

LODmilla: a Linked Data Browser for All

András Micsik, Sándor Turbucz, Attila Györök

Department of Distributed Systems, MTA SZTAKI

Lágymányosi u. 11., Budapest, Hungary

{andras.micsik, sandor.turbucz, attila.gyorok}@dsd.sztaki.hu

Abstract. Although the Linked Data paradigm is extremely popular, and there is immense amount of Linked Open Data available worldwide, the human exploration of these datasets is limited. In our work we try to evolve a generic platform called LODmilla for exploring and editing Linked Open Data. Our aim is to enable the extraction and sharing of data associations (or information) hidden in Linked Open Data. LODmilla is an open web application supporting graph views, graph searching and many other commodity features for surfing over Linked Data.

Keywords: Linked Data, LOD, Semantic Web, graph visualization

1 Introduction

In 2006 Tim Berners-Lee outlined a set of best practices for publishing and connecting structured data on the Web: the Linked Data (LD) principles [1]. This endorses the connection of RDF (Resource Description Framework) datasets with each other forming a global data network. The merge of the LD and Open Data concepts became very popular in last years named as Linked Open Data (LOD) [2].

Although the LOD cloud diagram [3] recorded the immense growth of available LOD datasets, the human exploitation of this data bonanza is still very ad hoc. In our work we try to evolve a generic platform for exploring and editing Linked Open Data. Our aim is to enable the extraction and sharing of data associations (or information) hidden in Linked Open Data.

Linked Data is built using triples, where each triple defines a statement in the form of subject-predicate-object. The graph representation of such data is quite straightforward and widely used. The subject and the object are nodes in the graph, and edges between them are labelled with the predicate name. This way we get a directed, labelled graph as a view of the Linked Data. Another natural way to present LOD content is using a tabular format.

Like in a spreadsheet, the three parts of a triple can be sorted or grouped in separate columns. The Graphity [4] and Tabulator [5] are examples of the tabular browsing with nested tables, and one could list several prototypes of graph-based LOD browsers (LodLive [6], RelFinder [8], oobian [7], etc.). Here we present LODmilla [9], which is a continuously improving service for generic Linked Data browsing trying to combine the best features from both tabular and graph-based browsers.

2 The LODmilla browser

The aim of LODmilla is to facilitate the human inspection of information accessible as Linked Data. LODmilla users may find associations between objects, and record various “mind map” views of the underlying data. For this, we provide both graph and table based browsing, and exploration functions specific to RDF.

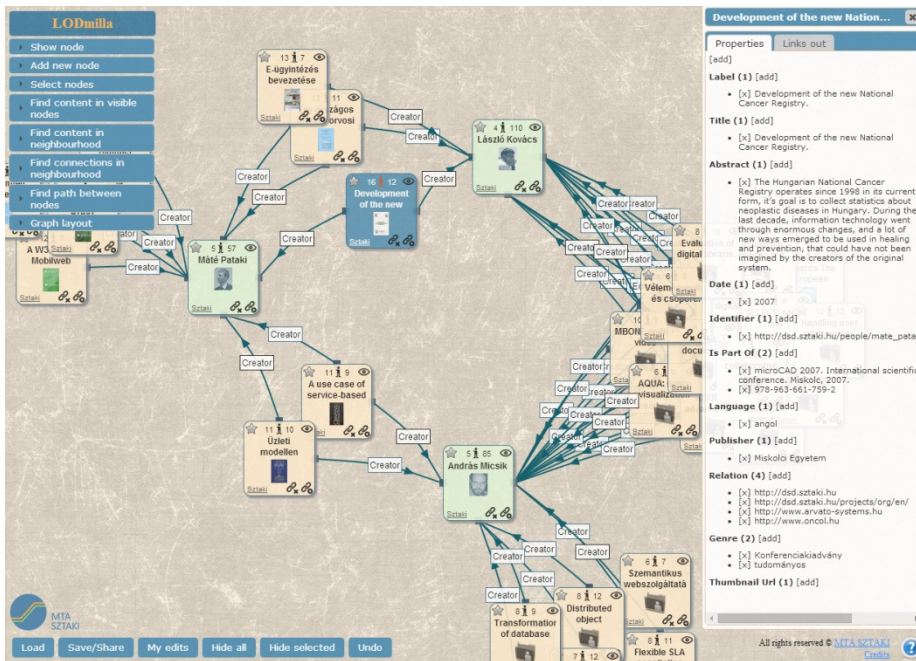


Fig. 1. A snapshot of LODmilla in action

LODmilla (Fig. 1) is running in conventional web browsers as a web app. While it is primarily visual, it also contains textual representations of resource properties in order to combine the best of both worlds. Its goal is to provide a single application for the interactive exploration of LOD content residing in multiple knowledge bases.

The browser provides the following function groups:

- Opening URIs as nodes, expanding and browsing by RDF properties,
- Zooming and panning in the graph view,
- Reorganization of the graph view,
- Various search operations in the graph,
- Saving and loading graph views as well as sharing them with other users,
- Editing Linked Data,
- Undo of previous actions.

The specific search operations allow users to find string occurrences hidden in triples both in the current view and in the neighborhood of selected nodes. This way one

can expand the graph view in the desired direction, for example by opening all nodes representing creators, or by searching for the word 'semantic' near one author. Furthermore, a path search is also offered, revealing connections between selected entities (nodes).

In order to facilitate caching and fast triple loading, the search operations use a dedicated backend, which is also responsible for saving and sharing graph views. LODmilla can switch between two methods for fetching triple data; the first one is based on SPARQL querying, the second one uses actionable URIs. By using the Jena toolkit at the backend, we can parse incoming RDF as Turtle, RDF/XML, JSON, etc. Therefore, a large variety of datasets can be used at the same time, even without configuring the dataset details in the frontend (in this case actionable URIs are used to load graph details). Future plans include the use of VoID [10] for automatic configuration of dataset-specific features in the frontend.

As we cannot rely on SPARQL querying for path extraction, we had to apply graph traversal methods, which have the advantage that they may work across datasets as well. Path finding currently works between two nodes, it starts from both nodes using simple heuristics to select the next path segment to explore. We plan to improve this algorithm both by improving the heuristics and by parallelizing it on several virtual machines. The connection and content search operations use breadth-first traversal of triples (naturally excluding too common connections such as *rdf:type* and nodes having too many connections). The content search is typically useful for finding text occurrences hidden in the multitude of properties, while the connection search helps users to see selected aspects (connection types) of the graph. In both cases, there is an important problem due to the unidirectionality of node connections: we can search only by outgoing connections (where the current node entity is the subject in the triple) and not by incoming connections, therefore some of the information sought may remain undiscovered.

During the processing of triples, we use some assumptions to improve the visual presentation, for example to show small icons for nodes, we use the *foaf:depiction* and *dbpedia:thumbnail* property values, or if not present, the *rdf:type* property values are mapped to a set of predefined icons (e.g. person, paper, organizational unit, etc.). Similarly, the texts shown in nodes are taken from *rdfs:label*, *dc:title*, *foaf:name* or *skos:prefLabel* properties. Properties are also scanned for images, geolocations and external URLs. Images are shown inline in the info panel, and locations are shown on a map.

Recent improvements of the browser include the editing and reorganization functions. Changing a graph is more natural by drawing than by modifying the triples, therefore we added the possibility to insert new nodes, draw new edges and also to remove edges and nodes in the graph. Such light-weight editing can be used to quickly fix errors or complete missing parts in the graph. These changes are translated into SPARQL Update statements for further use by the author. In this sense one can think LODmilla as a Linked Data Editor for non-professionals.

When the graph view gets cluttered the user can ask for rearrangement of nodes using several methods. We experiment with the adaptation of Spring, Grid and Radial layout algorithms to LOD graphs (which typically contain many cycles) and their

parametrizations to provide useful presets for various usage scenarios [10]. For example, it is possible to lay out strongly connected nodes closer to each other, or to group nodes by their types. In general, the insertion of new nodes is done in the least disruptive way, without moving existing nodes significantly, yet positioning new nodes closely in the free areas of the canvas. The layout algorithms also include genetic modifications of the layout between iterations based on graph details such as node distances or node types. In the future we would like to develop metrics for the ‘goodness’ of the layout and to guess the number of iterations necessary for a suitable layout.

We think that LODmilla unifies most features found in previously implemented LOD browsers and it also exhibits novel principles such as serving multiple LOD datasets at a time and presenting connections between nodes in separate datasets. Beyond the new graph search operations, our development of the browser continues to include and improve useful features for Linked Data exploration. LODmilla can be used as a public service¹ and its source code is available on GitHub².

References

1. Berners-Lee, T.: Linked Data - Design Issues. 2006, <http://www.w3.org/DesignIssues/LinkedData.html>
2. Bizer, C.: The Emerging Web of Linked Data. *IEEE Intelligent Systems*, Vol.24, no.5, pp.87-92, Sept.-Oct. 2009, doi: 10.1109/MIS.2009.102
3. Cyganiak, R., Jentzsch, A.: The Linking Open Data cloud diagram. <http://lod-cloud.net/>
4. Graphity, <http://graphity.org>
5. Berners-Lee, T., Chen, Y., Chilton, L., Connolly, D., Dhanaraj, R., Hollenbach, J., Lerer, A., Sheets, D.: Tabulator: Exploring and analyzing linked data on the semantic web. In *Proceedings of the 3rd International Semantic Web User Interaction Workshop (SWUI06)* (2006).
6. LodLive, <http://en.lodlive.it/>
7. ::oobian::, <http://oobian.com/>
8. Lohmann, S., Heim, P., Stegemann, T., Ziegler, J.: The RelFinder user interface: interactive exploration of relationships between objects of interest. In *Proceedings of the 15th international conference on Intelligent user interfaces (IUI '10)*. ACM, New York, NY, USA, 421-422. DOI=10.1145/1719970.1720052
9. Micsik, A., Tóth, Z., Turbucz, S.: LODmilla: Shared Visualization of Linked Open Data. In: L. Bolikowski, V. Casarosa, P. Goodale, N. Houssos, P. Manghi, J. Schirrwagen (eds.) *Theory and Practice of Digital Libraries - TPDL 2013 Selected Workshops*, Springer 2014 CCIS, DOI: 10.1007/978-3-319-08425-1_9
10. Alexander, K., Cyganiak, R., Hausenblas, M., and Zhao, J.: Describing Linked Datasets - On the Design and Usage of VoID, the 'Vocabulary of Interlinked Datasets'. In *WWW 2009 Workshop: Linked Data on the Web (LDOW2009)* (Madrid, Spain, 2009).
11. Golbeck, J., Mutton, P.: Spring-Embedded Graphs for Semantic Visualization. In: V. Geroimenko, Ch. Chen (eds.) *Visualizing the Semantic Web*, Springer 2006, pp 172-182, DOI: 10.1007/1-84628-290-X_10

¹ <http://munkapad.sztaki.hu/lodmilla/>

² <https://github.com/dsd-sztaki-hu/LODmilla-frontend>