LINKED DATA VIEWING AS PART OF THE SPATIAL DATA PLATFORM OF THE FUTURE

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

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

Commission IV, WG IV/4

KEY WORDS: Linked Data, geospatial data, GIS viewing, faceted browsing, data publishing

ABSTRACT:

The Land Registry and Mapping Agency of the Netherlands ('Kadaster' in Dutch) is developing an online publication platform for sharing its geospatial data assets called KDP (`Kadaster Data Platform' in Dutch). One of the main goals of this platform is to better share geospatial data with the wider, web-oriented world, including its developers, approaches, and standards. Linked Open Data (W3C), GeoSPARQL (OGC), and Open APIs (OpenAPI Specification) are the predominant standardized approaches for this purpose. As a result, the most important spatial datasets of the Netherlands – including several key registries – are now being published as Linked Open Data that can be accessed through a SPARQL endpoint and a collection of REST APIs. In addition to providing raw access to the data, Kadaster Data Platform also offers developers functionalities that allow them to gain a better understanding about the contents of its datasets. These functionalities include various ways for viewing Linked Data . This paper focuses on two of the main components the Kadaster Data Platform is using for this purpose: FacetCheck and Data Stories.

1. INTRODUCTION

The Land Registry and Mapping Agency of the Netherlands ('Kadaster' in Dutch) publishes many large authoritative geospatial datasets, including several key registers of the Dutch Government. This includes a detailed description of the full topography of the Netherlands, as well as registrations of all the addresses and buildings in the Netherlands. These data assets are published in the online PDOK data catalog (https://data.pdok.nl). PDOK is a data publication service that exposes over 130 geospatial datasets form various Dutch governmental institutes. Together, these datasets include descriptions of hundreds of millions of geospatial objects.

The Kadaster Data Platform (KDP) is publishing datasets as Linked Open Data not only for the Dutch Cadastre, but also as a shared government service for other Dutch governmental organizations. An example of the latter is the first Linked Open Data release of the spatial-statistical dataset of Dutch neighborhoods (called "Kerncijfers wijken en buurten" in Dutch) published by the Dutch Central Bureau of Statistics (https://cbs.nl). In addition to providing access to the raw data, the KDP is offering various support features to further assist its users and developers. These features include a metadata browsers, a graph viewer, a query editor, and a discussion forum. Still, in practice users and developers experience difficulties when trying to view (certain aspects of) the data. In order to address this issue, the KDP team has developed an approach for telling stories based on Linked Open Datasets (Data Stories, as well as a faceted browser (FacetCheck). Both components are based on Open Source libraries, and are themselves published as Open Source projects for others to reuse.

2. ARCHITECTURE

The Data Stories (Section 3) and FacetCheck (Section 4) tools are components within the larger Kadaster Data Platform (KDP) architecture (Figure 1). As is the case with many other geospatial organizations, the Kadaster currently publishes the majority of its 130 open datasets using one of the GIS-specific formats standardized by the OGC (e.g., GML). These formats are popular with GIS specialists, but are not used on a wider scale, e.g., by web developers. In the new Kadaster Data Platform (KDP) these existing formats will therefore be extended upon, to also offers Linked Open Data formats (i.e., RDF), as well as queryable REST APIs. In order to effectuate this process, an ETL procedure was designed that allows existing data assets to be transformed and loaded into an RDF triple store (GraphDB), as well as a document store (ElasticSearch). Based on these newly created Linked Data access points, it is possible to define novel 'Information Products', i.e., specific APIs and/or applications that are created with a specific business goal in mind. Because all data is semantically described as Linked Data, it is relatively easy to combine various datasets into one Information Product. An Information Product consists of a set of integrated Linked Data queries, that are exposed through a REST API that uses the OpenAPI specification. This is also where the main cost saving property of Linked Data resides: it significantly reduces the cost of integrating heterogeneous data sources with the purpose of generating new APIs. This is particularly useful when there are multiple business goals that need to be covered at the same time, and/or when the current business goal may change over time. The content of the integrated RDF triple store is exposed through a SPARQL endpoint. On top of this endpoint, the KDP had already implemented various front-end functionalities. Specifically, the following three Linked Data browsing paradigms were earlier introduced (Beek & Folmer, 2017): tabular browsing, hierarchical browsing, and graph navigation. Tabular browsing is a simple yet popular way for browsing database content, which displays records in rows and properties in columns. In addition to record-oriented tabular browsing, hierarchical browsing makes use of the tree structure of the concept and property hierarchies in order to display the various classes and properties that are present in the data. As such, a hierarchical browser gives the user a quick overview of the main classes and properties that are in a

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLII-4-W8-49-2018 | © Authors 2018. CC BY 4.0 License.

dataset. Hierarchical browsing works well for gaining an understanding of a concept schema. Both the tabular and hierarchical browser are implemented by the Linked Data Theatre (https://github.com/architolk/Linked-Data-Theatre). Graph navigation uses the graph-shape of the RDF data model in order to display concepts and instances as nodes, and properties as edges between those nodes. Graph navigation was observed to work well for explorative browsing, e.g., it allows the discovery of links to other datasets. For graph navigation LODLive (http://en.lodlive.it) is used. In addition to these data browsing approaches, the KDP also includes a SPARQL query editor with specific support for GeoSPARQL queries and geospatial visualizations of the query result set (W. Beek et al., 2017).

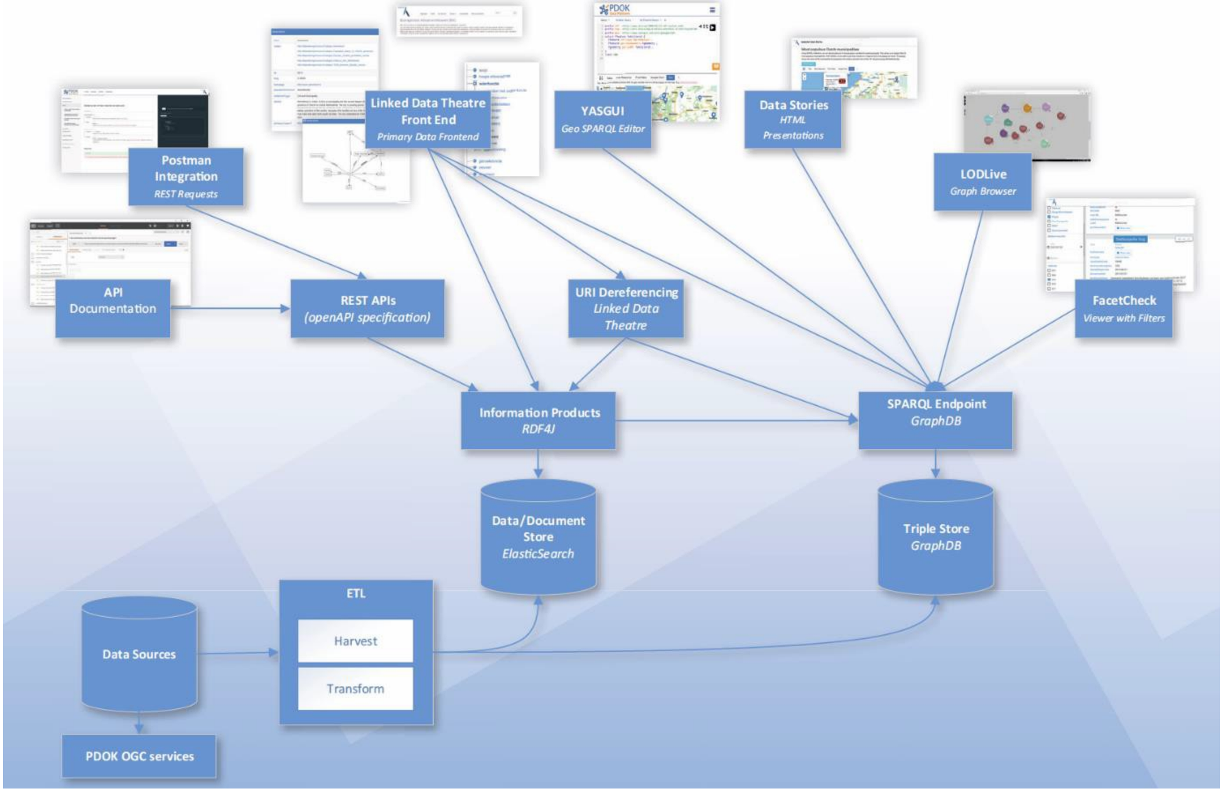


Figure 1. The Kadaster Data Platform (KDP) Architecture

3. DATA STORIES

Since Linked Data is a relatively new technology for most users, many of them are unaware of the potential that can be unlocked when querying geospatial and statistical Linked Data. Users are observed to have difficulty with determining whether or not a Linked Dataset is useful for there own use case. With the browsing features described in Section 2, a first step towards becoming familiar with a new dataset is to browse through that datasets' metadata description. A second step consists of browsing through dataset-specific data model, i.e., the concept and property hierarchy. Unfortunately, this approach is relatively complicated, since it requires a user to be able to identify the usefulness of a dataset based on the concepts it contains. For many users, a concept hierarchy does not immediately translate to potential use cases of the data. Furthermore, a dataset may contain a large number of concepts and/or properties. In such cases, a user may miss those parts of the data model that are most important to her use case.

In order to bridge the gap between (a) the vast but implicit potential that a Linked Dataset encapsulates, and (b) the specific and often more explicit use cases a prototypical user may have in mind, the KDP has developed Data Stories. A Data Story allows a specific use case to be explained to a potential user through a sequence of data examples that are connected by an overarching story. In order to be a generic as possible, data examples are taken to be the result sets of SPARQL queries. This ensures that the components of a Data Story are declarative, reproducible, and modifiable.

A Data Story allows the original data publishers to emphasize the potential use cases that they see for their datasets. This includes their ability to highlight interesting aspects of the dataset itself, e.g., interesting objects and/or interesting relationships between objects, as well as interesting ways in which the data can be combined with other Linked Data sources (e.g., DBpedia). A Data Story can be thought of a an 'advertisement of data'. It consists of a textual description/explanation of the story line, interspersed with SPARQL queries. When a story is read, the SPARQL queries are executed in sequence, and their result sets are displayed inline. During the creation of a Data Story, the writer can choose to visualized the results of queries in tables, diagrams/charts, pivot tables, or Leaflet maps. As such, many different types of information ,e.g., geographical and statistical information, can be combined in order to tell an engaging story with data. An example of such a combination is a thematic map, in which a statistical property is used in order to color the regions of a map. While it is possible to create diagrams with statistics program, and thematic maps with GIS toolkits, the queries in Data Stories are encoded in a standardized query language. For each of the displayed query results, an editor that contains the query itself can be opened. When a query is changed, the results of the change are calculated on the spot, making the elements of a Data Story more interactive/modifiable than their read-only counterparts from regular articles.

Many examples of concrete Data Stories can be found online: https://data.labs.pdok.nl/stories.

4. FACETCHECK

Since Linked Data does not have a static schema, each dataset can be structured in a different way. This provides great flexibility to the data publisher, and allows a wide variety of datasets to be published with high semantic detail, without requiring the introduction of non-standardized and/or domain-specific constructs. Unfortunately, on the side of the data consumer this great flexibility makes it difficult to understand how a specific dataset is structured, and how it can be queried.

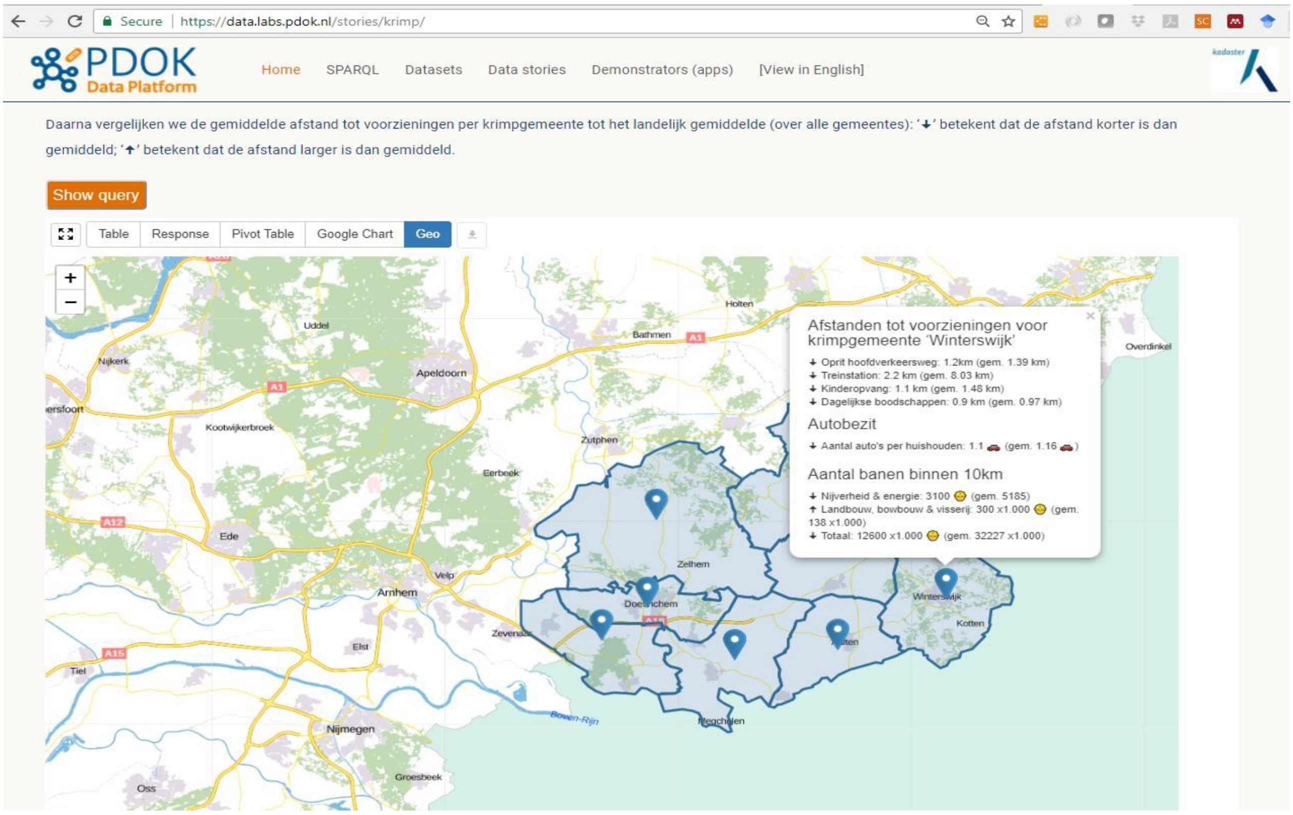
The problem of querying an unfamiliar schema is already ‘solved’ on today’s web by faceted browsers. For example, when a customer wants to buy a television, many store websites allow the customer to search for a television based on various properties such as minimum rating, price, weight, screen resolution, and screen diameter. Customers are able to express a relatively complicated SQL query by interacting with various widgets (check boxes and sliders) within the web UI.

Faceted browsers can easily be created when the database schema is stable: an application developer can create widgets that correspond to query filters. Selecting multiple facets results in a (conjunctive) composite query over the set of data entities. As such, creating a faceted browser is a relatively expensive and time consuming process since it requires non-trivial development effort for each database. In Linked Data, the properties in the database are described in semantic terms. For example, standards-compliant Linked Data specifies the domain and range types for each property. Based on this semantic description, the faceted browser widgets can be generated automatically.

FacetCheck is a specific implementation that maps semantic descriptions onto UI widgets and underlying SPARQL sub-queries. The FacetCheck UI consists of two components: the left hand side of the screen contains the various widget, while the right hand side of the screen displays the entities that conform to the specified filters. When making selections within the FacetCheck UI, a SPARQL query is automatically assembled out of the sub-queries associated to each widget. A limited number of entities that adhere to the specified query are retrieved and displayed for inspection by the user. An instance is also displayed by a compositional widget. The components of an entity widget are determined by the direct properties the corresponding entity has in the database. (This is sometimes referred to as the ‘Concise Bounded Description’ of an entity.) Based in the displayed entity widgets, the user can decide whether these are the entities she is looking for, or whether (other) widgets need to be set or changed in order to improve the results. Since FacetCheck allows the set of selection widgets and entity widgets to be generated automatically, it is relatively easy to create a FacetCheck browser over a specific Linked Dataset.

5. CASE STUDY: DUTCH NEIGHBORHOODS DATASET

An example configuration of FacetCheck can be used online (https://facetcheck.triply.cc). Currently configurations for several KDP datasets exist, including one over the spatial-statistical dataset of Dutch neighbourhoods (“Kerncijfers wijken en buurten” in Dutch). This dataset links geospatial data assets of the Land Registry and Mapping Agency to statistical data from the National Statistics Office (CBS). In November 2017, two data journalists were invited to express their interests in Dutch neighbourhood data. They were in particular interested in data



Daarna vergelijken we de gemiddelde afstand tot voorzieningen per krimpgemeente tot het landelijk gemiddelde (over alle gemeentes): ‘↓’ betekent dat de afstand korter is dan gemiddeld; ‘↑’ betekent dat de afstand larger is dan gemiddeld.

Afstanden tot voorzieningen voor krimpgemeente ‘Winterswijk’

↓ Oprit hoofdverkeersweg: 1.2km (gem. 1.39 km)
↓ Treinstation: 2.2 km (gem. 8.03 km)
↓ Kinderopvang: 1.1 km (gem. 1.48 km)
↓ Dagelijkse boodschappen: 0.9 km (gem. 0.97 km)

Autobezit

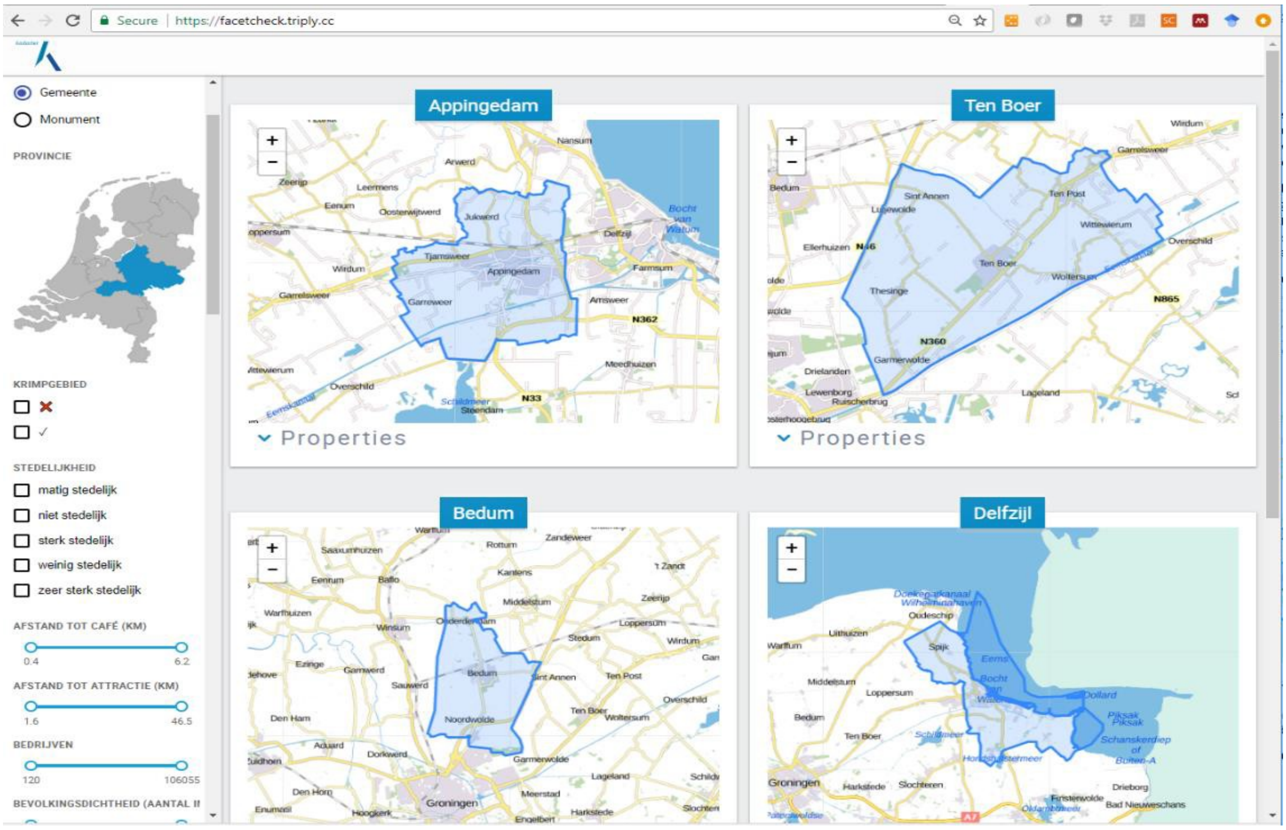
↓ Aantal auto's per huishouden: 1.1 🚗 (gem. 1.16 🚗)

Aantal banen binnen 10km

↓ Nijverheid & energie: 3100 😊 (gem. 5185)
↑ Landbouw, bowbouw & visserij: 300 x1.000 😊 (gem. 138 x1.000)
↓ Totaal: 12600 x1.000 😊 (gem. 32227 x1.000)

Figure 2. Screenshot from the depopulation Data Story: characteristics of depopulation areas

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLII-4-W8-49-2018



Secure | https://facetcheck.triply.cc

● Gemeente
○ Monument

PROVINCIE

KRIMPGEBIED
□ ✗
□ ✓

STEDELIJKHEID
□ matig stedelijk
□ niet stedelijk
□ sterk stedelijk
□ weinig stedelijk
□ zeer sterk stedelijk

AFSTAND TOT CAFÉ (KM)
0.4 – 6.2

AFSTAND TOT ATTRACTIE (KM)
1.6 – 46.5

BEDRIJVEN
120 – 106055

BEVOLKINGSDICHTHEID (AANTAL II

Figure 3. Screenshot of FacetCheck running of the Dutch neighborhood dataset.

about depopulated areas, possibly related to economic and/or social trends in those areas. Together with the data journalists, several Data Stories were created, and FacetCheck was used in order to find interesting filter criteria for identifying depopulation areas.
The data story includes multiple queries that show various characteristics of depopulation areas, such as the average distance to public transport, car ownership, and access to jobs (Figure 2).

By pressing the orange “Show Query” button (Figure 2), the user can verify the query, and with a bit of SPARQL knowledge, the query can also be adapted. For instance, the specific depopulation areas the query retrieves can be changed with a small edit. Now we focus on the job market participation rate in shrink areas. The query results in Figure 3 show that access to work in the northern depopulation areas is below the national average.

By looking at the results, and zooming in, we learn that only 2 out of 9 depopulation areas have lower than average employment rates. Additionally, we see that there are areas without depopulation (such as Rotterdam and The Hague) that have have lower employment rates. In combination with the depopulation Data Story, the FacetCheck browser was used in order to filter for depopulation areas based on various filters. In Figure 3, the left hand side of the screen shows the filters that are based on the properties in the dataset. By scrolling, over 100 data properties can be selected through a map, a slider or a checkbox list. The right hand side shows widgets for 4 of the currently selected neighbourhoods.

6. CONCLUSION

This paper has introduced two new viewing components for Linked Open Data: Data Stories and FacetCheck. Both have proven valuable for the Land Registry and Mapping Agency (Kadaster) and the National Statistics Office (CBS). The development of both components is currently being finalized in the context of the Kadaster Data Platform.

Over the last couple of years, Linked Open Data has seen a relatively slow adoption speed which may in part be due to the lack of sufficiently distinctive and functional browsing and viewing facilities. We hope that both Data Stories and FacetCheck will be picked up by other organizations that are setting up Linked Data platforms, specifically other governmental organizations that publishing geospatial data assets.

7. REFERENCES

Beek, W., & Folmer, E. (2017). An Integrated Approach for Linked Data Browsing. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* (Vol. 42), pp. 35-38.

Beek, W., Folmer, E., Rietveld, L., & Walker, J. (2017). GeoYASGUI: The GeoSPARQL Query Editor and Result Set Visualizer. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* (Vol. 42), pp. 39-42.