AN INTEGRATED APPROACH FOR LINKED DATA BROWSING

W. Beek[a,c,d], E. Folmer[a,b]

[a] Kadaster, Apeldoorn, Netherlands
[b] University of Twente, Twente, Netherlands (e.j.a.folmer@utwente.nl)
[c] Triply, Amsterdam, Netherlands
[d] VU University Amsterdam, Amsterdam, Netherlands (w.g.j.beek@vu.nl)

Commission VI, WG VI/4

KEY WORDS: Data Browsing, Faceted Browsing, Geo-spatial Data, Graph Navigation, Linked Open Data

ABSTRACT:

The Netherlands' Cadastre, Land Registry and Mapping Agency – in short Kadaster – collects and registers administrative and spatial data on property and the rights involved. Currently, the Kadaster is publishing its geo-spatial data assets as Linked Open Data. The Kadaster manages hundreds of datasets that describe hundreds of millions of geospatial objects, including all Dutch buildings, roads, and forests.
The Kadaster exposes this large collection of data to thousands of daily users that operate from within different contexts and that need to be supported in different use cases. Therefore, Kadaster must offer diverse, yet complementary, approaches for browsing and exploring the data it publishes. Specifically, it supports the following paradigms for browsing and exploring its data assets: hierarchical browsing, graph navigation, faceted browsing, and tabular browsing. These paradigms are useful for different tasks, cover different use cases, and are implemented by reusing and/or developing Open Source libraries and applications.

1. INTRODUCTION

The Netherlands' Cadastre, Land Registry and Mapping Agency – in short Kadaster[1] – collects and registers administrative and spatial data on property and the rights involved. This includes ships, aircraft and telecommunications networks. Doing so, Kadaster protects legal certainty. The Kadaster publishes many large authoritative datasets including several key registers of the Dutch Government (Topography, Addresses and Buildings). Furthermore Kadaster is also developing and maintaining the PDOK shared service, in which about 100 spatial datasets are being published in several formats, including an incredible amount of detailed geospatial objects. Geospatial objects include all plots of land, all buildings, all roads and all lampposts. These objects are spatially and/or conceptually related, but are maintained by different data curators. As a result these datasets are syntactically and architecturally disjoint, and using them together currently requires non-trivial human labor.

Because the Kadaster exposes this large collection of data to thousands of daily users that operate from within different use cases, it must offer diverse, yet complementary, approaches for browsing and exploring the data (Marie and Gandon, 2014). Specifically, it supports the following paradigms for browsing and exploring its data assets: hierarchical browsing, graph navigation, faceted browsing, and tabular browsing. These paradigms are useful for different tasks and cover different use cases.

The rest of this paper is structured as follows. Section 2 explains the facilities Kadaster provides for browsing schema data. Section 3 explains the facilities Kadaster provides for browsing instance data. Section 4 relates the supported data browsing paradigms to categories of use cases. Section 5 concludes.

[1] See http://kadaster.nl

2. BROWSING SCHEMA DATA

In this section we discuss the approaches for browsing schema data. In Linked Data a schema is a collection of classes and properties that model a certain domain. Classes and properties are denoted by IRIs that all share the same prefix, which forms the namespace of the schema. An example of a schema is the Dutch Base Registry for Buildings (BAG). It contains classes, e.g., 'building', and properties that instances of the classes can have. For example, a building can have a build date and a status (e.g., occupied/unoccupied).

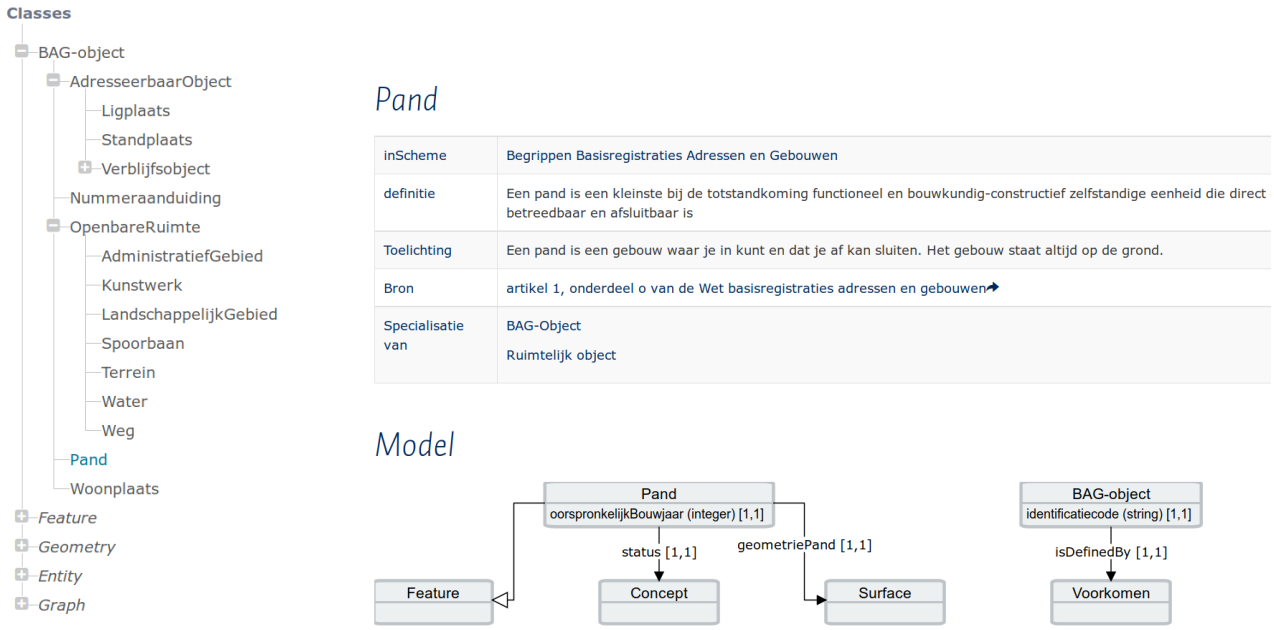
Hierarchical browsing uses the tree structure of the concept hierarchy in order to display the various classes and properties that are in the data. A hierarchical browser gives the user a quick overview of the main classes and properties that are in a dataset. It also allows the user to unfold specific classes and properties that she is interested in, in order to reveal their subclasses and subproperties. Hierarchical browsing works well for gaining an understanding of a concept schema.

Figure 1 shows the hierarchical browser from the Linked Data Theater[2] (LDT), that is developed in a collaboration between the Kadaster and Ordina[3]. In LDT users can browse the class and property hierarchies of each Linked Open Dataset. In Linked Data, every class and every property is denoted by an IRI. Following the Linked Data best practice of IRI dereferencing, the LDT allows the definition of each schema term to be looked up simply by clicking on their IRI name in a web browser.

LDT has already seem significant uptake, which several other Dutch governmental agencies deploying it to serve their Linked Data browsing need.

[2] See https://github.com/architolk/Linked-Data-Theatre
[3] See https://www.ordina.nl

Figure 1. A view of the Linked Data Theater hierarchical browser. The left-hand side shows a tree widget that represents the subclass hierarchy of the Ditch Base Registry for Buildings (BAG), from which the user has selected the building class (Dutch: "Pand"). The right-hand side shows a tabular and a schematic description of that class.



Graph navigation for schema data works well for gaining an overview of how certain key concepts are related. This is similar to the modeling of database schema's in modeling languages like UML, although there are differences between UML and the RDF(S) and OWL formalisms. The main difference is that schema definitions in RDF(S) and OWL cannot be used to data data shapes. Formally speaking, these traditional Linked Data schema's are not defined as constraints that pose strictures on the data, but as entailment rules that are used to derive new facts through entailment. However, the recent (and ongoing) standardization of Linked Data validation languages such as SHACL[4] and ShEX[5] provides new opportunities for formulating data shapes. Linked Data Theater comes with SHACL definitions for each of the Kadaster base registries, and therefore allows for UML-like descriptions of classes and properties such as cardinality constraints. An example of this is given on the right-hand side of Figure 1.

3. BROWSING INSTANCE DATA

In this section we discuss the approaches for browsing instance data. In Linked Data, instances are particular objects. These particulars are related to one or more schema's, because a particular is always an instance of one or more classes. For example, 'Amsterdam' is an instance of class 'Municipality'.

Faceted browsing (Heim et al., 2008) is an advanced data browsing technique that turns the properties that appear in a dataset into widgets that can be set by the user in order to formulate a conjunctive set of filters over the set of instances that is described in the data. Because of the intersection semantics of faceted browsing, it is possible to narrow a set of millions of instances down to a set of only a handful of relevant instances, by setting only three well-chosen filters. Faceted browsing works well for searching a specific instance.

Existing faceted browsers require a per-dataset configuration. This does not scale to the hundreds of datasets maintained by the Kadaster. For this reason, Triply[6] has built FacetCheck, a faceted browser that configures itself. When FacetCheck is run for the first time, it streams through the entire dataset in order to extract the data schema. The schema is stored in the Shapes Constraint Language (SHACL), a formal specification of the data schema that can be understood by machines. Based on the SHACL definition, a web UI is generated that is custom-tailored towards a particular dataset. Depending on the range of a property, different UI widgets are used: a slider for numbers, a calendar for dates, a checkbox for Booleans, etc. The widgets are drawn from the Open Source Material UI library[7].

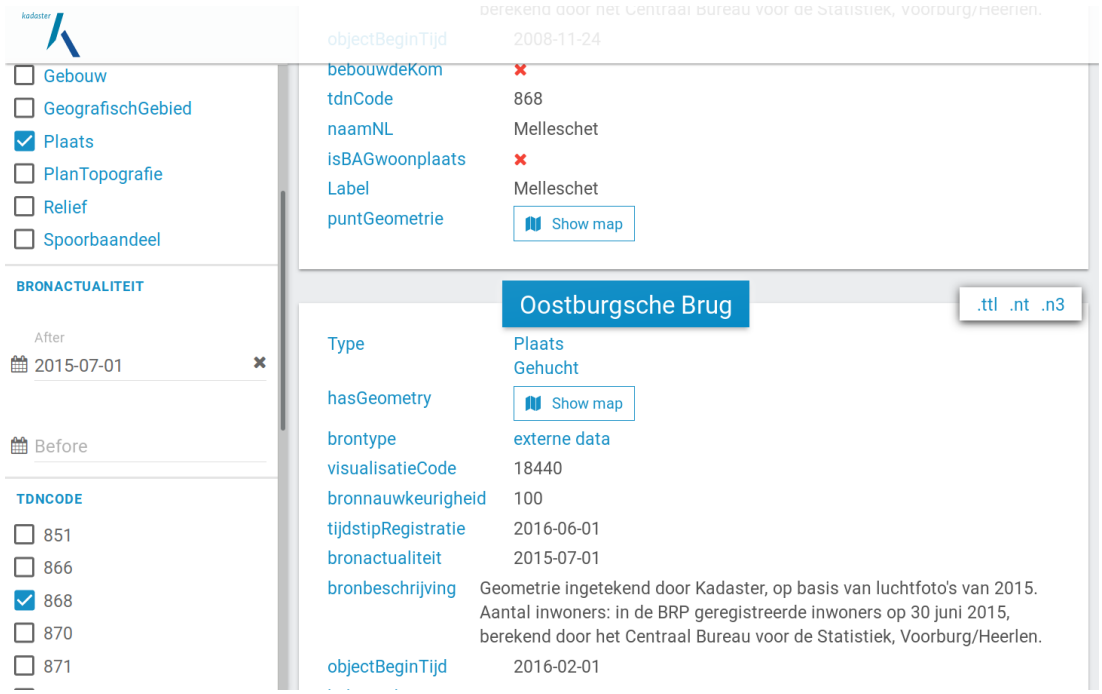
Figure 2 shows an example in which Dutch places whose record has been updated on or after the first of July 2015, and whose TDN code is 868. Even through there are millions of instances that are described in the Kadaster data collection, it is often possible to drill down to a small subset of them by setting three or four automatically generated widgets.

The main benefit of the faceted browsing paradigm for Linked Data is that the user only has to select dates, check boxes, and shift sliders in order to add a complicated conjunctive filter over the set of instances. To illustrate the benefit of this, we express the selection in Figure 2 in terms of a SPARQL (Garlik et al., 2013) query (SPARQL is the standardized query language for Linked Open Data):

```
select distinct ?s {
  ?s a def:Place ;
     def:TDN_code ?tdn .
  filter (?tdn == 868 &&
          ?updated >= "2015-07-01"^^xsd:date)
}
```

[4] https://www.w3.org/TR/shacl/
[5] http://shex.io/
[6] See https://triply.cc
[7] See https://github.com/callemall/material-ui

Figure 2. A view of the FacetCheck faceted browser. Selections that correspond to conjunctive filter expressions are made on the left-hand side. In this example this results in a list of places whose record has been updated on or after the first of July 2015, and whose TDN code is 868.




```
sort by desc(?s)
limit 100
```

Tabular browsing is a popular approach for displaying database content (Berners-Lee et al., 2006). Instances can be displayed as rows in a table, where properties are displayed as columns, and values of instances are displayed as table cells. This approach is used extensively in spreadsheet software.

Tabular browsing is also known to have limited utility for Linked Data: because the number of properties can be very large, it is generally not possible to show columns for all properties. In practice this is solved by displaying a small subset of the columns that could be shown, e.g., based on the classes to which an instance belongs. An example of this is given in Figure 2, where the per-instance panel shows some but not all property/value pairs in an HTML table (properties are displayed in blue, while values are displayed in black).

Besides FacetCheck, tabular browsers are also used in Linked Data Theater to show the results of dereferencing resource IRIs. The basis of IRI dereferencing is to use the depth-one links for a given resource-denoting IRI, possibly closed under RDF blank nodes, resulting in the so-called Concise-Bounded Description (CBD) (Stickler, 2005) of a resource. Linked Data Theater extends this notion of a CBD by grouping together properties that conceptually belong together (e.g., the version information of an object is displayed separately). LDT provides extensive configurability of these tabular views.

Instance-based **graph navigation** uses the assertional graph-shape of the RDF datamodel, under appropriate abstractions, to display instances as nodes, and properties between instances as edges between nodes. A user can browse the graph by clicking on nodes, thereby expanding the relationships (edges) of that node, which exposing new nodes to click on, etc.

Graph navigation for instance data is different from graph navigation for schema data (Section 2). In a schema the number of nodes and edges is relatively small. Moreover, there are only few types of edges (subclass, subproperty, etc). In instance-based graph navigation there can be arbitrarily many nodes (e.g., millions), and there can be a large number of edge types (e.g., hundreds). For these reasons, graph navigation for instance data is mainly used for *explorative browsing*, while graph navigation for schema data is mainly used to obtain an overview of a conceptualization.

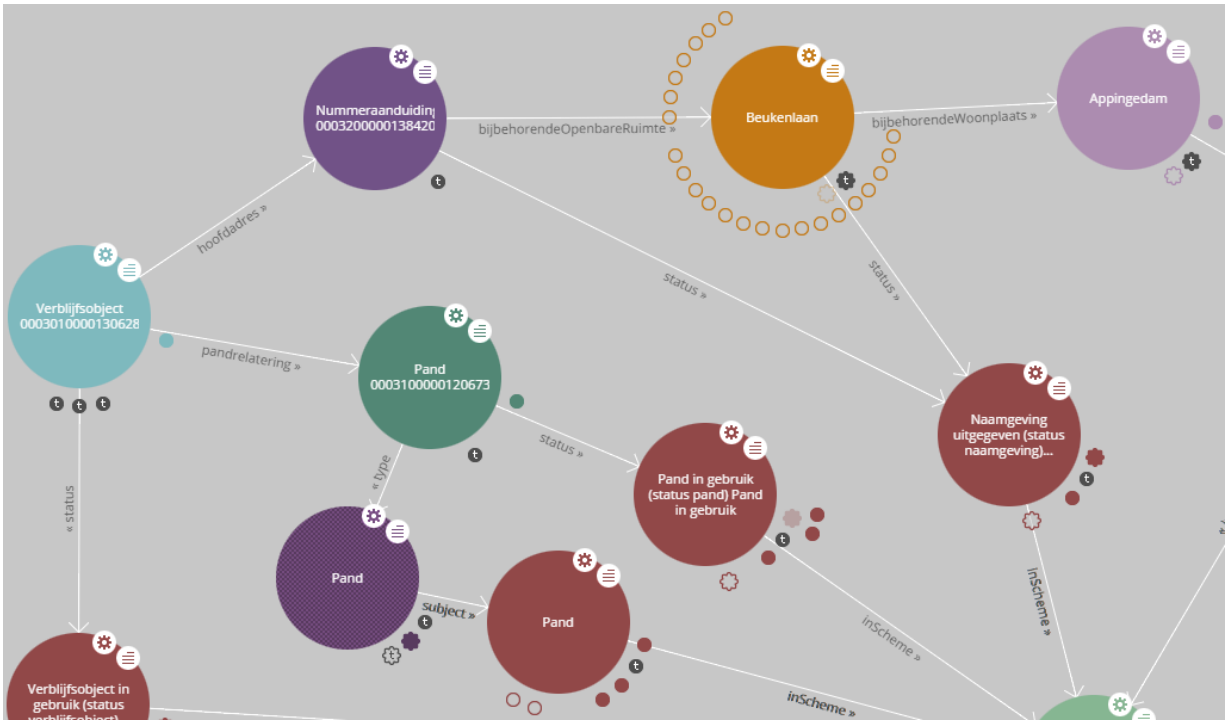
For graph navigation of instance data the GRID project uses the existing LodLive (Camarda et al., 2012) application. Figure 3 shows a view of LodLive, where a Dutch building and its relationships in the Base Registry for Buildings (BAG) are shown.

4. USE CASES FOR DATA BROWSING

Since the Kadaster wants to support a large group of users working in multiple use cases, it implements complementary forms of Linked Data browsing into one integrated platform. It does so by reusing, combining and contributing to state-of-the-art Open Source solutions. In terms of the browsing paradigms that have been described in Sections 2 and 3, we can identify the following classes of use cases where these paradigms can be applied:

Obtaining an overview of a conceptualization Sometimes a user wants to gain insight into what kind of information is contained in a certain dataset. Alternatively, a user knows what is contained in a certain dataset, but is not yet familiar with the way in which the data is structured. In such use cases the hierarchical browser makes most sense, because it gives a quick overview of a dataset schema. The hierarchical browser can be complemented with a tabular overview of the definitions of individual classes and properties, and may be fur-

Figure 3. A view of the LodLive graph browser. Each node represents a resource. Edges between nodes denote properties that hold between resources.



ther supplemented with graph navigation to a limited depth (typically one or two hops from the target concept).

Obtaining an authoritative definition Sometimes a user is looking for the official definition of a given instance, class or property. This is especially the case with Kadaster data, which has a legal interpretation. In such use cases the tabular browser makes most sense, because this shows all the data about a given resource. The tabular approach may be combined with graph navigation until a limited depth (typically two or three hops from the target instance).

Drilling down to one instance Sometimes a user is looking for one particular instance, or for a small number of instances, of which the user already knows some properties. In such use cases the faceted browser makes most sense, because it allows the user to drill down to a small number of instances by selecting concrete values (or value ranges) for one or more properties.

Exploration Sometimes a user does not have a concrete information question but takes a generic interest in a certain data collection. In such cases graph navigation with unlimited traversal depth is appropriate. This allows the user to dive into parts of the dataset that are arbitrarily far removed from her point of entry, and that she did not anticipate visiting in the beginning.

5. CONCLUSION

The Netherland's Land Registry and Mapping Agency, or Kadaster, exposes a large collection of geo-spatial Linked Open Data to thousands of daily users. Because these users operate from within different contexts and need to be supported in different use cases, the Kadaster offers diverse, yet complementary, approaches for browsing and exploring the data. It uses a combination of hierarchical browsing, graph navigation, faceted browsing, and tabular browsing in order to support obtaining overviews, authoritative definitions, drilling down to instances, and open-ended exploration.

All software that is used and/or developed by Kadaster is published as Open Source, e.g., under the MIT license. Please visit the Kadaster Data Platform over at `https://data.pdok.nl` to start browsing the data!

REFERENCES

Berners-Lee, T., Chen, Y., Chilton, L., Connolly, D., Dhanaraj, R., Hollenbach, J., Lerer, A. and Sheets, D., 2006. Tabulator: Exploring and analyzing linked data on the semantic web. In: *Proceedings of the 3rd international semantic web user interaction workshop*, Vol. 2006, Citeseer, p. 159.

Camarda, D. V., Mazzini, S. and Antonuccio, A., 2012. LodLive, exploring the Web of Data. In: *Proceedings of the 8th International Conference on Semantic Systems*, ACM, pp. 197–200.

Garlik, S. H., Seaborne, A. and Prud'hommeaux, E., 2013. SPARQL 1.1 query language. *World Wide Web Consortium*.

Heim, P., Ziegler, J. and Lohmann, S., 2008. gfacet: A browser for the web of data. In: *Proceedings of the International Workshop on Interacting with Multimedia Content in the Social Semantic Web (IMC-SSW'08)*, Vol. 417, Citeseer, pp. 49–58.

Marie, N. and Gandon, F., 2014. Survey of linked data based exploration systems. In: *Proceedings of the 3rd International Conference on Intelligent Exploration of Semantic Data-Volume 1279*, CEUR-WS. org, pp. 66–77.

Stickler, P., 2005. CBD - Concise Bounded Description. *W3C Member Submission* 3, pp. 29.