required a high effort in order to get started with the handling of Linked Data sources. Approaching a Linked Data set remains, especially in the first phase, a hard and onerous task even for expert users. In fact, accessing and querying Linked Data sets require a deep knowledge about their classes, instances and predicates. In general, users often need to figure out how a dataset

[1] http://lod-cloud.net

is structured, how many instances it describes, how the ontology is organized, what kind of instances it contains, how they are linked to each other, and so on. Basically, the key question is "How is the dataset like?". In our opinion, there is a lack of effective and easy-to-master instruments for answering these important questions. To the best of our knowledge, the currently available tools are mainly focused on the presentation of aggregated data, or on the visualization of the details related to a single instance or a small group of them, and do not show a complete representation of the dataset.

In order to fill this gap, we think that applications acting as *entry points* to Linked Data sets would ease the access to these large and complex data sources, by guiding users from a high level overview to the most specific details of RDF triples that compose them. We propose Linked Data Maps, an interactive visualization approach based on the work carried out on DBpedia in [25], which is focused on the automatic transformation of an RDF graph into a map designed according to cartographic principles. A map can leverage human visual perception abilities and consolidated map-reading skills to achieve a high level of efficacy in communicating the properties of large and complex structures such as Linked Data sets [23, 1]. Moreover, an interactive zoomable map literally fits the Visual Information-Seeking Mantra proposed by Ben Shneiderman (*"Overview first, zoom and filter, then details-on-demand"*) [22, 8], which recommends to firstly show an overview in order to let users understand which are the general features of a dataset and then provide mechanisms for deeply exploring the dataset details. We adapted the work by Auber et al. on Gosper treemaps [3] for assigning a position and a shape to the abstract and non-geometrical data generally contained in Linked Data sets. Our solution produces an interactive map displaying all the instances of a dataset, organized in regions according to their ontological classification. The map represents a ground layer on top of which an *atlas* can be constructed by integrating multiple kinds of visualizations and functionalities in order to bring out different aspects of a dataset.

In order to prove the effectiveness of the approach we applied it to some well-known Linked Data sets such as DBpedia and LinkedMDB, and we performed a user evaluation on the applications of both datasets.

1.1 Related Work

The large amount of work carried out for the visualization of Linked Data [8, 19] reflects the strong need of the Semantic Web community to provide users with applications that let them easily access and explore Linked Data sources. The most complete survey [8] about the approaches for visualising and exploring Linked Data states that most of the existing tools i) do not provide an overview of the dataset; ii) are designed only for expert users. In order to tackle and solve those problems, different models and frameworks [6, 18, 17] for dynamically creating and customizing data visualizations have been designed.

Other works employ information visualization techniques for statistical analysis. Cubeviz [20] is a visualization platform that allows to configure, display and analyse statistical RDF data through different chart types. Linked Data

Query Wizard [12] is an analysis tool that, given a SPARQL endpoint, allows to search, filter and visualize resources using multiple diagrams. Payola [15] is a web visualization framework that allows to run an analysis on a dataset and then display the results with a set of visualization plugins. Although these tools are very useful for obtaining interesting aggregated data, they neglect to provide a complete representation of the dataset.

Several graph-based visualization tools with different features have been developed. The LODLive RDF browser [7] allows to manually expand the connections of a given resource and explore the resulting node-link diagram by consulting relationships and attributes of the expanded resources. gFacet [11] combines graph-based visualization and faceted filtering techniques for navigating a certain data source. The Linked Open Graph tool [5] provides a collaborative web interface for examining RDF instance networks of different datasets. These applications support users in understanding the connections between instances but they do not scale well with large datasets.

Other applications are focused in solving specific tasks such as Relfinder [10] which reveals if and how two given resources are connected by visually showing all the paths between them within the RDF graph they belong. Other tools provide visualizations of restricted domains such as Spacetime [26] for spatio-temporal data and Linked Geo Data Browser [24] for spatial data.

To summarize, the strong need for Linked Data exploration tools is still an open issue in the Semantic Web community. We believe that the need becomes even stronger when users approach a dataset for the first time, and that a novel approach should be proposed in order to tackle the issue.

2 The Linked Data Maps Approach

The approach presented in this article is aimed to ease the comprehension of the structure and content of Linked Data sources through interactive web applications developed around carefully designed visualizations. The main users for which our applications are intended are those who want to develop tools on or learn a specific Linked Data set even if are not very familiar with Semantic Web technologies. Furthermore, casual users interested in the informative and educational content of a dataset could benefit from it by using the atlas. Finally, Linked Data experts that approach a new dataset for the first time could also gain precious insights from this kind of application, which would quicken the access to the target information or even help in finding anomalies in the data.

Our main objective is to provide those users with an entry point to the dataset. By "entry point", we mean an initial overview of the dataset that can be queried and interacted with according to Shneiderman's mantra, thus providing an easy access to the information space. The initial overview should allow users to perform the following high-level tasks: i) get a general idea about the way the dataset is structured; ii) perceive its size; iii) compare different parts of it in terms of both size and complexity. More specific tasks are also defined, to characterize the user's wish to get detailed information by interacting with the visualization space: i) lookup or locate an instance; ii) consult its properties; iii) browse the list of its connections; iv) inspect to find the location of instances to which it is connected; v) discover which are the classes to which it is more connected; vi) compare its connections with the ones of other instances; vii) locate a class.

Our solution defines a pipeline that transforms an RDF graph into a map as a result of two steps: *data abstraction* and *spatialization*. The map is then visualized in a web application allowing to query and explore it (Figure 1).

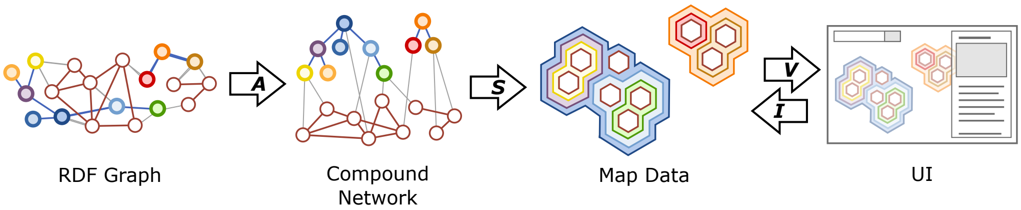

Fig. 1. The Linked Data Maps approach. First the RDF Graph is transformed into a Compound Network (A - Data Abstraction step), then the network is used to compute the Map Data (S - Spatialization step). Finally, these data are visualized (V) in a web application from which the user can interact (I).

2.1 Data Abstraction

As discussed in [25], an RDF graph (including its ontology) may be seen as a *compound network*. Also known as multi-level network, a compound network is a combination of a network and a tree, where the latter is seen as an organizing structure for the former[2]. Hierarchical ontologies are often the base structure of Linked Data sets [8] as in the case of DBpedia, Geonames, Yago and Freebase. However, other datasets such as LinkedMDB, Lexvo and Wordnet RDF are organized as *forests* (i.e., undirected acyclic graphs whose connected components are rooted trees). We deal with such structures by extending our approach to handle common cases of ontologies: i) proper forests composed by n trees; ii) forests comprising only one tree and iii) the degenerate case of a forest in which there are n isolated classes. As shown in Figure 2, the forest is defined by *class nodes* (i.e., ontological classes) while the graph is composed by *instance nodes* (i.e., distinct URIs found as subjects or objects of RDF triples) that represent the leaves of the forest trees. Three kinds of links are considered: *vocabulary links* (VL) are derived from the ontology (i.e., rdfs:subClassOf property), *relationship links* express the connections between instance nodes (e.g., birthPlace between

[2] Recently, many approaches for the scalable drawing of networks have used this kind of abstraction [2, 13]. Our approach uses the ontological classes instead of clusters to play the role of organizing structures.

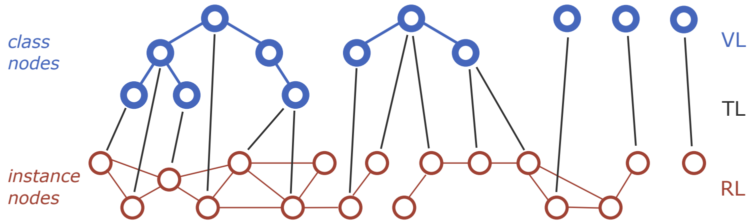

Fig. 2. A graph and an associated forest define the extended compound network. In our case, it is composed by class nodes, instance nodes, vocabulary links (VLs), relationship links (RLs) and type links (TLs).

Galileo and Pisa) and *type links* (TL) define the membership of an instance to a certain class (i.e., rdf:type). Table 1 shows some popular Linked Data sets and the amount of nodes that would compose their compound network.

In general, an instance node could be a member of n classes, but, for the dataset to be compatible with the Linked Data Maps approach, an instance node can only be connected to *compatible* class nodes through a TL. Two class nodes are compatible if they belong to the same tree branch of a connected component of the forest (e.g., a *Scientist* is a *Person*). It is important not to violate this compatibility constraint because otherwise instance nodes would be connected to distinct conflicting class nodes, and consequently would have to be displaced simultaneously in more than one region (e.g., Leonardo Da Vinci is both a *Scientist* and a *Painter*). In order to handle the missing types issue [21], instance nodes without a corresponding class are allowed not to have any TLs.

Dataset	Geonames	DBpedia	LinkedMDB	Wordnet	Lexvo
Class Nodes	7	721	51	5	5
Instance Nodes	8.3M	4.7M	694.400	647.215	128.945

Table 1. The amount of classes and instances of some well-known Linked Data sets compatible with the Linked Data Maps approach.

2.2 Spatialization

As described in [25], the spatialization process adopted for constructing the visualizations is based on the work of Auber et al. on Gosper treemaps [3]. Treemaps [14] efficiently use the available space for compactly displaying complex trees by representing nodes as a hierarchy of regions contained into one another. We choose to use Gosper treemaps because they strongly resemble islands, and therefore support users in intuitively reading the generated geographic-like maps as they would with traditional ones. We generalized the approach in order to visualize forests: in the resulting map, each tree of the forest is depicted as a distinct treemap, and thus a distinct island. A collision detection and a force-directed graph algorithm are jointly adopted in order to obtain a layout in which islands are displaced without overlapping.

Gosper treemaps represent each leaf of a tree as a hexagonal tile having a specific position within the visualization. Instance nodes belonging to the same ontological class are placed together in the same region. By construction, the adopted spatialization process shows the amount of instance nodes linked to a certain ontological class with the area of the corresponding region.

Beyond assuring the compactness of regions, the Gosper treemap layout guarantees a great level of stability, which is useful if an updated version of the map needs to be created, because the new version will not confuse the user with considerable differences in the spatial arrangement. This is an ideal property for Linked Data sets that are often published with yearly updates.

2.3 Visualization and Interaction

To render the map data in a way that allows users to get an overview, zoom and filter the visualization and get details on demand, the approach includes the realization of a web application, mainly constituted by three components:

1. *Map.* It firstly displays the overview of the dataset showing all the instances and classes within it. Zoom and pan mechanisms allow users to filter out certain regions, move within the map and focus their attention on specific regions and instances. Initially, only the regions with a suitable size show the label identifying their ontological class, while the others are automatically loaded during a zoom action. A metaphor for portraying instances as *cities* have been introduced, in order to support users to get orientation within the map and get a feel about the content of a certain region. Clicking an instance on the map triggers the loading of its properties both in the map itself and in the right-side infobox. All the instances connected to the selected one are highlighted on the map as red tiles linked by red lines.
2. *Infobox.* It comprises all the RDF triples in which a selected instance is involved. In particular, classes, data properties, incoming and outgoing relations are organized in a user-friendly way for helping users in the consultation. Furthermore, by clicking on an outgoing or incoming property, the relative content is loaded both on the map and in the infobox, allowing the dataset navigation.
3. *Search Box.* The search functionality allows to perform a text search for a specific instance, show it on the map along with its connections, and get its properties loaded in the infobox.

Multiple aspects of a dataset can be displayed on top of the main visualization as additional thematic maps. Each region may be colored to represent for example: the depth of the corresponding class in the forest, the density (i.e., a normalization on the number of instances) of RDF triples, the density of the object properties, or the density of data properties describing the instances. The layers menu (on the top left) allows to switch between different thematic maps.

3 Evaluation

We applied our approach to two well-known Linked Data sets: DBpedia[4] and LinkedMDB[9], very different in size (Table 1), structure and content. DBpedia is a cross-domain dataset with a hierarchical tree as its main ontology, while LinkedMDB is a film industry dataset with an ontology structured as a collection of disconnected classes. Both visual representations (Figure 3 and 4) reflect the respective dataset structure. The tree of DBpedia, rooted on the class "Thing", is represented by one island composed by several regions, themselves partitioned in many hierarchical sub-divisions. An additional island has been added for containing instances without a class [21]. Instead, the disconnected classes of LinkedMDB produce a completely fragmented representation displayed as an archipelago of plain islands, since a single root is not defined.

3.1 Setup

The previous version of the application [25] had been preliminarily tested with a very limited number of users. For this work, we carried out a more extensive user evaluation with 19 users, collecting both qualitative and quantitative data. Since the intended users are people wanting to approach or to better understand a Linked Data set, all the testers we chose have a computer science background (no lay users were recruited). The 19 participants were 12 males and 7 females and they had different age ranges: 5 were between 20 and 30, 7 between 30 and 40 and 7 above 40.

Half of the people were presented the DBpedia application, while the other half the LinkedMDB one. At the beginning of the session with each user, we asked them to read a short description about the dataset[3]. Then, we gave them some time (5 to 10 minutes) for freely interacting with the application. Finally, they had to fill out a questionnaire. To answer some of the questions they had to perform some tasks using the application. The questionnaire has been structured in four distinct parts for evaluating different aspects of the visualizations and the applications.

1. *Free interaction.* Three questions have been defined for verifying whether the main features (e.g., zoom and pan, search, and click on the map) of the interface had been discovered and used by testers during the initial free interaction phase.

[3] For instance, in the case of DBpedia we used the following description: *DBpedia is a large collection of structured data automatically extracted from Wikipedia. It contains a lot of resources (e.g. "Galileo Galilei", "Pisa", "The Beatles", "Divine Comedy") organized in classes (e.g. "Person", "City", "Band", "Book"). Classes are connected together, forming a hierarchical structure from generic to specific. For example, "FootballPlayer" is a more specific class than "Athlete", which itself is more specific than "Person". Resources are also connected to each other (e.g. "Pisa" is the place of birth of "Galileo Galilei").*

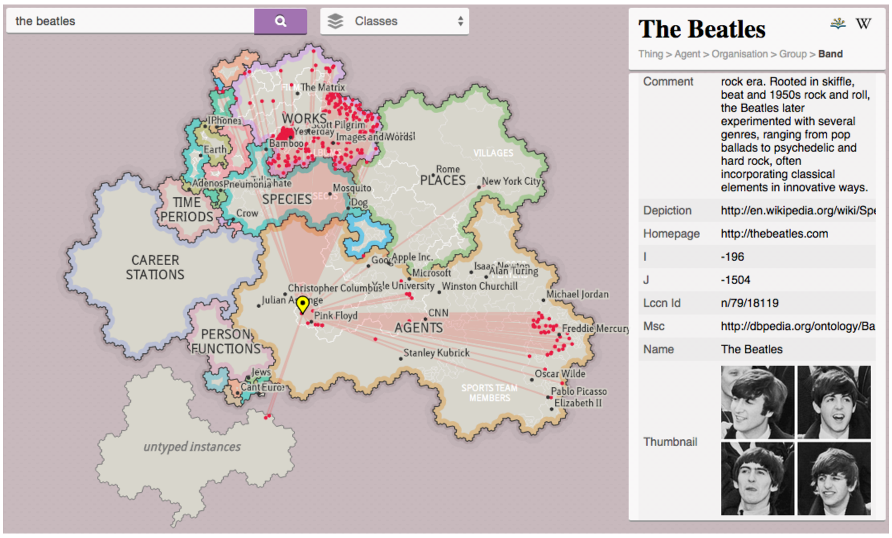

Fig. 3. A screenshot of the DBpedia application, available online at `http://wafi.iit.cnr.it/lod/dbpedia/atlas`. The code is open source and hosted on GitHub (`https://github.com/fabiovalse/dbpedia_atlas`). The map, composed by a main island and a smaller one containing the untyped instances, reflects the hierarchical structure of the ontological tree at the base of DBpedia.

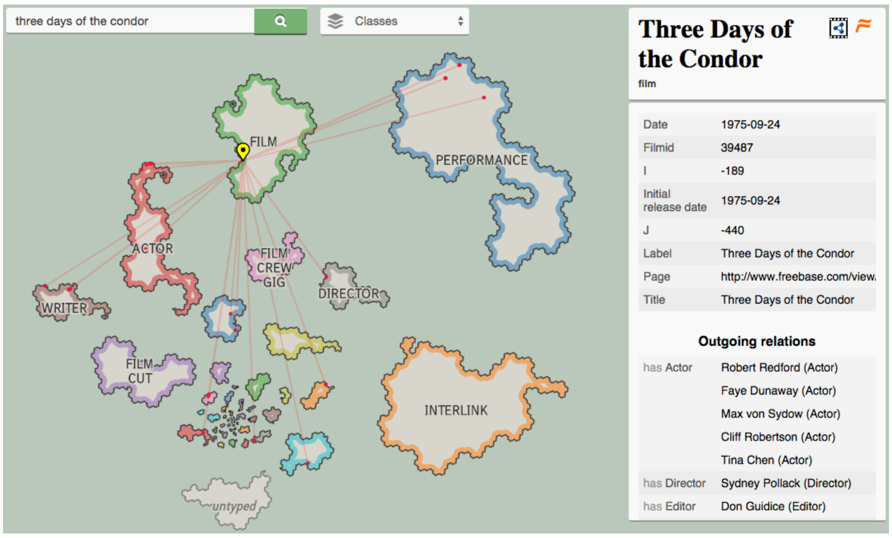

Fig. 4. A screenshot of the LinkedMDB application, available online at `http://wafi.iit.cnr.it/lod/linkedmdb/atlas`. The map, composed by an archipelago of islands, depicts the structure of the disconnected classes forming the ontology of LinkedMDB.

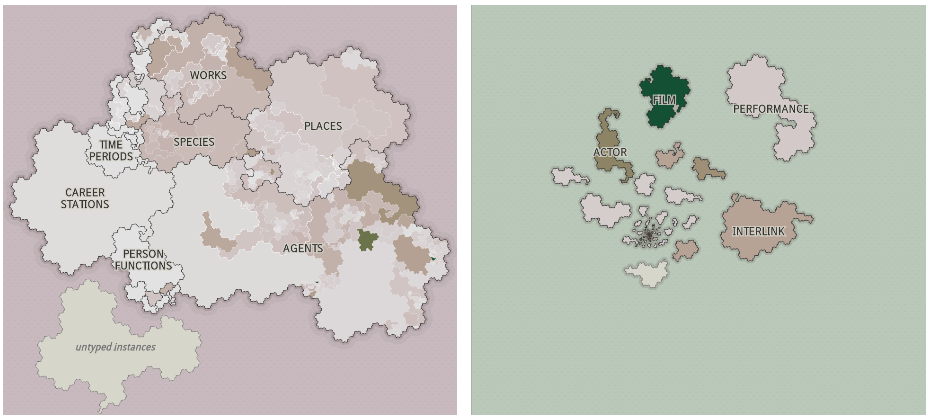

Fig. 5. Thematic maps of DBpedia and LinkedMDB, showing the density of object properties of each class (the darker, the more dense). By inspecting the maps, it can be seen that the most dense classes are Soccer manager, Horse trainer and Jockey for DBpedia and Film for LinkedMDB. The maps can also be compared in size since they share the same scale.

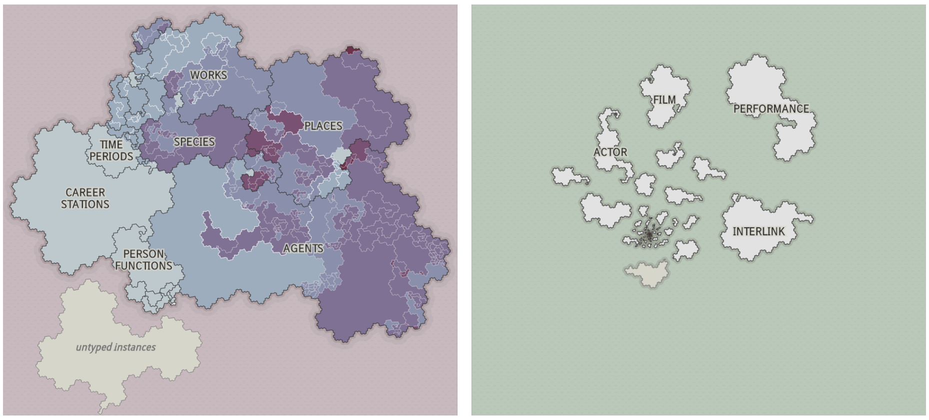

Fig. 6. Thematic maps of DBpedia and LinkedMDB, showing the depth of the classes in their ontology (the darker, the deeper). DBpedia has an island composed by several hierarchical levels, while LinkedMDB has an archipelago of totally flat islands. By inspecting the map of DBpedia, it can be seen that the deepest level of the ontology corresponds to the small Diocese class (top right).

2. *Tasks*. Eleven tasks have been presented to users for testing if the application is useful for: i) visually identifying which are the main regions within a map; ii) looking up and locating resources; iii) recognizing which are the connections between resources, and iv) understanding how resources are categorized.

After they finished each task, users have been asked to state whether it was easy to solve. We adopted a scale of 5 balanced values (i.e., really difficult, difficult, neither easy nor difficult, easy and really easy) ranging from 1 to 5 for scoring the results of each task.

3. *Comprehension.* Three questions have been asked for measuring whether the users had grasped the main aspects of the visualizations (e.g., the meaning of regions and their size).
4. *Final comments.* Four questions have been finally asked for understanding whether the users thought the application was aesthetically pleasing, easy to use, useful and/or self-explanatory. We used a traditional Likert scale [16] ranging from 1 to 5 for scoring the answers of the users about the aforementioned properties.

3.2 Results

The results of the test show that in the case of DBpedia the 80% of the users autonomously discovered they could use the search feature and click on the map, while 70% the zoom and pan behaviour. In the case of LinkedMDB, all the users discovered they could click on the map, the 88% use the search function and the 75% of them used the zoom and pan mechanism. By taking into account those results, now we think that the application requires a clearer way of showing that the zoom and pan feature is available. The plus/minus zoom buttons often included in web interactive maps could be a solution.

The study produced very promising results (Table 2 and 3) and also gave directions for improving the strength of the application in communicating the dataset characteristics. The mean score of every task is above average for both DBpedia and LinkedMDB. One of the main aspects that came out from the study is the need for a closer interaction between the visualization of the map and the infobox. In fact, by observing the users when executing the tasks, we noticed that some are more inclined to navigate the map, while others prefer to just read the infobox.

All the users understood that the size of a region is determined by the amount of resources contained within it, while only a few of them really perceived the complexity of its hierarchical sub-structure. Almost all the users agreed on the aesthetic, ease and usefulness of the application while only a part of them thinks that it is self-explanatory, suggesting that a more intuitive user interface can be envisaged. A user suggested to add an introductory tutorial for explaining the main features the first time the application is accessed.

	Questions	%	Avg
Free Interaction	Have you used the pan & zoom feature?	70%	-
	Have you used the search feature?	80%	-
	Have you clicked on the map?	80%	-
Tasks	Which is the biggest class in the dataset?	90%	3.6
	Which are the main classes? Try to explain your answer.	100%	4.0
	Where is the resource "The Matrix"?	100%	4.2
	Which are the resources connected to "The Matrix"?	100%	4.2
	Where is the resource "The Beatles"?	100%	4.2
	Which are the resources connected to "The Beatles"?	100%	4.3
	Is "Alan Turing" directly connected to "Noam Chomsky"?	90%	3.7
	Is "Divine Comedy" directly connected to "Dante Alighieri"?	90%	3.5
	Is "Earth" directly connected to a resource classified as "Place"?	70%	3.4
	Is "Crow" directly connected to a resource classified as "Food"?	80%	3.9
	Which are the classes of the resource "Danube"?	100%	3.7
Comprehension	Some regions are bigger than others because: 1) they contain more resources; 2) they contain more classes; 3) they are deeper in the hierarchy; 4) they have more connections.	100%	-
	Does region "Agent" seem more complex than "Place"? 1) Yes, "Agent" is more complex; 2) No, "Place" is more complex; 3) They are of similar complexity; 4) I don't know.	20%	-
	Does region "CareerStation" seem more complex than "Place"? 1) Yes, "CareerStation" is more complex; 2) No, "Place" is more complex; 3) They are of similar complexity; 4) I don't know.	90%	-
Final Comments	The map is aesthetically pleasing	-	4.1
	The map is easy to use	-	3.4
	The map is useful	-	3.6
	The interface is self-explanatory	-	3.0

Table 2. The overall result of each question of the survey about DBpedia grouped by category. The percentage of correct answers and the average score reported by users are shown in the last two columns. Scores range on a scale from 1 to 5 (from *really difficult* to *really easy* in the case of the tasks, while from *totally disagree* to *totally agree* in the case of the final comments).

	Questions	%	Avg
Free Interaction	Have you used the pan & zoom feature?	78%	-
	Have you used the search feature?	89%	-
	Have you clicked on the map?	100%	-
Tasks	Which is the biggest class in the dataset?	88%	3.7
	Which are the main classes?	67%	3.8
	Where is the resource “The Matrix”?	100%	4.6
	Which are the resources connected to “The Matrix”?	78%	3.9
	Where is the resource “Sidney Lumet” (the film director)?	89%	4.3
	Which are the resources connected to “Sidney Lumet” (the film director)?	77%	4.4
	Is “Blade Runner” directly connected to “Rutger Hauer”?	100%	3.8
	Is “Marcello Mastroianni” directly connected to “Claudia Cardinale”?	78%	3.2
	Is “Christmas (Film Subject)” directly connected to a resource classified as “Actor”?	100%	3.9
Comprehension	Some regions are bigger than others because: 1) they contain more resources; 2) they contain more classes; 3) they are deeper in the hierarchy 4) they have more connections.	89%	-
	Does region “Film” seem more complex than “Actor”? 1) Yes, “Film” is more complex; 2) No, “Actor” is more complex; 3) They are of similar complexity; 4) I don’t know.	44%	-
	Does region “Performance” seem more complex than “Film”? 1) Yes, “Performance” is more complex; 2) No, “Film” is more complex; 3) They are of similar complexity; 4) I don’t know.	22%	-
Final Comments	The map is aesthetically pleasing	-	3.8
	The map is easy to use	-	3.6
	The map is useful	-	3.9
	The interface is self-explanatory	-	2.9

Table 3. The overall result of each question of the survey about LinkedMDB grouped by category. The percentage of correct answers and the average score reported by users are shown in the last two columns. Scores range on a scale from 1 to 5 (from *really difficult* to *really easy* in the case of the tasks, while from *totally disagree* to *totally agree* in the case of the final comments).

4 Conclusion and Future Works

In this paper, we presented an approach for visualizing and exploring Linked Data sets based on information visualization and cartographic techniques. We extended the work started in [25] to handle datasets organized by forest-based ontologies. We applied the approach to DBpedia and LinkedMDB, implementing two web applications available on-line. We presented the outcome of the user evaluation that shows promising results, but also some weak points. Beside addressing these flaws, next studies will be focused on other improvements. The zoom behaviour could be strengthened by progressively loading the most important "cities" (i.e., instances), selected according to a ranking factor. This score could be based on the number of links to which a node is connected (i.e., in-degree, out-degree or degree). Finally, a similarity measure between instances could be introduced. This aspect could be very useful for further organizing regions in clusters of similar instances. A layout strategy completely based on such a measure could also solve the incompatible classes problem described in Section 2.1, making it possible to handle an even wider range of datasets.

References

1. Abrate, M.: Data Cartography: atlases and maps for non-geographical data. Ph.D. thesis (2014)
2. Archambault, D., Munzner, T., Auber, D.: Grouseflocks: Steerable exploration of graph hierarchy space. Visualization and Computer Graphics, IEEE Transactions on (2008)
3. Auber, D., Huet, C., Lambert, A., Renoust, B., Sallaberry, A., Saulnier, A.: Gospermap: Using a gosper curve for laying out hierarchical data. IEEE Transaction on Visualization and Computer Graphics (2013)
4. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: A nucleus for a web of open data (2007)
5. Bellini, P., Nesi, P., Venturi, A.: Linked open graph: browsing multiple sparql entry points to build your own lod views. Journal of Visual Languages & Computing (2014)
6. Brunetti, J.M., Auer, S., García, R.: The linked data visualization model. In: International Semantic Web Conference (ISWC) (2012)
7. Camarda, D.V., Mazzini, S., Antonuccio, A.: Lodlive, exploring the web of data. In: Proc. of the International Conference on Semantic Systems (2012)
8. Dadzie, A.S., Rowe, M.: Approaches to visualising linked data: A survey. Semantic Web (2011)
9. Hassanzadeh, O., Consens, M.P.: Linked movie data base. In: LDOW (2009)
10. Heim, P., Hellmann, S., Lehmann, J., Lohmann, S., Stegemann, T.: Relfinder: Revealing relationships in rdf knowledge bases. In: Semantic Multimedia (2009)
11. Heim, P., Ziegler, J., Lohmann, S.: gfacet: A browser for the web of data. In: Proc. of the International Workshop on Interacting with Multimedia Content in the Social Semantic Web (2008)
12. Hoefler, P., Granitzer, M., Veas, E., Seifert, C.: Linked data query wizard: A novel interface for accessing sparql endpoints. In: Proceedings of Linked Data on the Web at WWW (2014)
13. Hu, Y.: A gallery of large graphs. `http://yifanhu.net/GALLERY/GRAPHS/` (2014)
14. Johnson, B., Shneiderman, B.: Tree-maps: A space-filling approach to the visualization of hierarchical information structures. In: IEEE Proc. of Conference on Visualization (1991)

15. Klímek, J., Helmich, J., Nečaský, M.: Payola: Collaborative linked data analysis and visualization framework. In: The Semantic Web: ESWC 2013 Satellite Events (2013)
16. Likert, R.: A technique for the measurement of attitudes. Archives of psychology (1932)
17. Luggen, M., Gschwend, A., Anrig, B., Cudré-Mauroux, P.: Uduvudu: a graph-aware and adaptive ui engine for linked data. In: Proceedings of Linked Data On the Web (LDOW) (2015)
18. Lukovnikov, D., Stadler, C., Lehmann, J.: Ld viewer-linked data presentation framework. In: Proceedings of the 10th International Conference on Semantic Systems (2014)
19. Marie, N., Gandon, F.: Survey of linked data based exploration systems. In: Proceedings of Intelligent Exploitation of Semantic Data (IESD) (2015)
20. Martin, M., Abicht, K., Stadler, C., Ngonga Ngomo, A.C., Soru, T., Auer, S.: Cubeviz: Exploration and visualization of statistical linked data. In: Proceedings of the 24th International Conference on World Wide Web Companion (2015)
21. Paulheim, H., Bizer, C.: Type inference on noisy rdf data. In: The Semantic Web–ISWC (2013)
22. Shneiderman, B.: The eyes have it: A task by data type taxonomy for information visualizations. In: IEEE Symposium on Visual Languages (1996)
23. Skupin, A.: From metaphor to method: Cartographic perspectives on information visualization. In: IEEE Symposium on Information Visualization (2000)
24. Stadler, C., Lehmann, J., Höffner, K., Auer, S.: Linkedgeodata: A core for a web of spatial open data. Semantic Web (2012)
25. Valsecchi, F., Abrate, M., Bacciu, C., Tesconi, M., Marchetti, A.: Dbpedia atlas: Mapping the uncharted lands of linked data. In: Proceedings of the 8th Linked Data On the Web Workshop (LDOW) (2015)
26. Valsecchi, F., Ronchetti, M.: Spacetime: a two dimensions search and visualisation engine based on linked data. In: The Eighth International Conference on Advances in Semantic Processing (2014)