

DISI - Via Sommarive 14 - 38123 Povo - Trento (Italy)  
<http://www.disi.unitn.it>

# ENTITY-CENTRIC VISUALIZATION OF OPEN DATA

Sajan Raj Ojha, Mladjan Jovanovic, Fausto Giunchiglia

January 2015

Technical Report # DISI-16-003  
Version 1.0

**Published also:** Human-Computer Interaction – INTERACT 2015,  
September

# Entity-Centric Visualization of Open Data

Sajan Raj Ojha<sup>1</sup>, Mladjan Jovanovic<sup>1</sup>, Fausto Giunchiglia<sup>1</sup>

<sup>1</sup> Department of Information Engineering and Computer Science, University of Trento,  
Via Sommarive 9, 38123 Trento, Italy

{sajanraj.ojha, Mladjan.Jovanovic, Fausto.Giunchiglia}@unitn.it

**Abstract.** The diversity, which is intrinsic to open data, affects intuitiveness and acceptance from the end users perspective. While the existing visualization techniques provide various ways to exploit open data in different domains, they still lack mechanisms to capture generality and flexibility across highly heterogeneous data representations that are crucial in the open data domain. We propose a novel visualization approach that exploits an entity-centric representation of open data. The key idea is that people intuitively perceive things as entities and categorize them according to their similarities and differences. The approach is supported by a UI architecture that allows aligning underlying open data described by the entities with a meaningful user experience. A preliminary user study shows the feasibility of our approach and its potential in turning the exploitation of open data catalogues into a positive user experience.

**Keywords:** HCI, Open Data, Linked Open Data (LOD), Usability, User Experience (UX), Visualization.

## 1 Introduction

Open data<sup>1</sup> are data that can be freely used, reused and redistributed by anyone - subject only, at most, to the requirement to attribute and share alike. A huge volume of information is made available through data portals<sup>2</sup> that supports citizen to access data. In its original form, open data are not known in advance and varied in format and content. Moreover, different datasets have different provenances and use different modeling approaches which make it even harder for consumers to consume the content of data catalogue.

The Semantic web community is facing similar challenge in linked data consumption [35]. Linked open data has already made a big step by adding both diversity and machine-readable semantics of data on the Web. However, the scale of the Web provides unlimited amounts of cross-domain data whose contexts impose various perspectives and interpretations from a human side. This requires a general method to handle open data, which is domain-independent and user-centered at the same time.

---

<sup>1</sup> <https://okfn.org/>

<sup>2</sup> <http://datahub.io/dataset>: As of November 2014 there are 8733 datasets registered

Research done in [41] shows that people want more interactivity than simply downloading and manipulating files. The major research challenge currently faced by the linked open data community is to figure out how to present the structured data in an intuitive and generic way for common users [8].

Various visualization applications are being developed in Open Linked Data community. Most of these applications are domain-specific and serve specific purposes [11]. They work with specific scenarios and require fixed, predefined data. Moreover, these applications use different visualization and navigation techniques. They usually require certain level of technical knowledge and thus add an extra effort for the common users [12, 21, 37].

We propose a novel approach to align open data with user experience. In this approach people intuitively perceive things as entities and categorize them according to their similarities and differences. We describe open data as a network of interrelated entities. On top of this entity-centric description of open data, we build visualization layer.

An entity represents any unique object that can be described by a set of attributes and relations to other objects. Each entity has a reference class that actually determines its type. Entity type is defined in terms of attributes (such as name, age), relations (such as located-in, friend-of), services (such as computeAge or computeInverseRelation) and categories of Meta attributes (such as timespan, validity and provenance). Entity types are organized into hierarchy of domains, with an ordering on attributes, relations and services. There are relatively few commonsense entity types (such as person, event, location) and many domain dependent entity types.

The paper is organized as follows: Section 2 gives a brief overview of visualization tools found in Open Data and Linked Open Data communities. Then it describes UX in general. In Section 3, we show a motivating problem. Section 4 gives a solution methodology and proof-of-concept prototype. Section 5 describes the user study we have conducted on the prototype. Section 6 concludes the paper.

## 2 State of the Art

Open Data are being used for different purposes and in different domains, including business, military intelligence, research and innovation, tourism and others. Since our focus is in effective visualization, navigation and exploration of open data, we describe existing solutions relevant to our problem domain. Then we describe general UX dimensions that we aim to exploit in our solution.

### 2.1 Open and Linked Data Visualization Tools

In this section, we give comparative analysis of the open data visualization systems found in literature. Comparison is done against a set of criteria adapted from [11]. The criteria are:

1. **Interactive Visualization (IV)**: Refers to the use of interactive representation through different kind of widgets (such as images, buttons and maps). Here, human

perception is considered in understanding the complexity of the data structure and discovery and analysis of the data [3, 8, 10, 25].

2. **Relations (R)**: Denotes different kinds of relationships within one or multiple datasets to understand the data and discover new data [4, 7, 16, 21].
3. **Details on demand (DOD)**: Deals with exposing different level of details for the data as needed [10, 25].
4. **Scalability (S)**: Denotes the ability to manage and link large amount of heterogeneous data, which are loosely coupled [4, 10, 22].
5. **Filtering (F)**: Refers to an ability to suppress irrelevant information and focus only on information relevant for particular context or user session [4, 10].
6. **History (H)**: Describes the ability to record history of interaction allowing the user to review or retrace paths, undo/redo their actions [8, 27].
7. **Faceted exploration and Navigation (FEAN)**: Refers to flexible mechanism that enables setting particular context for search and exploration, and switching to another context based on relations the user explores during the session [9, 39, 15].
8. **Domain Independency (DI)**: Means that applications are not coupled with the specific domain and can exploit wide range of underlying datasets [20, 6, 8, 25].
9. **Target User Group (TUG)**: Means we differentiate three target user groups [11, 35].
  - (a) Common User (CU): End user who does not have any background in ontologies.
  - (b) Tech User (TU): End user who has understanding of the underlying technology and ontologies.
  - (c) Domain Expert (DU): End user with expertise in data of a particular domain. DU might or might not have knowledge of an underlying technology.

Table 1 provides a brief comparison of existing Open Data applications. When visualizing open linked data to make them accessible for common end users who are not familiar with Semantic Web languages, it is important not to present data as URIs or triples but in a more user-friendly way. Although some systems provide high-level interactivity and emphasize different kinds of relations in visualized datasets, they mostly use RDF to describe the data. From the usability viewpoint, it adds an extra effort to the end user to interpret and understand what is being visualized. Applications like Dipper<sup>3</sup>, Disco<sup>4</sup>, Marbels<sup>5</sup>, Piggy Bank<sup>6</sup>, Sig.ma, URI Burner<sup>7</sup>, Zitgist<sup>8</sup> and IsaViz<sup>9</sup> employ complex notations for visualization. They mainly use knowledge graphs that quickly become cumbersome as users drill down the data. Tools like

---

<sup>3</sup> <http://api.talis.com/stores/iand-dev1/items/dipper.html>

<sup>4</sup> <http://www4.wiwiss.fu-berlin.de/bizer/ng4j/disco>

<sup>5</sup> <http://www5.wiwiss.fu-berlin.de/marbles>

<sup>6</sup> <http://simile.mit.edu>

<sup>7</sup> <http://linkeddata.uriburner.com>

<sup>8</sup> <http://dataviewer.zitgist.com>

<sup>9</sup> <http://www.w3.org/2001/11/IsaViz>

OpenLink<sup>10</sup>, RDF Gravity<sup>11</sup>, RelFinder<sup>12</sup>, SIMILE/Exhibit<sup>13</sup> and LESS<sup>14</sup> provide good visualization support, but are highly contextualized and does not support better filtering and tracking history of interaction. In terms of cross-domain support, most of the systems allow to visualize in a predefined context, without the ability to switch to another context based on the relations user perceives during the session. This limitation also comes from the automated production of Linked Data which raises the problem of the accuracy and completeness of the datasets. In particular, incorrect or missing values, or incorrect links makes it hard to correlate data [11]. All the factors above require a certain level of expertise, either in ontologies or domain of the visualized data or even knowledge of specific UI notation. This is the reason why majority of systems is well suited for more experienced users. Thus, LOD community is still struggling to come up with common visualization tool that captures open data diversity and unexpectedness in a user-friendly way.

**Table 1.** Comparing Functionality of different tools

Application	IV	R	DOD	S	F	H	FEAN	DI	TUG
Dipper	L	L	L	L	L	L	L	L	TU
Disco	L	L	M	L	L	L	L	H	TU
Marbels	L	M	M	L	L	L	L	H	TU
Piggy Bank	L	M	H	L	L	H	H	H	CU
Sig.ma [43]	L	L	M	H	H	M	L	H	TU
URI Burner	L	L	M	L	L	L	L	H	TU
Zitgist	L	L	H	M	H	M	H	H	TU
DBPediaMobile [4]	H	L	H	M	H	H	L	M	CU
Fenfire [25]	H	H	H	H	M	L	L	H	CU-M; TU
IsaViz	H	H	H	H	H	H	H	H	CU; TU
LESS	H	L	M	L	L	L	L	H	CU-M; TU
OpenLink	H	H	H	M	H	H	L	H	CU-M; TU
RDF Gravity	H	H	H	M	H	L	L	H	CU-M; TU
RELFinder	H	H	H	H	H	L	L	L	CU-M; TU
Tabulator [7]	H	M	H	L	L	M	L	H	TU
SIMILE/Exhibit	M	L	L	L	H	L	L	L	CU
Rhizomer [17]	L	M	H	L	M	L	H	L	TU
Sgvizler [40]	H	M	L	L	L	L	L	L	TU
LODWheel [42]	L	H	M	M	M	M	L	L	TU
Calluna [36]	H	H	H	L	H	H	L	M	CU

Low (L) - The system has no or low support for the category.

Medium (M) - The system supports the feature to some extent.

High (H) - The system has full support for the category.

<sup>10</sup> <http://lod.openlinksw.com/ode>

<sup>11</sup> <http://semweb.salzburgresearch.at/apps/rdf-gravity>

<sup>12</sup> <http://relfinder.dbpedia.org>

<sup>13</sup> <http://www.simile-widgets.org/exhibit/>

<sup>14</sup> <http://less.aksw.org>

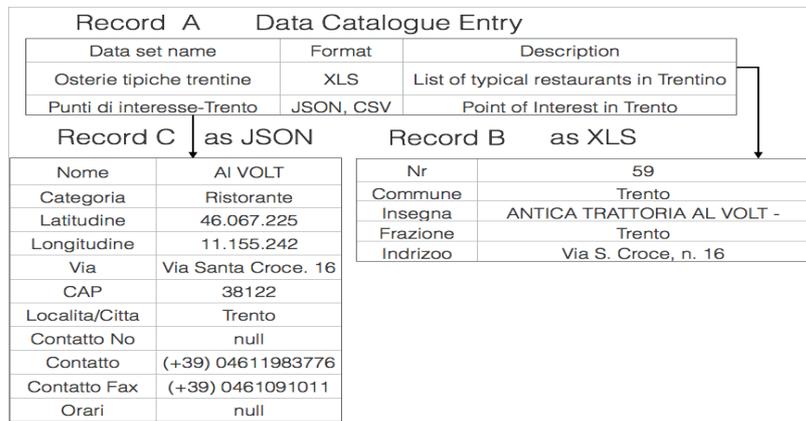
## 2.2 Common UX Dimensions

An overall positive UX is the key to the wide acceptance and increased usage of the open data. In scoping UX for open data, we use existing body of knowledge that expresses UX through a set of the measurable attributes [44]. Here, we enumerate these properties as follows:

1. **Visual and Aesthetic Experience:** Shows how aesthetics affects user perception of the system. It deals with the pleasure that users gain from immediate perception of the system [29, 32, 23, 2].
2. **Emotion:** Describes affective side of UX in terms of feelings and emotions elicited as an outcome of the interaction with the system [29, 2, 14, 33].
3. **Identification:** Addresses the human need to express one self through the interaction with the system [34, 30].
4. **Stimulation:** Relates to the quality of the system to encourage user to use it [24, 38, 31].
5. **Meaning and Value:** Denotes the quality of the system to reflect or represent values that are important to the user [29, 30].
6. **Social relatedness/coexperience:** Describes the pleasure that comes from the social interactions [30, 18].

## 3 A Motivating Example

Open data is widely used in various kinds of services aimed at different categories of users (citizens, domain experts, technical users) [11]. Figure 1 illustrates the inherent



**Fig. 1.** An excerpt from open data catalogue that illustrate the feature of open data critical for end-users acceptance - variety in format and content. It shows data catalogue entry that points to different data structures representing the same data. Record A shows the data catalogue entry for restaurants. Record B points to the tabular representation. Record C refers the JSON representation. Both records B and C contain information about the same restaurant AI Volt.

property of open data - variety in format and content. This feature of open data is critical from the usability viewpoint. With respect to the example, we enumerate major elements of this variability:

1. **Variability in data formats:** Datasets referred by data catalogue are stored in various formats such as XML, XLS, CSV JSON and so on. It is assumed that end user knows how to deal with multiple formats and use them [13]. This extra effort creates a barrier in utilizing the datasets by most of the users. In figure 1, record B<sup>15</sup> is stored in XLS format, whereas record C<sup>16</sup> is written as JSON CSV.
2. **Tabular presentation:** On one hand, it provides a concise overview of information to the user as data are aggregated and list view flattens the depth of the data itself [35, 1]. On the other hand, different types of data require proper visualizations to be perceived and to extract needed information. For example, numbers can be expressed as chart and/or graphs, whereas map-like presentations are more convenient for spatial data [28]. Coming up with a uniform generic visualization is rather difficult, as data from different domains require different ways to visualize information. In the example above, it would be more convenient to display location-based data on a map.
3. **Inefficient search and exploration of datasets:** In general, open data lacks a proper interface that allows for an efficient interaction and exploration [1]. Finding a specific tuple in the dataset requires an extra effort for a user. The data may overlap, be incorrect or incomplete (as shown in figure 1). An interface that allows for multi-faceted interaction and supports exploratory search is still missing [26].
4. **Complex data correlation process:** The process of discovering related data tuples within the datasets is rather difficult. Looking at the record B from figure 1, we can notice that extracting different points of interest located nearby or within the address via Santa Croce is almost impossible from the dataset.
5. **Multiple languages:** Datasets are usually localized. For example, all datasets in Figure 1 are in Italian. This requires a potential user to have knowledge of the language.
6. **Scope of specific domain:** Existing open data visualizations are tailored to specific domains and aimed to be used in different contexts. They often use different ways to interact with the data. This requires users to use different tools in different manner and increases cognitive effort when users switch among the tools.

## 4 Entity-centric Open Data Visualization

In this chapter, we propose a generic user interface framework that aims to bridge the gap between what is called data-driven and user-centered approach to describe open data. In general, we describe a method that captures diversity and unexpectedness of open data in a generic way.

---

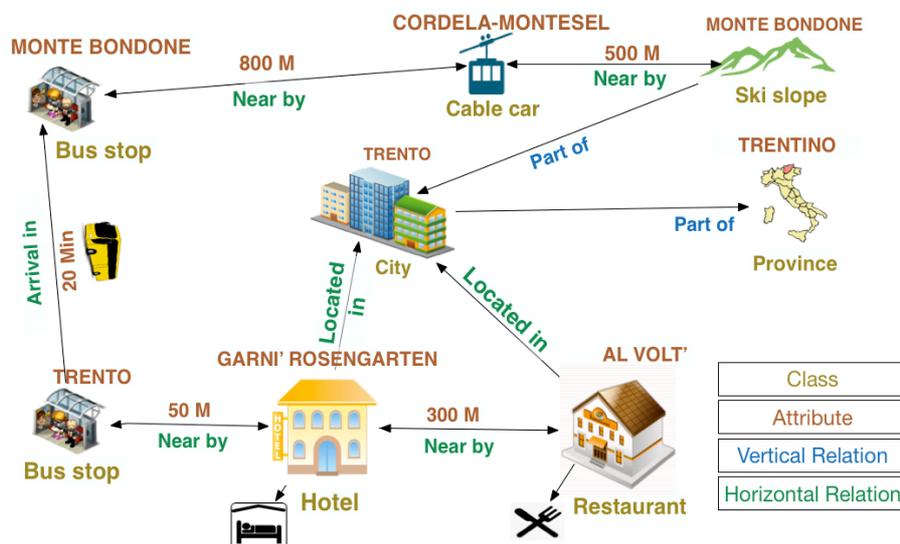
<sup>15</sup> <http://goo.gl/iH0xfD>

<sup>16</sup> <http://goo.gl/77UbzF>

We see two basic steps to this process. First we describe and encode open data in a domain-independent manner. We do that by identifying fine-grained elements that can be modeled individually and used to compose and encode open data for different domains. On top of this model, we build UI layer that brings the open data to the level of a user experience. In this section, we describe basic principles and concrete elements that make our solution.

#### 4.1 Entity-Centric Open Data

In an entity-centric view on open data, instances (objects) are described as entities [19]. Accordingly, tabular view on open data (Figure 1) is transformed into the knowledge graph showing relevant entities and their attributes and relations (Figure 2).



**Fig. 2.** Simplified entity-centric knowledge graph that describes open data as linked entities. Each entity is described with the class and with the name attribute. It also gives different types of relations. Vertical relations are defined on the level of classes, whereas horizontal are generated at runtime.

**Entity:** Denotes representation of real-world object that can be found in different contexts in our everyday life. Each entity is described with a set of attributes and relations with other entities.

**Attribute:** Represents a property of an entity. Each attribute has a name and one or more values. Except the value of the property it represents, each attribute has associated meta attributes such as provenance (describes the origin), permanence,

dependency (whether it is computed) or validity time periods (case of an attribute whose value has changed over time). In addition, we distinguish different kinds of attributes like qualitative (numeric), quantitative (expressed as an adjective) or descriptive (given in natural language).

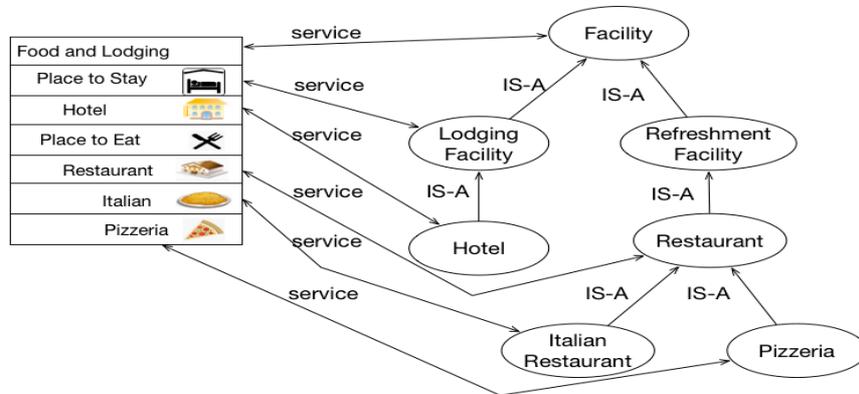
**Relation:** Defines links between entities. In general, we differentiate between two types of relations - vertical (ontological) and horizontal (epistemic). Vertical relations are used to compose hierarchies and mainly for the purpose of classification. These relations are created at design time, on the level of classes. The examples are *is-a* and *part-of*. Horizontal (epistemic) relations link entities according to their specific properties. In this case, the entities can be instances of different, unrelated eTypes. They are defined at the level of instances and are computed at runtime from attribute values. The examples include *located-in* and *near-by*.

**eType:** Presents a reference class of an entity. It is a template that defines the constraints for creating attributes and relations of an entity. Some common examples of eTypes include Location (geospatial entities), Facility (physical entities providing services to the people) or MindProduct (the result of a human intellectual effort).

Clear separation of concepts of attributes, relations and entity classes leads to flexible mechanism that combines these elements to represent domain knowledge.

**Lightweight ontology:** Refers to a method to dynamically compose entity classes, attributes and relations to create the background knowledge about the domain of interest (e.g., tourism, transportation). Thus, we have hierarchies of entity classes (eTypes), attributes and relations. These hierarchies are also known as facet ontologies. Facet is a term widely used in knowledge representation community and denotes aspect of meaning. Each facet contains group of homogeneous terms, where each term in the hierarchy denotes a primitive atomic concept (whether it is a class, a relation or an attribute). Then the lightweight ontology is dynamically created by linking the elements belonging to respective facets into unique hierarchy. The mechanism basically enables construction of multi-perspective representation schemes combining the terms from the facets. This concretely means we can view an entity from different perspectives. While vertical relations provide precompiled classification and description of entity types, horizontal relations are created at runtime. For example, a restaurant can be described as a refreshment facility, as part of the city, as near-by hotel, as on the lake, as where the event is being held, depending on the context provided. Context can be defined by a user request. In this sense, they provide flexible and scalable mechanism to represent domain-specific knowledge that combines different views on the same concept. From the technical side, it brings performance benefits since it may be generated at the beginning of a user session and thereby provide flexible, but also efficient navigation and search.

Figure 3 provides a simplified example of using lightweight ontology to generate UI navigation control. The ontology classifies entity types from a tourism domain according to their purpose. The level of a menu item matches to the depth of a node that represents the entity type.



**Fig. 3.** An example that illustrates the usage of a lightweight ontology to generate a menu for navigating domain entities.

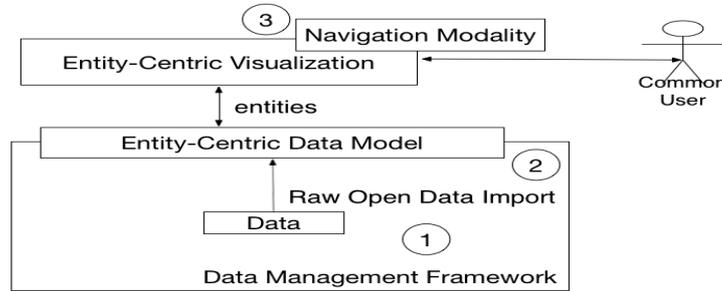
## 4.2 Navigation Modalities

Based on the type of relations among entities, we define two basic kinds of navigation modalities. Combined together, they allow flexible navigation among open data entities. They are as follows:

10. **Vertical (Ontological) Navigation:** Follows vertical relations that exist in hierarchical structures of entity classes. If we look at the Figure 2, we can see *part-of* relations between location-based entities (province, mountain, city). On the other hand, Figure 3 shows lightweight ontology with *is-a* relations between eTypes, starting from Facility as the highest level propagating to more specific types of Italian Restaurant and Pizzeria.
11. **Horizontal (Epistemic) Navigation:** Follows the relations that exist between entities. In the Figure 2, it might not be easy to find Restaurants near Hotel using vertical relation. Horizontal relations connect entities that can belong to different eTypes. They are generated at runtime (for example on a user request). If we look at Figure 2, Hotel and Restaurant are connected with *near-by* relation. In addition, the network of horizontal relations gives the possibility to reach different means of transportation.

## 4.3 Reference Architecture

We describe our solution with the general, high-level architecture (Figure 4). The architecture consists of the three main components:



**Fig. 4.** System architecture

1. **Domain Component:** Describes source open data from the domain of interest. These data are taken from the open data catalogue.
2. **Semantic Component:** Handles entity-centric representation of open data. This component contains mechanisms that transform source open data into entities. The process and the platform behind are described elsewhere [5]. Converted open data are stored in an entity base.
3. **Interaction Component:** Provides UI layer on top of the entity base.

#### 4.4 Experimental Prototype

We have developed experimental prototype<sup>17</sup> that uses the entity base containing open data from the province of Trento<sup>18</sup>. The Entity Base is called Entitypedia. It is a large-scale knowledge base. Currently, it provides 98 percent of correctness with the goal and tendency to improve it. Aside from serving as an entity repository, it provides entity-based services ranging from simpler (such as standard CRUD operations) to more advanced (such as context-dependent operations of search, matching or navigation) The contents of the Entity Base is described in Table 2.

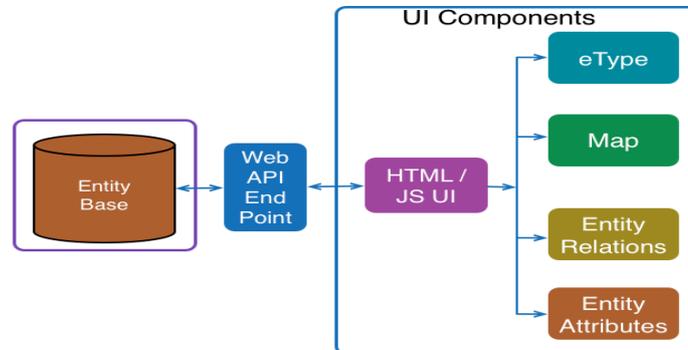
**Table 2.** Entities classified according to eType inside an Entity Base

eType	Total
Location	20704
Facility	141
Lodging Facility	442
Shopping Facility	219
Refreshment Facility	229
Transportation Facility	3379

Figure 5 shows main components of the UI. The UI calls the Web API endpoint which is responsible for querying the data from the Entity Base. The UI

<sup>17</sup> <http://www.opendata.entitypedia.org/spacetime>

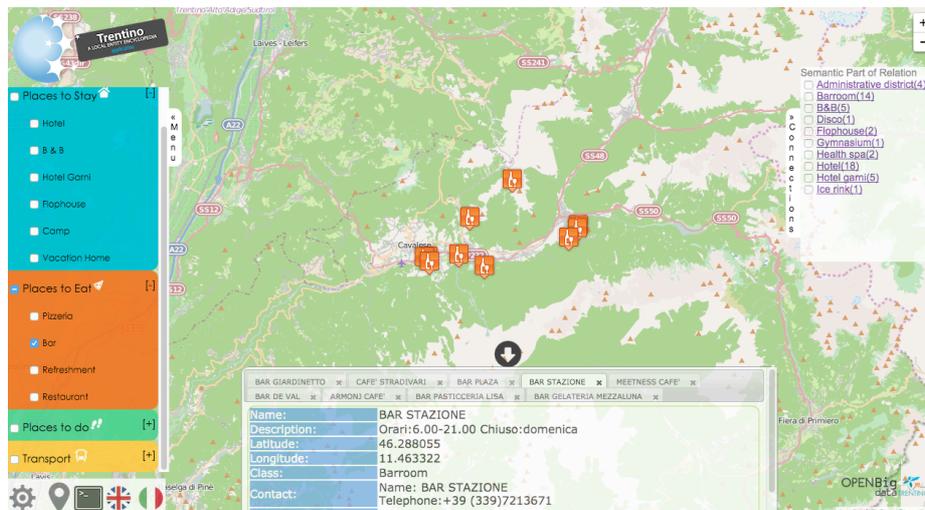
<sup>18</sup> <http://dati.trentino.it>: It is one of the major data catalogue with 875 datasets from different sectors.



**Fig. 5.** Main Architecture

components are responsible for visualizing the entities returned as the query result.

Developed prototype is shown in Figure 6. It contains four main parts: ontological menu, map, information container, and relational menu. Ontological menu is placed on the left side and contains different eTypes. It is created from a lightweight ontology. Currently, it supports Italian and English. Based on the selected menu item (eType), the entities are retrieved and visualized on the map.



**Fig. 6.** UI of the working prototype

Users can also select multiple menu items. Once an entity is selected on the map, the information about that entity is shown on the screen and relational menu is created on the right side of the screen. The relational menu lists the entities that are either near or located in the same place. There are also standard settings, such as the default language, the radius to search within, and geo-locational search for the entities

relative to the user's position. More detailed demonstration of the prototype can be seen in accompanied video.

## **5 Experiment**

The goal of the initial evaluation of the system prototype was to obtain insights on how different UX and usability dimensions are addressed, specifically, how people perceived entities and their categories in an entity centric world was understood. During the period of one week, nine students from different faculties participated in the evaluation. The users were students from Bachelors to PhD with age ranging from 21 years to 29 years. Five users were male and four were female. They were given a brief overview of the system and shown how to use the prototype application called Trentino Entitypedia (TE). The users were then asked to find entities in TE. Each session lasted maximum for half an hour. All the interaction of the users and their comments were transcribed and video was recorded for future analysis. The users were instructed to think aloud and they were asked to perform the task in normal way without feeling any discomfort or stress. After completing the task, respondents were asked to fill questionnaires. The questionnaire consists of background information and questions that use 5-point Likert Scale to assess different usability and user experience criteria. In the subsections that follow we describe the results of the evaluation.

### **5.1 Task Description**

Each user was asked to spend some time on the interface in order to get familiar with it. Then they were asked to read the instructions about the system. Four related tasks were assigned to the users as follows:

- Task 1 (T1): Find Bed and Breakfast.
- Task 2 (T2): Based on the T1, find any other Point of Interest (POI) that is in the same location (commune) as that of previously selected Bed and Breakfast.
- Task 3 (T3): Find POI within the range of 500 m in Cavalese.
- Task 4 (T4): Find Sports Club in Cavalese.

Most of the users were able to complete task 1 within minutes. They were also able to understand the relationship and to navigate through different relations. Task 2 also took less time to complete. The possibility to set the radius made the task 3 simple to be achieved and users were also excited to see the results. Task 4 was also easily performed by most of the users.

## 5.2 Usability Evaluation

On average, users' assessments were quite high for all measured variables. The overall assessment of the system was positive. Five usability dimensions were selected as follows: Usefulness, Learnability, Memorability, Satisfaction and Visibility of system status. Evaluation study is summarized in table 3. Detailed evaluation dimensions and statistics can be found elsewhere<sup>19</sup>.

**Table 3.** Usability evaluation based on five-point Likert scale.

User	Usefulness		Learnability		Memorability	Satisfaction	Visibility of System Status	
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
1	4	3	5	2	2	3	3	3
2	5	5	4	2	2	4	4	5
3	4	3	4	3	3	3	4	4
4	5	4	4	2	2	4	4	4
5	4	4	4	2	2	4	4	4
6	4	4	5	2	2	3	3	5
7	4	4	5	1	1	4	5	5
8	4	5	5	2	1	4	5	5
9	4	3	4	3	2	3	4	5
Mean	4.22	3.88	4.44	2.11	1.89	3.55	4.0	4.4
S.D	0.44	0.78	0.52	0.60	0.60	0.52	0.70	0.72

Q1 - I find the application simple to use.

Q2 - The system meets my needs.

Q3 - It is easy to learn how to use the system.

Q4 - I had to put extra effort to use the system.

Q5 - It takes some time to remember application controls.

Q6 - I didn't notice any inconsistencies while using it.

Q7 - I receive timely feedback on my actions.

Q8 - The information provided by the system is easy to understand.

Overall, the TE system was assessed as quite useful. Standard deviation from the mean value for each of the questions for the usability dimensions is also minimum. This proves that system was useful for almost all of the users.

Ontological menu on the left and the semantic menu were both considered as user-friendly. Left menu proved to be well organized as it allows user to find specific service in less time. Users also liked the possibility of setting entity search range on the system. Overall, the interactivity and the integrated environment was interesting for users. Some minor bugs were also discovered during the process. The check-box on the left menu behaved with certain issues. When an item was checked on semantic

<sup>19</sup> <http://goo.gl/6KtFLe>

menu (right menu), the users expected that the previously selected entity will also be shown on the map. Some suggestions about font size and type were also provided. The system was little slower and users expected results in less time.

For the future version, users also suggested some features. Directional features like in Google Maps, direct search functionality, facet level service on attribute level. More interactive help was also pointed out as needed. Users also requested putting icon on both the menus. Some users also suggested accommodating the relational menu (right menu) directly inside the information container.

### 5.3 UX Evaluation

The system was evaluated considering five UX dimensions i.e. Aesthetics and Visual appearance, Emotion, Identification, Stimulation and Meaning and value. Table 4 shows responses related to specific UX dimensions.

**Table 4.** UX evaluation based on five-point Likert Scale

User	Aesthetics and visual appearance		Emotion		Identification	Meaning and Value	
	Q1	Q2	Q3	Q4	Q5	Q6	Q7
1	4	4	4	3	4	3	5
2	4	5	5	4	4	4	4
3	3	4	4	3	4	4	4
4	4	4	4	4	4	4	4
5	4	4	4	4	4	4	4
6	5	5	5	4	4	4	4
7	4	5	1	3	5	5	5
8	3	5	5	5	5	5	1
9	2	4	4	4	4	4	4
Mean	3.66	4.4	4.0	3.77	4.2	4.11	3.88
S.D	0.86	0.52	1.22	0.66	0.44	0.60	1.16

Q1 - Seeing different types of entities on the same page is visually pleasant

Q2 - Icons clearly reflect the purpose of the specific entity on the map

Q3 - I like the idea that, using the system, I can discover new information connected to what I was looking for in the first place

Q4 - The fact that I can propagate through the network of connected entities makes this application somehow playful

Q5 - I see myself in using the application.

Q6 - The application gives me reliable evidence of entities in a sense that I get the information I expect

Q7 - The meaning of entities and their relations enables me to quickly navigate and extract the information I find useful

On average, the users' assessments were positive for all of the UX variables. User 7 and 8 gave negative remarks for Q3 and Q7, respectively. This resulted in higher

deviation for those questions from the mean. We explain this with the comments they have provided. User 7 noted that the system has easy traversal mechanism, but the discovery of information may not be intuitive. Similarly, User 8 mentioned that though it is easy to find entities and relations, the meaning of the words in Italian was not intuitive in some cases. For example, Cateratta was used to describe waterfall, whereas the Cascata may be an appropriate word. This issue comes from the quality of underlying entity-centric open data.

The idea of seeing entities of different types on the same page really excited the users. Users felt it is less time-consuming and can get a clear picture of what they were looking for. User 1 mentioned that the interface was intuitive and exciting. Users liked the idea of consistency between menu color with the color of an icon color which showed clarity and symmetry. For some users, the green icon was not helpful as it also matched with the color of the map. Icon for street also did not make any sense to the users. For most of the users finding new information was an innovative thing. User 6 expressed that she enjoyed the way in which connected entities can be traversed. User 4 was surprised to see the opening and closing time for POI. All users asked if they could start using the application immediately. They also suggested that it will be extremely helpful for them if the application can be made mobile. User 3 liked the menu. The arrangement of menu items seemed logical for him. User 2 was doubtful whether the menu item ‘Things to do’ was expressive. He also thought that menu ski and club should be placed together under same menu.

Table 5 gives percentage of users that indicated application features with respect to specific UX dimensions. With respect to the identification, the users were asked to choose the features that reflect their habits in using similar systems. Regarding stimulation, the users were asked to choose the specific features that encourage them to use the application.

**Table 5.** Application features rating with respect to UX dimensions

Application Feature	Identification	Stimulation
Relations/links between entities on the map	63%	63%
Entities details on demand	38%	38%
Entities view scalability	25%	25%
Entities view filtering	38%	50%
History of browsed entities	25%	13%
Flexible exploration and navigation through entities		50%
Flexibility /Contextual switch in showing entities from different domains		38%

The table shows that the most interesting application features are relations between entities, view filtering and flexible exploration and navigation through entities. It also shows that some of the features were not noticed by the users. These features require further improvement.

## 6 Conclusion and Future Work

The paradigm shift to make the data free raises an issue of the effective usage by common users. This requirement comes from the fundamental properties of open data - diversity in format and content and unexpectedness (meaning they are not known in advance). Thus far, they are handled with predefined, hard-wired solutions.

We handle open data explicitly through entities as a domain-independent and a user-centric methodology. From the data perspective, entities address fundamental issues of open data. By design, they capture context very well by encapsulating all the relevant properties (attributes) in a component. Once designed, they serve as good data aggregators (such as People, Locations, Events, Facilities). Moreover, different kinds of relations among entities make them usable across domains. From the usability perspective, we perceive that people intuitively think in terms of entities as objects (such as friends, events and places) and we aim to exploit this notion in human mental model.

In this paper, we have proposed an entity-centric solution to visualize open data. The solution is supported by the high-level, reference architecture. Upon the architecture, we have developed proof-of-concept prototype. Insights that are gained from the user study demonstrate the feasibility of our solution.

Our next step is to design entity-centric UI framework that will use generic UI components - entigets to improve development of open data applications for different domains and contexts of use.

## References

1. M. Agrawala, W. Li, and F. Berthouzoz. Design principles for visual communication. *Communications of the ACM*, 54(4):60–69, 2011.
2. L. Alben. Defining the criteria for effective interaction design. *interactions*, 3(3):11–15, 1996.
3. R. A. Amar and J. T. Stasko. Knowledge precepts for design and evaluation of information visualizations. *IEEE Trans. Vis. Comput. Graph.*, 11(4):432–442, 2005.
4. C. Becker and C. Bizer. Exploring the geospatial semantic web with dbpedia mobile. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(4):278–286, 2009.
5. I. Bedini, T. Rise, F. Farazi, I. Tankoyeu, D. Leoni, S. Leucci, F. Gleria, and P. A. di Trento. Open government data: Fostering innovation. 2014.
6. T. Berners-Lee, J. Hollenbach, K. Lu, J. Presbrey, and M. Schraefel. Tabulator redux: Browsing and writing linked data. 2008.
7. T. Berners-Lee. Linked data-design issues (2006). URL <http://www.w3.org/DesignIssues/LinkedData.html>, 2011.
8. C. Bizer, T. Heath, and T. Berners-Lee. Linked data - the story so far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22, 2009.
9. N. Cao, J. Sun, Y.-R. Lin, D. Gotz, S. Liu, and H. Qu. Facetatlas: Multifaceted visualization for rich text corpora. *Visualization and Computer Graphics, IEEE Transactions on*, 16(6):1172–1181, 2010.

10. S. K. Card, J. D. Mackinlay, and B. Shneiderman, editors. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.
11. A.-S. Dadzie and M. Rowe. Approaches to visualising linked data: A survey. *Semant. web*, 2(2):89–124, Apr. 2011.
12. S. Davies, J. Hatfield, C. Donaher, and J. Zeitz. User interface design considerations for linked data authoring environments. *LDOW*, 628, 2010.
13. T. Davies. Supporting open data use through active engagement. In *Position paper: W3C Using Open Data Workshop*. W3C, 2012.
14. P. M. Desmet and P. Hekkert. Framework of product experience. *International Journal of Design*, 1(1):57–66, 2007.
15. M. S. F. Farazi. *Faceted Lightweight Ontologies: A Formalization and some Experiments*. PhD thesis, University of Trento, 2010.
16. J.-D. Fekete, J. J. Van Wijk, J. T. Stasko, and C. North. The value of information visualization. In *Information visualization*, pages 1–18. Springer, 2008.
17. R. Garca, J. M. Gimeno, F. Perdrix, R. Gil, and M. Oliva. The rhizomer semantic content management system. In *Emerging Technologies and Information Systems for the Knowledge Society*, pages 385–394. Springer, 2008.
18. B. Gaver and H. Martin. Alternatives: exploring information appliances through conceptual design proposals. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 209–216. ACM, 2000.
19. F. Giunchiglia and B. Dutta. Dera: A faceted knowledge organization framework. 2011.
20. F. Giunchiglia, V. Maltese, and B. Dutta. Domains and context: first steps towards managing diversity in knowledge. *Web Semantics: Science, Services and Agents on the World Wide Web*, 12:53–63, 2012.
21. W. Halb, Y. Raimond, and M. Hausenblas. Building Linked Data For Both Humans and Machines. In *WWW 2008 Workshop: Linked Data on the Web (LDOW2008)*, Beijing, China, 2008.
22. W. Halb, Y. Raimond, and M. Hausenblas. Building linked data for both humans and machines. In *LDOW*, 2008.
23. M. Hassenzahl and N. Tractinsky. User experience-a research agenda. *Behaviour & Information Technology*, 25(2):91–97, 2006.
24. M. Hassenzahl. The thing and i: understanding the relationship between user and product. In *Funology*, pages 31–42. Springer, 2005.
25. T. Hastrup, R. Cyganiak, and U. Bojars. Browsing linked data with fenfire. 2008.
26. M. Hearst. Design recommendations for hierarchical faceted search interfaces. In *ACM SIGIR workshop on faceted search*, pages 1–5. Seattle, WA, 2006.
27. T. Heath. How will we interact with the web of data? *Internet Computing, IEEE*, 12(5):88–91, 2008.
28. J. Heer, M. Bostock, and V. Ogievetsky. A tour through the visualization zoo. *Commun. ACM*, 53(6):59–67, 2010.
29. P. Hekkert. Design aesthetics: principles of pleasure in design. *Psychology science*, 48(2):157, 2006.
30. V. Jääskö and T. Mattelmäki. Observing and probing. In *Proceedings of the 2003 international conference on Designing pleasurable products and interfaces*, pages 126–131. ACM, 2003.
31. E. Karapanos, J. Zimmerman, J. Forlizzi, and J.-B. Martens. Measuring the dynamics of remembered experience over time. *Interacting with Computers*, 22(5):328–335, 2010.

32. T. Lavie and N. Tractinsky. Assessing dimensions of perceived visual aesthetics of web sites. *International journal of human-computer studies*, 60(3):269–298, 2004.
33. S. Mahlke and M. Thüring. Studying antecedents of emotional experiences in interactive contexts. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 915–918. ACM, 2007.
34. H. Marc. Aesthetics in interactive products: Correlates and consequences of beauty.
35. S. Mazumdar, D. Petrelli, and F. Ciravegna. Exploring user and system requirements of linked data visualization through a visual dashboard approach. *Semantic Web*, 5(3):203–220, 2014.
36. B. Otjacques, M. Stefas, M. Cornil, and F. Feltz. Open data visualization keeping traces of the exploration process. In *Proceedings of the First International Workshop on Open Data*, pages 53–60. ACM, 2012.
37. L. Rutledge, J. van Ossenbruggen, and L. Hardman. Making rdf presentable: Integrated global and local semantic web browsing. In *Proceedings of the 14th International Conference on World Wide Web, WWW '05*, pages 199–206, New York, NY, USA, 2005. ACM.
38. K. M. Sheldon, A. J. Elliot, Y. Kim, and T. Kasser. What is satisfying about satisfying events? testing 10 candidate psychological needs. *Journal of personality and social psychology*, 80(2):325, 2001.
39. B. Shneiderman. *Designing the user interface: strategies for effective human-computer interaction*, volume 2. Addison-Wesley Reading, MA, 1992.
40. M. G. Skjæveland. Sgvizler: A javascript wrapper for easy visualization of sparql result sets. In *Extended Semantic Web Conference*, 2012.
41. Socrata. Open Government Data Benchmark Study Report. Technical report, 2010.
42. M. Stuhr, D. Roman, and D. Norheim. Lodwheel–javascript-based visualization of rdf data–. 2011.
43. G. Tummarello, R. Cyganiak, M. Catasta, S. Danielczyk, R. Delbru, and S. Decker. Sigma: Live views on the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(4):355–364, 2010.
44. M. Winckler, C. Bach, and R. Bernhaupt. Identifying user experience dimensions for mobile incident reporting in urban contexts. *Professional Communication, IEEE Transactions on*, 56(2):97–119, 2013.