

Relations in a Click, browsing LODs

What are the problems addressed by Linked Open Graph and Social Graph

Linked Open Graph is a unified solution for problems and cases such as:

- 1) Linked Open Data, are complex structures to be understood if you just access to them for their exploitation and reuse.
 - A tool for browsing LOD selecting relationships among URI elements and their attributes can be a solution for developer for data and knowledge engineers. Therefore, services that allow you to insert URI of LOD to navigate on their structure are very important.
- 2) portals and services need to cope with relationships among Users, Contents and external sources. Thus the users need to have a clear view about the relations among these entities to better understand context at a glance, such as:
 - Who has provided a given content: group, user, etc.; How a given content has been used by other users: promoted, played, uploaded, updated, etc.; Which are the geographical locations mentioned by a content, and where is located, may be connected with external geoname, and GPS coordinates; Which are the person names cited into a content: vip names related to dbPedia, users of the portals, other person names cited in many other content; Who has been contact with how, what they have done, etc. The users need to discover and navigate among these relations by selecting those of interest for each specific case.

For these cases the Linked Open Graph provides algorithms and tools to solve the problem dynamically showing the relationships and resolving connections. The Users can play with graph elements moving and zooming, turning on/off connections, filtering and expanding, exploring and playing to see more information, exploiting connection with LOD, etc. Social Graph is a simplified Linked Open Graph for social networks

How it has been solved

The Linked Open Graph allows to display and browse the structure and relations among the RDF entities. The graph is made of two kinds of nodes, rectangular shaped nodes represent entities (content, terms, users, etc.) while circular shaped nodes represent relations. Directed edges connect an entity node to a relation node and a relation node to an entity node. Examples of relations are shown in Figure.

Regarding the user interactions the User is able to: **Expand** an entity node with its relations adding them to the graph; **Focus** on an entity, in this case the graph is cleared and only the focused node is shown with its relations; **Open**, that is the play of the page or content associated with the node; use the Back button to go back to previous states of the graph (e.g. after a focus); Zoom/Pan the view; Hide/show types of relations to reduce the complexity of the graph. A special node is the 'More' node that is presented when in a relation are present many nodes (e.g., the content associated with a group). In this case, providing all nodes could be infeasible thus a limited number of nodes is provided and a 'more' node is added to the relation. Clicking on it other nodes are added to the relation in a way similar to classical pagination used to present long lists in HTML.



