The Semantic Web in Earth and Space Science.
Current Status and Future Directions
T. Narock and P. Fox (Eds.)

Chapter 4

Managing Geospatial Linked Data in the GeoKnow Project

Jens LEHMANN[a,1], Spiros ATHANASIOU[b], Andreas BOTH[d], Alejandra GARCIA ROJAS[c], Giorgos GIANNOPOULOS[b], Daniel HLADKY[c], Jon Jay LE GRANGE[c], Axel-Cyrille NGONGA NGOMO[a], Mohamed Ahmed SHERIF[a], Claus STADLER[a], Matthias WAUER[d], Patrick WESTPHAL[a], Vadim ZASLAWSKI[c]
[a] *Institute of Applied Informatics, University of Leipzig, Germany*
[b] *Athena Research and Innovation Center, Greece*
[c] *Ontos AG, Switzerland*
[d] *Unister GmbH, Germany*

Abstract. Within the GeoKnow project, various tools are developed and integrated which aim to simplify managing geospatial Linked Data on the web. In this article, we summarize the state of the art and describe the status of open geospatial data on the web. We continue by presenting the Linked Data Stack as technical underpinning of GeoKnow and give a first presentation of the platform providing a light-weight integration of those tools.

1. Introduction

In recent years, Semantic Web methodologies and technologies have strengthened their position in the areas of data and knowledge management. Standards for organizing and querying semantic information, such as RDF(S) and SPARQL have been adopted by large academic communities, while corporate vendors adopt semantic technologies to organize, expose, exchange and retrieve their data as Linked Data. RDF stores have become robust enough to support volumes of billions of records (RDF triples), and also offer data management and querying functionalities very similar to those of traditional relational database systems.

Among the existing knowledge bases, those with geospatial data are among the largest in existence and have high importance in a variety of everyday applications. The data can be mapped and often manipulated with Geographic Information Systems

[1] Corresponding Author: Jens Lehmann; lehmann@informatik.uni-leipzig.de

(GIS), however the integration of external data sets into these systems is time-consuming and complex. The aim of the GeoKnow project is to provide the necessary tools and methods to easily integrate and process data across a wide range of data sources on the web of data. Producing and updating geospatial data is expensive and resource intensive. Hence, it becomes crucial to be able to integrate, repurpose and extract added value from geospatial data to support decision-making and management of local, national and global resources. Spatial Data Infrastructures (SDIs) and the standardization efforts from the Open Geospatial Consortium (OGC) serve this goal, enabling geospatial data sharing, integration and reuse among Geographic Information Systems (GIS). Geospatial data are now, more than ever, truly syntactically interoperable. However, they remain largely isolated in the GIS realm and thus absent from the Web of Data. Linked data technologies enabling semantic interoperability, interlinking, querying, reasoning, aggregation, fusion, and visualization of geospatial data are only slowly emerging. The vision of GeoKnow is to leverage geospatial data as first-class citizens in the Web of Data, in proportion to their significance for the data economy.

The remainder of the article is structured as follows: In Section 2, we describe the status of open geospatial data on the web, followed by semantic web technologies for geospatial web data specifically in Section 3. After that, we give a general overview of the GeoKnow project in 4. A central vision behind the GeoKnow project is the Linked Data Life-Cycle described in Section 5. The technical realization of this vision is done in the Linked Data Stack (Section 6), which consists of a variety of components (Section 7). Those components are integrated in an interface, which we call the GeoKnow Generator (Section 8). An application scenario using the GeoKnow Generator and Linked Data Stack is presented in Section 9. Related work is presented in Section 10 and we finally conclude in Section 11.

2. Open Geospatial Data on the Web

Currently, there are three major sources of open geospatial data in the Web: Spatial Data Infrastructures, open data catalogues, and crowdsourcing initiatives.

Spatial Data Infrastructures (SDIs) were created to promote the discovery, acquisition, exploitation and sharing of geographic information. They include technological and organizational structures, policies and standards that enable efficient discovery, transfer and use of geospatial data using the web [37]. Research and development in this field is closely tied to standardization activities led by international bodies, namely the ISO/TC 211[2], OGC[3] and W3C[4]. In Europe, the INSPIRE Directive[5] follows the OGC open standards, and has defined common data models for a number of application domains, such as hydrography, protected sites and administrative units, to enhance interoperability of spatial data sets of the different European countries. It provides the legal and technical foundations to ensure member state SDIs are compatible and usable on a trans-boundary context. The major open standard Web

[2] http://www.isotc211.org/
[3] http://www.ogc.org
[4] http://www.w3.org/
[5] http://inspire.jrc.ec.europa.eu/

services regarding discovery and querying of geospatial data in SDIs are OGCs Catalogue Service and Web Feature Service respectively. The first allows the discovery of geospatial data based on their metadata (e.g. scale, coverage) and the second enables querying of geospatial data. Additional standards provide access to maps and tiles (Web Map Service, Web Tile Service) and enable developers to programmatically invoke and compose complex geospatial analysis services (Web Processing Service). Currently practically all GIS and geospatial databases are fully compatible with these standards; GIS users can consume geospatial data from SDIs and publish geospatial data to SDIs with a few clicks. On a practical level, it is clear that SDIs must be considered as diachronic and stable data infrastructures. They represent a significant investment from the public and private worldwide and are the basis for interoperability among significant scientific domains. Further, they constitute the most prominent source for high-quality open geospatial data. Thus, any contribution and advancement must either be directly involved in standardization efforts, or be based solely on existing standards, without directly affecting their applications.

Open data catalogues provide open geospatial data by a) encapsulating existing SDIs and/or b) ad hoc publishing available geospatial data. In the latter case, geospatial data are published as regular open data. The only difference regards the use of file formats of the geospatial domain (e.g. shapefile, GML) and availability of data for specific coordinate reference systems (typically national CRS). In the former case, an available national/regional SDI is exploited as a source for harvesting its geospatial data. The Catalogue Service is used to discover available data, and their metadata are added in the open data catalogue for homogenized data discovery.

The actual data are available as exported file snapshots in common geospatial formats as before, or through the query services provided by the SDI. Consequently, open data catalogues typically offer geospatial data as files and at best expose any available SDI services for data access.

Crowdsourced geospatial data are emerging as a potentially valuable source of geospatial knowledge. Among various efforts we highlight OpenStreetMap, GeoNames, and Wikipedia as the most significant. GeoNames[6] provides some basic geographical data such as latitude, longitude, elevation, population, administrative subdivision and postal codes. This data is available as text files and also accessible through a variety of web services such as free text search, find nearby or even elevation data services. OpenStreetMap[7] (OSM), a community initiative for crowdsourced production of open global maps, has emerged as a significant platform for creating, sharing, mapping, browsing and visualizing geospatial data on the Web. OSM includes geospatial data of various types (e.g. roads, public transit) daily increase in coverage, accuracy and quality. Data are integrated from public and private sector sources (e.g. transit authorities). Further, easy to use tools, straightforward publishing workflows, and support from the industry, have created a sustainable pathway for establishing OSM as the leading source of open geospatial data in the Web.

[6] http://www.geonames.org/
[7] http://www.openstreetmap.org/

3. Semantic Web Technologies for Geospatial Data

The benefits of semantic technology for spatial data management are explored in a number of topics. For example, ontologies have been used in the form of taxonomies on thematic web portals (e.g. habitat or species taxonomies, categories of environmentally sensitive areas, or hierarchical land use classifications). The role of these ontologies is however limited. They provide background knowledge, but only in some experimental prototypes they are used for constructing search requests or for grouping of search results into meaningful categories. Further, in experimental settings, there are examples of using OWL for bridging differences in conceptual schemas, e.g. [12]. The role of ontologies and knowledge engineering in these prototypes is basically to provide methodologies for integration and querying [64, 8]. Ontologies have played an important role in structuring data of geospatial domains [1, 23]. However, semantic technology has not influenced spatial data management yet, and mainstream GIS tools are not yet extended with semantic integration functionality.

3.1. Standardization Efforts

Early work includes the Basic Geo Vocabulary[8] by the W3C, which provides a namespace for representing lat(itude), long(itude) and other information about spatially-located entities, using the WGS84 CRS as its standard reference datum. This vocabulary explored the possibilities of representing mapping/location data in RDF, so it was not intended to address all issues covered by OGC. Instead, it was meant to provide just a few basic terms that can be used in RDF (e.g., RSS 1.0 or FOAF documents) so as to describe latitudes and longitudes. The motivation for using RDF as a carrier for lat/long information is RDF's capability for cross-domain data mixing.

GeoRSS[9] has been designed as a lightweight, community-driven way to extend existing RSS feeds with geographic information, thus providing an interoperable manner to enable processing, aggregation, sharing and mapping of geographically tagged feeds. Two encodings of GeoRSS are available. GeoRSS-Simple is a very lightweight format that can be easily added to existing feeds. It supports basic geometries (point, line, box, polygon) and covers the typical use cases when encoding locations. GeoRSS GML is a formal GML Application Profile, and supports a greater range of features, notably coordinate reference systems other than WGS84 latitude/longitude.

GeoOWL[10] provides an ontology, which closely matches the GeoRSS feature model and utilizes the existing GeoRSS vocabulary for geographic properties and classes. Fragments of GeoRSS XML within RSS 1.0 or Atom, which conform to the GeoRSS specification, will also conform to the Geo OWL ontology. Thus, the ontology provides a compatible extension of GeoRSS for use in more general RDF contexts. Furthermore, topological modelling of geometric shapes in RDF is supported by the NeoGeo Geometry Ontology[11]. NeoGeo is a still incomplete attempt to establish a vocabulary for describing geographical regions in RDF. It aims to support typical geometric objects as well as WKT serialization. However, both GeoOWL and NeoGeo

[8] http://www.w3.org/2003/01/geo/
[9] http://georss.org/Main_Page
[10] http://www.w3.org/2005/Incubator/geo/XGR-geo-20071023/W3C_XGR_Geo_ files/geo_2007.owl
[11] http://geovocab.org/

ontologies only supported the WSGS84 CRS (thus leading to gross errors in other CRSs), and offered limited support for geospatial operations required in real world GIS workloads. GeoJSON[12] is a geospatial data interchange format based on JavaScript Object Notation (JSON). A GeoJSON object may represent a geometry, a feature, or a collection of features. Features in GeoJSON contain a geometry object and additional properties, and a feature collection represents a list of features.

GeoRDF was intended as an RDF compatible profile[13] for geographic information (points, lines and polygons). Vocabularies like RDFGeom, and its 2d companion, RDFGeom2d, provide an RDF framework that is extensible via subclassing to all kinds of geometric data, although the class hierarchy is currently only sparsely populated.

The class hierarchy is loosely based on the geometric part of SVG. Since lines, curves, and transformations are geometric, and not specifically geographical notions, RDFGeom and RDFGeom2d formulate geometry without reference to any particular application. Proper-ties that connect geometry to intended interpretations are asserted by application-specific vocabularies.

3.2. GeoSPARQL

GeoSPARQL [20] has emerged as a promising standard from W3C for geospatial RDF, with the aim of standardizing geospatial RDF data insertion and query. Standardization of GeoSPARQL is among the goals of OGC in order to ensure a consistent representation of geospatial semantic data across the Web, thus allowing to both vendors and users of data and applications to achieve uniform access to geospatial RDF data. GeoSPARQL provides various conformance classes concerning its implementation of advanced reasoning capabilities, as well as several sets of terminology for topological relationships between geometries. Therefore, different implementations of the GeoSPARQL specification are possible, depending on the respective domain/application. In addition, GeoSPARQL closely follows existing standards from OGC for geospatial data, to facilitate spatial indexing from relational databases. GeoSPARQL defines a small, but concrete ontology for representing features and geometries, as well as a set of SPARQL query predicates and functions, all according to spatial OGC standards. In order to cope with diverse and incompatible methods for representing and querying spatial data, GeoSPARQL follows the existing OGC standards concerning spatial indexing in relational databases. Hence, spatial ontologies can be combined, indexed and queried along with other proprietary ontologies from data providers. Equally important, interoperability among compliant triple stores is achieved, so spatial RDF data can be commonly accessed and exchanged. GeoSPARQL is designed to accommodate systems based on qualitative spatial reasoning and systems based on quantitative spatial computations. Systems based on qualitative spatial reasoning, (e.g. those based on the Region Connection Calculus [51] do not usually model explicit geometries, so the geometries are either unknown or cannot be made concrete. Thus, queries in such systems will likely test for binary spatial relationships between features rather than between explicit geometries. A quantitative spatial reasoning system involves concrete geometries for features, so

[12] http://geojson.org/
[13] http://www.w3.org/wiki/GeoRDF

distances, areas and topological relations can be explicitly calculated. To allow queries for spatial relations between features in quantitative systems, GeoSPARQL defines a series of query transformation rules that expand a feature-only query into a geometry-based query. With these transformation rules, queries about spatial relations between features will have the same specification in both qualitative systems and quantitative systems. The qualitative system will likely evaluate the query with a backward-chaining spatial reasoner, and the quantitative system can transform the query into a geometry-based query that can be evaluated with computational geometry. With a common set of topological relations, GeoSPARQL allows conclusions from quantitative applications to be used by qualitative systems and a single query language for both types of reasoning. Future extensions are oriented towards definitions of new conformance classes for other standard serializations of geometry data (e.g. KML, GeoJSON). Developing vocabularies for spatial data, and expanding the GeoSPARQL vocabularies with OWL axioms to aid in logical spatial reasoning is also considered as a valuable contribution. Standard processes could also be developed for converting to RDF and exposing large amounts of existing feature data represented either in GML-like formats or in a data store supporting the general feature model [20].

3.3. Research Efforts

There has been a growing research interest towards representing and querying geospatial data in RDF. An extension to SPARQL, termed SPARQL-ST [50] proposed a modified SPARQL syntax for specifying spatial queries against data modelled in a GeoRSS-like ontology. This dialect supported data in different spatial reference systems, something missing from many vocabularies such as GeoOWL and the Basic Geo vocabularies. It also included support for temporal and thematic features. But the proposed query syntax deviates from the standard SPARQL, whereas exposed data cannot be accessed from third-party systems that do not use SPARQL-ST.

The issue of adding topological predicates to SPARQL was also examined in [63]. The proposed ontology takes advantage of OGC Simple Features [19] in order to provide a basic set of geometries and relations. However, relations have to be specifically encoded in RDF whereas there is no support for multiple CRS in the data. A hierarchical approach is proposed in [18], using a meta-level for abstract space-time knowledge, a schema-level for well-known models in spatial and temporal reasoning (e.g., RCC), and an instantiation-level for mappings and formal descriptions. This model refers to various spatiotemporal statements in the Linked Data clouds and nicely abstracts spatial knowledge from its underlying representation. However, mappings must be defined for each dataset at the instantiations level. A research prototype was presented in [7] that supports a native RDF triple store implementation with deeply integrated spatial query functionality. Spatial features in RDF are modelled as literals of a complex geometry type so spatial predicates can be expressed as SPARQL filter functions on this type. This makes it possible to use W3C´s standardized SPARQL query language as-is, i.e., without any modifications or extensions for spatial queries. It is noteworthy that OGC Simple Features relations are used as the background for posing queries with spatial predicates. In parallel to research on geospatial support, Semantic Web technologies have also provided a great deal of schema flexibility useful in analyzing and integrating poorly structured data, e.g., web- or community-based data, such as map data from the OpenStreetMap project [3]. This LinkedGeoData set

offers a spatial knowledge base, derived from OpenStreetMap[14] and is interlinked with DBpedia[15], GeoNames[16] and other datasets as well as integrated with icons and multilingual class labels from various sources. It contains over many million triples describing the nodes and paths from OpenStreetMap. The LinkedGeoData set is accessible through SPARQL endpoints running on Virtuoso platform[17], as well as via a REST interface in its most recent release [53].

4. The GeoKnow Project

GeoKnow is an EU research project running for three years from December 2012 to November 2015. It is motivated by previous work in the LinkedGeoData [53] project (LGD), which makes OpenStreetMap data available as an RDF knowledge base. As a result, OSM data were introduced in the LOD cloud and interlinked with GeoNames, DBpedia [32, 35], and multiple other data sources. LGD intended to simplify information creation and aggregation related to spatial features. During this exercise, several research challenges were found such as scalability with spatial data, query performance, spatial data modeling, flexible transformation of special data, as well as data operations such as routing data. It was realized that geospatial data, especially scientific data, available on the web can open new opportunities to improve management and decision making applications.

Consequently, the vision of the project is to make geospatial data more easily accessible on the web and improve its publishing, querying, interlinking and quality assessment based on the Linked Data principles and the Linked Data Life-Cycle vision [2]. This will facilitate the development of applications and backend functionality or enable answering questions that were not possible with isolated geospatial data. This change is also a step towards the discoverability of data that share geospatial features (i.e. supported by querying and reasoning), and a boosting for the geospatial data integration through geospatial data merging and fusing tools. The project applies the RDF model and the GeoSPARQL standard as the basis for representing and querying geospatial data. In particular, GeoKnow contributions are in the following areas:

Efficient geospatial RDF querying. Existing RDF stores lack performance and geospatial analysis capabilities compared to geospatially-enabled relational DBMS. We introduce query optimization techniques for accelerating geospatial querying significantly.

Fusion and aggregation of geospatial RDF data. Given a number of different RDF geospatial data for a given region containing similar knowledge (e.g. OSM, PSI and closed data[18]) we devise semi-automatic fusion and aggregation techniques in order to consolidate them and provide a data set of increased value and quantitative quality metrics of this new data resource

[14] http://www.openstreetmap.org/
[15] http://dbpedia.org/About
[16] http://www.geonames.org/
[17] http://virtuoso.openlinksw.com/
[18] http://ec.europa.eu/information_society/policy/psi/indexen.htm

Visualization and authoring. We develop reusable mapping components, enabling the integration of geospatial RDF data as an addition data resource in web map publishing. Further, we enable the light-weight creation of simple geospatial applications by shifting the complexity of development to data integration and modeling.

GeoKnow Generator. The GeoKnow Generator consists of a full suite of tools supporting the complete life-cycle of geospatial linked open data. The GeoKnow Genera-tor enables publishers to triplify geospatial data, interlink them with other geospatial and non-geospatial Linked Data sources, fuse and aggregate linked geospatial data to provide new data of increased quality, visualize and author linked geospatial data in the Web

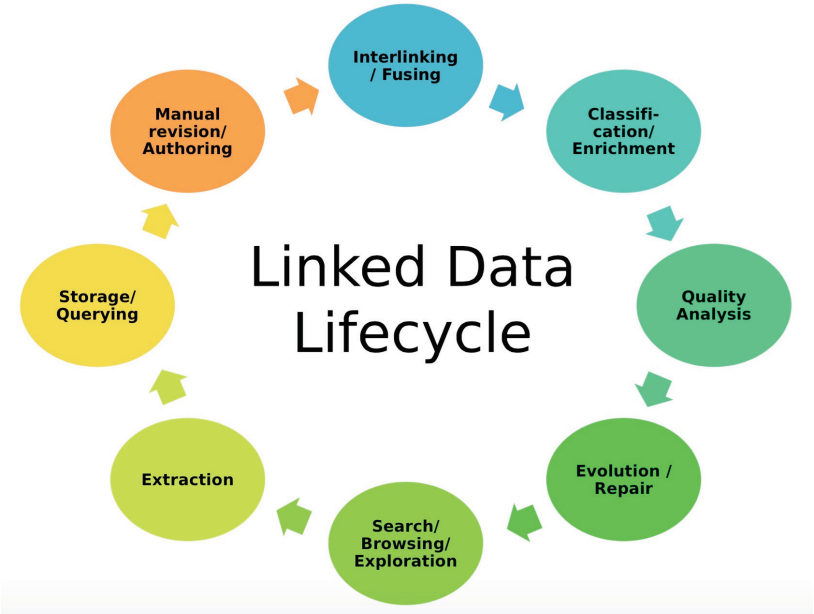

Figure 1. Stages of the Linked Data life-cycle supported by the Linked Data Stack.

5. The Linked Data Life-Cycle

The different stages of the Linked Data life-cycle (depicted in Figure 1) include:

Storage. RDF Data Management is still more challenging than relational Data Management. This is especially true for the large volume of geo-spatial data available on the Data Web. We aim to close this performance gap by employing column-store technology, dynamic query optimization, adaptive caching of joins, optimized graph processing and cluster/cloud scalability. Moreover, we aim to provide techniques that allow SPARQL to SQL mapping with the aim of making the results of decades of research on geo-information systems available to the Data Web community.

Authoring. In GeoKnow, we aim to facilitate the authoring of rich semantic knowledge bases by leveraging Semantic Wiki technology, the WYSIWYM paradigm (What You See Is What You Mean) and distributed social, semantic collaboration and networking techniques. Moreover, we aim to provide frameworks that allow for the time-efficient development of applications based on geo-spatial data.

Interlinking. Creating and maintaining links in a (semi-)automated fashion is still a major challenge and crucial for establishing coherence and facilitating data integration. We seek linking approaches yielding high precision and recall, which scale to large knowledge bases. These approaches need to support geo-spatial features such as vector geometry. In addition, we aim to devise approaches for efficient memory management through link discovery approaches, which can configure themselves automatically or with end-user feedback.

Classification. Linked Data on the Web is mainly raw instance data. For data integration, fusion, search and many other applications, however, we need this raw instance data to be linked and integrated with upper level ontologies.

Quality. The quality of content on the Data Web varies, as the quality of content on the document web varies. We aim to develop techniques for assessing the quality of RDF data based on characteristics such as provenance, context, coverage or structure.

Evolution/Repair. Data on the Web is dynamic. We need to facilitate the evolution of data while keeping things stable. Changes and modifications to knowledge bases, vocabularies and ontologies should be transparent and observable. We also develop methods to spot problems in knowledge bases and to automatically suggest repair strategies.

Search/Browsing/Exploration. For many users, the Data Web is still invisible below the surface. We develop search, browsing, exploration and visualization techniques for different kinds of Linked Data (i.e. spatial, temporal, statistical), which make the Data Web sensible for real users.

These life-cycle stages, however, should not be tackled in isolation, but by investigating methods, which facilitate a mutual fertilization of approaches developed to solve these challenges. Examples for such mutual fertilization between approaches include:

- Ontology matching and instance matching, as the detection of adequate class mappings to facilitate the correct detection of links across knowledge bases and vice-versa.
- Ontology schema mismatches between knowledge bases can be compensated for by learning which concepts of one are equivalent to which concepts of another knowledge base.
- Feedback and input from end users (e.g. regarding instance or schema level map-pings) can be taken as training input (i.e. as positive or negative examples) for machine learning techniques in order to perform inductive reasoning on larger knowledge bases, whose results can again be assessed by end users for iterative refinement.

- Semantically enriched knowledge bases improve the detection of inconsistencies and modeling problems, which in turn results in benefits for interlinking, fusion, and classification.
- The querying performance of RDF data management directly affects all other components, and the nature of queries issued by the components affects RDF data management.

As a result of such interdependence, we should pursue the establishment of an improvement cycle for knowledge bases on the Data Web. The improvement of a knowledge base with regard to one aspect (e.g. a new alignment with another interlinking hub) triggers a number of possible further improvements (e.g. additional instance matches).

The challenge is to develop techniques, which allow exploitation of these mutual fertilizations in the distributed medium Web of Data. One possibility is that various algorithms make use of shared vocabularies for publishing results of mapping, merging, repair, or enrichment steps. After one service published its new findings in one of these commonly understood vocabularies, notification mechanisms (such as Semantic Ping-back [56]) can notify relevant other services (which subscribed to updates for this particular data domain), or the original data publisher, that new improvement suggestions are available. Given proper management of provenance information, improvement suggestions can later (after acceptance by the publisher) become part of the original dataset.

6. The Linked Data Stack

The Linked Data Stack serves two main purposes. Firstly, the aim is to ease the distribution and installation of tools and software components that support the Linked Data publication cycle. As a distribution platform, we have chosen the well-established Debian packaging format. The second aim is to smooth the information flow between the different components to enhance the end-user experience by a more harmonized look-and-feel.

6.1. Deployment Management Leveraging Debian Packaging

In the Debian package management system [36], software is distributed in architecture-specific binary packages and architecture-independent source code packages. A Debian software package comprises two types of content: (1) control information (incl. meta-data) of that package, and (2) the software itself.

The control information of a Debian package will be indexed and merged together with all other control information from other packages available for the system. This information consists of descriptions and attributes for:

(a) The software itself (e.g. licenses, repository links, name, tagline, ...),

(b) Its relation to other packages (dependencies and recommendations),

(c) The authors of the software (name, email, home pages), and

(d) The deployment process (where to install, pre and post install instructions).

The most important part of this control information is its relations to other software. This allows the deployment of a complete stack of software with one action. The following dependency relations are commonly used in the control information:

Depends: This declares an absolute dependency. A package will not be configured unless all of the packages listed in its Depends field have been correctly configured. The Depends field should be used if the depended-on package is required for the depending package to provide a significant amount of functionality. The Depends field should also be used if the install instructions require the package to be present in order to run.

Recommends: This declares a strong, but not absolute, dependency. The Recommends field should list packages that would be found together with this one in all but unusual installations.

Suggests: This is used to declare that one package may be more useful with one or more others. Using this field tells the packaging system and the user that the listed packages are related to this one and can perhaps enhance its usefulness, but that installing this one without them is perfectly reasonable.

Enhances: This field is similar to Suggests but works in the opposite direction. It is used to declare that a package can enhance the functionality of another package.

Conflicts: When one binary package declares a conflict with another using a Conflicts field, dpkg will refuse to allow them to be installed on the system at the same time. If one package is to be installed, the other must be removed first.

All of these relations may restrict their applicability to particular versions of each named package (the relations allowed are <<, <=, =, >= and >>). This is useful in forcing the upgrade of a complete software stack. In addition to this, dependency relations can be set to a list of alternative packages. In such a case, if any one of the alternative packages is installed, that part of the dependency is considered to be satisfied. This is useful if the software depends on a specific functionality on the system instead of a concrete package (e.g. a mail server or a web server). Another use case of alternative lists are meta-packages. A meta-package is a package, which does not contain any files or data to be installed. Instead, it has dependencies on other (lists of) packages.

Installing the Linked Data Stack - The Linked Data Stack is available at http://stack.linkeddata.org/. Our reference OS is Ubuntu 12.04 LTS. Most of the components run on old or more recent releases without a problem. In general, deploying the Linked Data Stack software or parts of it is simple. There are only two steps to execute in order to install Linked Data Stack software: (1) Add the Linked Data Stack package repository to the system's repository list and update the repository index. (2) Install desired software packages by using a graphical or text-based package management application. The procedure can be executed using graphical front-ends like Synaptic[19]. Using the command line the Linked Data Stack installation is performed as follows[20]:

[19] http://www.nongnu.org/synaptic/

[20] More information, tutorials and FAQs at

```
# download the repository package
wget http://stack.linkeddata.org/download/lds-repo.deb

# install the repository package
sudo dpkg -i lds-repo.deb

# update the repository database
sudo apt-get update
```

This action will download, install and update the repository package. The actual list of components available to install can be found in the LDStack website[21].

GeoKnow has also contributed to the Stack by providing a web-based application: the GeoKnow Generator that integrates some of the components of the stack. The Generator can be installed with the command line as follows:

```
# install GeoKnow Generator
# with dependent components from Linked Data Stack
sudo apt-get geoknow-generator-ui
```

6.2. Data integration based on SPARQL, Authentication and Provenance

The basic architecture of a local installation of Linked Data Stack including GeoKnow Generator is depicted in Figure 2. All components in the Linked Data Stack act upon RDF data and are able to communicate via SPARQL with the central system-wide RDF quad store (i.e. SPARQL backend). This quad store (Openlink Virtuoso) manages user graphs (knowledge bases) as well as a set of specific system graphs where the behavior and status of the overall system is described. The following system graphs are currently used:

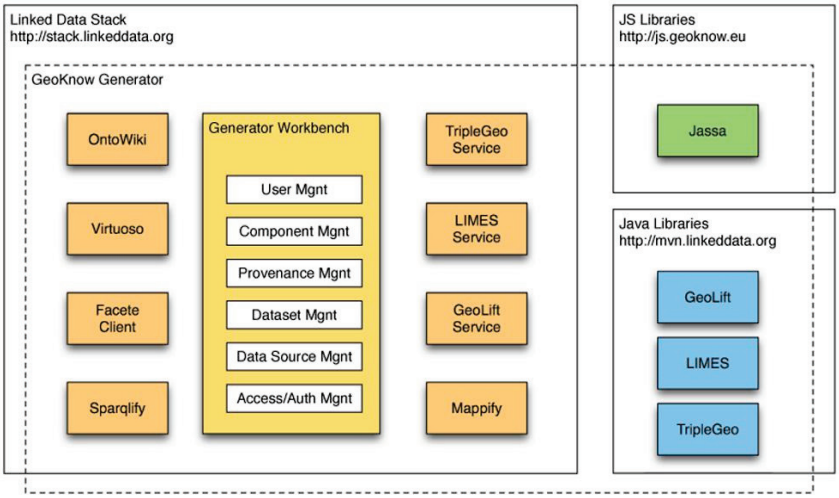

Figure 2. Basic architecture of GeoKnow Generator within Linked Data Stack.

http://stack.linkeddata.org/documentation/

[21] http://stack.linkeddata.org/download/repo.php

Package Graph: In addition to the standard Debian package content, each Linked Data Stack package consists of a RDF package info which contains:

(a) The basic package description, e.g. labels, dates, maintainer info (this is basically DOAP data and redundant to the classic Debian control file);
(b) Pointers to the place where the application is available;
(c) A list of capabilities of the packed software (e.g. resource linking, RDB extraction). These capabilities are part of a controlled vocabulary. The terms are used as pointers for provenance logging and access control definition.

Upon installation, the package info is automatically added to the package graph to allow to query, which applications are available and what is the user able to do with them.

Access Control Graph: This system graph implements a simple authentication and a graph level authorization. It describes which users are able to use which capabilities and have access to which graphs. The default state of this graph contains no restrictions, but could be used to restrict certain authorization control to specific capabilities.

Provenance Graph: Each software package is able to log system wide provenance information to reflect the evolution of a certain knowledge base. Different ontologies are developed for that use-case. To keep the context of the Linked Data Stack, we use the controlled capability vocabulary as reference points.

In addition to the SPARQL protocol endpoint, application packages can use a set of APIs, which allow queries and manipulation currently not available with SPARQL alone (e.g. fetching graph information and manipulating namespaces). The Debian system in-staller application automatically adds and removes package descriptions during install / upgrade and remove operations. All other packages are able to use the APIs as well as to create, update and delete knowledge bases.

7. GeoKnow Linked Data Stack Components

Table 1 shows the current Linked Data Stack components in alphabetic order. In the following, we give a brief summary on some of the most important packages.

Table 1. Overview on GeoKnow Linked Data Stack components.

Tool	Category	Supported Stages
DL-Learner [28,29,31]	Machine Learning in OWL	Enrichment
Facete [54]	Faceted browser for spatial data	Browsing, Exploration
GeoLift	Enrichment with geo-spatial enrichment	Enrichment
LIMES [44, 39,40,42,43]	Linking Workbensch	Interlinking
Mappify	Map view generator	Browsing, Exploration
OntoWiki [15]	Generic Data Wiki	Authoring, Exploration
ORE [30]	Knowledge Base Debugging	Repair
R2RLint	RDB2RDF quality assessment	Quality Analysis

RDFauthor [57]	RDFa authoring	Authoring
RDFUnit [25]	Quality assessment tool	Quality Analysis
Sparqlify	RDB2RDF Mapping	Extraction
TripleGeo	Geo-spatial feature extraction	Extraction
Virtuoso	Hybrid RDBMS/Graph Column Store	Storage/Querying

7.1. Extraction and Loading

Nowadays, data are available in diverse forms ranging from raw text files to databases dumps. Moreover, for each data type there is a special format. For example, geospatial data are normally available in the form of shape files. Given such heterogeneity of raw data forms and formats, the RDF extraction and loading is a challenging task. In the following we present utilities and tools used for extraction of RDF data from different raw data forms and formats.

TripleGeo: TripleGeo is a utility developed by the Institute for the Management of Information Systems at Athena Research Center. This generic purpose, open-source tool can be used for integrating features from geospatial databases into RDF triples.

Sparqlify: Sparqlify is a SPARQL-to-SQL rewriter that enables one to define RDF views on relational databases and query them with SPARQL. It is currently in beta state and features basic support for spatial data types. Sparqlify powers the Linked Data interface and a SPARQL endpoint of the LinkedGeoData Server, where access to billions of virtual triples from the OpenStreetMap database is provided.

7.2. Querying and Exploration

Virtuoso: Virtuoso is a scalable high-performance RDF Quad Store available in open source and commercial forms, providing the core geo spatial knowledge storage for the GeoKnow generator. Virtuoso provides optimized distributed SPARQL and SQL query processing for heterogeneous data integration across data sources.

Virtuoso implements a quad GSPO, composed of Graph, Subject, Predicate, and Object. All quads are stored in one table, where GSP values are IRIs, whereas O is any SQL serializable object. SPARQL is embedded and translated into SQL for querying RDF data stored in Virtuoso's database. The default RDF index scheme consists of two full indices over RDF quads plus three partial indices. This index scheme is generally adapted to all kinds of workloads, regardless of whether queries generally specify a graph. Spatial indexing is performed with a 2-dimensional R-tree implementation. Currently, Virtuoso fully supports handling of points, while full support of operations on more complex geometries (e.g. lines, polygons) is under development. With respect to geospatial queries, the following topological operations are supported ST_intersects, ST_contains and ST_within. A handful of geometric functions are also available (e.g. ST_distance, ST_x, ST_y, ST_AsText).

At its latest version 7.1, Virtuoso delivers massively scalable hybrid (RDF Graph & SQL) DBMS for secure, high-performance, Big Data management and integration. Columnar storage strongly reduces the RAM needs of RDF workloads, which benefits performance and cost of deployment. Vectorized execution boosts performance of RDF-oriented index-lookup joins and shared-nothing cluster communications. Using column store techniques boosts space efficiency and query performance for RDF-based Linked Data deployment and management.

7.3. Authoring

OntoWiki: OntoWiki is an application that facilitates the visual presentation of a knowledge base as an information map, with different views on instance data. It enables intuitive authoring of semantic content, with an inline editing mode for editing RDF content, similar to WYSIWIG for text documents.

RDFauthor: RDFauthor is a JavaScript library that provides web developers with a set of 10+ widgets for editing RDF data. This library powers the edit features in OntoWiki. The widget collection includes, among others, a markdown editor; time, date and image pickers; a map widget with geocoding and reverse-geocoding support; and an RDF term editor. RDFa and SPARQL data sources are supported. RDFauthor is capable of computing changes between the state of the data before and after an edit. In the case of SPARQL data sources, RDFauthor can directly apply changes at the source using SPARQL 1.1/Up-date queries. Alternatively, custom strategies for processing data modifications can be plugged in.

7.4. Linking

LIMES: LIMES is a link discovery framework for the Web of Data, which addresses both the time complexity of link discovery and the complexity of discovering accurate link specifications [39]. To address the time complexity problem, LIMES implements a hybrid approach, which uses set semantics to combine the output of manifold measure-specific algorithms [38, 40]. The second problem is addresses by novel unsupervised as well as supervised (batch and active) approaches to learning link specifications [41, 42].

7.5. Enrichment

RDF data normally contain implicit references to other data types, and the geographic data in not an exception. For example, music datasets such as musicbrainz[22] include references to locations of record labels, places where artists were born or have been, etc. The aim of the spatial enrichment process is to retrieve such information and make it explicit. In the following, we present frameworks used for to enrich the spatial dimension of input datasets.

Geolift: GeoLift is a spatial mapping component, which aims to enrich RDF datasets with geospatial information. To achieve this goal, GeoLift relies on three atomic modules based on dereferencing, linking and Named Entity Recognition and Disambiguation. The dereferencing module enriches the geo-spatial datasets by finding the URI objects in the source dataset and dereferencing the URIs in order to add related geographical information, e.g. geo:lat if it exists. The linking module adds valuable geospatial information to the datasets through linking the dataset that is targeted to be enriched with another dataset. This produces geo-spatial relationships that are associated to the linked URIs in the enriched dataset. The Named Entity Recognition and Disambiguation is based on the FOX and AGDISTIS frameworks, which find person, location and organization mentions in text and ground them in DBpedia.

[22] http://wiki.musicbrainz.org/LinkedBrainz

DL-Learner: The DL-Learner framework provides a set of (semi-)supervised machine learning algorithms (see e.g. [31]) for knowledge bases, specifically for OWL ontologies and SPARQL endpoints. The goal of DL-Learner is to support knowledge engineers in constructing knowledge and learning about the data they created, by generating axioms and concept descriptions, which fit the underlying data. DL-Learner is used in the backend of the ORE and RDFUnit tools.

7.6. Quality Analysis

Besides the authoring, linking and enrichment of Linked Data, regarding quality related issues is also an important task. The consideration and study of data quality has been performed in many different domains within the last decades, which led to generally accepted perceptions, like viewing data quality as a multi-dimensional concept expressing 'fitness for use'. This research is currently continued for Linked Data also covering geospatial information. Some of the corresponding tools are introduced in the following.

RDFUnit: The RDFUnit framework is a software tool, implementing the test-driven data quality methodology [26]. The main idea of this approach is to provide general SPARQL query patterns to discover common data quality issues. These patterns can then be instantiated for a given dataset or vocabulary, which leads to executable and reusable test cases. Such instantiations can be performed manually by a domain expert or generated automatically considering the schema information of the vocabulary. Besides the core framework, there is also a web front end[23] allowing assess to arbitrary SPARQL endpoints.

R2RLint: Since relational databases are an important source considered in the extraction phase, the RDB2RDF approach should also be assessed with regards to the quality of the RDB2RDF mappings and the generated RDF data. R2RLint is a framework, currently developed for RDB2RDF quality assessment taking the specifics of the underlying relational database and its transformation process to RDF into consideration. R2RLint is a command line tool, designed to be easily extendible with own metrics.

7.7. Repair

ORE: ORE [30] (Ontology Repair and Enrichment) allows knowledge engineers to improve an OWL ontology or SPARQL endpoint backed knowledge base by fixing logical errors and making suggestions for adding further axioms to it. ORE uses state-of-the-art methods to detect errors and highlight the most likely sources for the problems. To harmonize schema and data in the knowledge base, algorithms of the DL-Learner framework are integrated.

7.8. Browsing and Exploration

To provide usable and intuitive interfaces for the visualization and exploration of geospatial RDF data, new paradigms like faceted browsing have to be investigated.

[23] http://rdfunit.aksw.org

Moreover, techniques for the actual combination of semantic information and geospatial data need to be developed, as well as approaches to display them on a map. In the following, two tools are introduced, built to cover the browsing and representation aspect, respectively.

Facete: Facete is a web-based exploration and visualization application for the spatial-faceted browsing of RDF data. It provides domain independent faceted filtering capabilities operating directly on SPARQL. Facete automatically detects spatial information of resources satisfying a given facet selection and displays them on a map. This mechanism also works for cases where the geometric information is only indirectly related to the corresponding resources. Thus, the tool allows one to browse SPARQL based spatial datasets, giving visual feedback, instantly updating the map according to the adjusted selection criterion.

Mappify: Mappify is a web application to easily create map views displaying concept based points of interest. Providing faceted exploration capabilities, Mappify allows one to define custom concepts on a SPARQL endpoint. These might be e.g. restaurants that serve fish and are accessible by wheelchair. Users are enabled to quickly style the map display of such custom concepts by choosing marker icons and defining templates for the content to show when clicking them. These settings can be exported, embedded in an HTML snippet, which contains all the information to render the configured map and can be integrated in a web site. This enables users to easily create own individual map views combining the presentation of different custom points of interest on one map.

8. GeoKnow Generator Architecture and Implementation

The GeoKnow Generator unifies several different software tools for application users or application developers. The initial architecture is depicted in Figure 2. The software tools that target expert users in DB administration or designers are essentially web applications and accessible through the Debian repository of the Linked Data Stack. The Generator Workbench is GeoKnow's main application that integrates preconfigured components from the Stack according to the Linked Data life-cycle as a workflow. It provides access to public data catalogues of the domain of knowledge and the option to add proprietary datasets. It also aims to provide a layer for user administration, authorization and provenance. The components that are integrated in this Workbench communicate using HTTP, REST or SPARQL protocols.

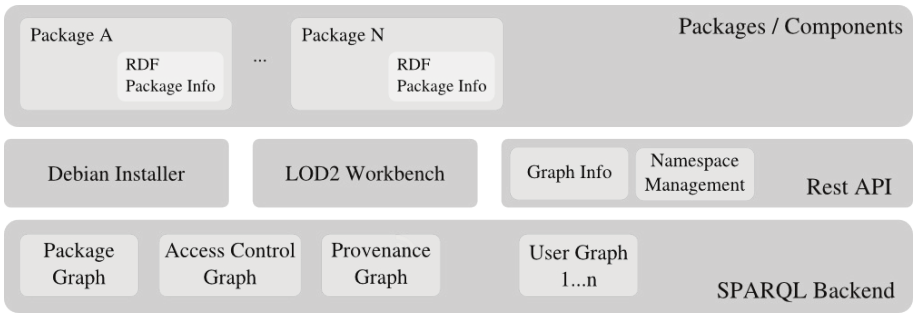

Figure 3: Current version of the GeoKnow Generator

A prototype of the GeoKnow Generator is already available at http://generator.geoknow.eu. It allows the user to triplify geospatial data, such as ESRI shapefiles and spatial tables hosted in major DBMSs using the GeoSPARQL, WGS84 or Virtuoso RDF vocabulary for geospatial representation of point features (TripleGeo). Non-geospatial data in RDF (local and online RDF files or SPARQL endpoints) or data from relational databases (via Sparqlify) can also be entered into the Generators triple store. Data from the Generator's triple store can be linked (using LIMES), enriched (using GeoLift), queried (using Virtuoso) , visualized (using Facete) and used to generate light-weight applications as JavaScript snippets (using Mappify) for specific geospatial applications. Most steps in the Linked Data Life-Cycle have been integrated in the Generator as a graph-based workflow, which allows the user to easily manage new generated data. The current version of the GeoKnow Generator is presented in Figure 3. The components comprising it are available in the Linked Data Stack (http://stack.linkeddata.org)

9. Applications of GeoKnow Components in E-Commerce

The E-Commerce domain is one of the primary usage scenarios in GeoKnow. Many use cases in this domain depend on explicit and detailed knowledge of geographical information, in particular in the tourism industry sector. We identified four major use cases:

- In geographical search, customers are searching using geospatial features and semantic knowledge, such as a "snorkelling holiday anywhere near the Mediterranean sea". Such search requests require a comprehensive knowledge base of properly interlinked information, a high accuracy, and very good query response times.
- For a geospatial market basket analysis, a marketing expert uses a data mining sys-tem to identify suitable products for a certain customer or entire target groups. The currently sparse internal information is not sufficient for great suggestions. Suggesting holiday regions and accommodation types similar to the ones a user preferred in the past would improve the quality of service. This depends on matching the properties of places and regions along with their geographic representation.
- In order to support strategic decisions, similar information is required for planning suitable regions that are most promising for establishing new products, such as building a new hotel.
- A spatial-semantic visualization would help a customer with vague and non-specific ideas about a holiday to find a suitable product. Again, this typically requires a coherent knowledge consisting of internal data, e.g., on hotel properties, and geographic properties like the distance of mountains, beaches, or cultural activities.

The components implemented in GeoKnow can be applied for generating the knowledge base and visualizing the data required for these use cases. In contrast to existing solutions, the geospatial RDF data integration and query optimization enables a much more coherent search infrastructure compared to the previously required

different databases. On that foundation, provided by the Virtuoso RDF Quad Store, we can apply the components developed along the Linked Data Life Cycle:

- the *Storage* and associated query capabilities on the vast amount of RDF data is crucial for our use cases.
- *Authoring* enables domain experts to adapt available information. Since data quality is crucial in our use cases, we typically have to limit direct modifications on the knowledge base by the public.
- via *Interlinking* we can connect data from different resources, including geospatial and non-geospatial properties. Since we are considering hundreds of thousands of places from many databases, this task requires specialized methods, such as the geospatial interlinking algorithms implemented in LIMES.
- *Classification* is required to cluster the potential large result set and gain further insights, for example with regards to the geospatial market basket analysis use case.
- for *Quality* assurance and Evolution/Repair, we can apply RDFUnit and ORE in order to filter out erroneous data and provide a high quality knowledge base required for any serious application.
- *Search/Browsing/Exploration* functionalities benefit from components like Mappify, which can be integrated in search result pages on tourism portals to visualize the points of interest in the suggested regions.

In general, GeoKnow provides the means for generating an interlinked semantically annotated geospatial knowledge base, on which novel E-Commerce applications can be built. We already tested the currently available tools on a subset of the E-Commerce datasets in the project in order to validate their general functionality, and plan to create a knowledge base using entire data sets as a validation of GeoKnow's scalability.

10. Related Work and Projects

10.1. Related Funded Projects

Next, we present some prominent projects that handle several aspects of managing Linked Data, emphasizing mainly on projects handling geospatial Linked Data.

LOD224 is a large-scale 4-year project aiming to address the following challenges: improve coherence and quality of data published on the Web, close the performance gap between relational and RDF data management, establish trust on the Linked Data Web and generally lower the entrance barrier for data publishers and users. The project is undergoing its final year, having developed a stack of tools and methodologies for exposing and managing, interlinking and fusing, searching, browsing and authoring very large amounts of Linked Data. The implemented technologies are applied on three use case scenarios: media and publishing, corporate data intranets and eGovernment.

[24] http://lod2.eu/Welcome.html

GeoKnow project builds on, extends and enriches the technologies developed in LOD2, emphasizing on the geospatial aspect of Linked Data. TELEIOS25 was an EU FP7 project that implemented technologies for developing Virtual Earth Observatories promoting the use of ontologies and linked geospatial/temporal data. The TELEIOS advances to the state of the art have been demonstrated in two use cases: (a) A Virtual Earth Observatory for the TerraSAR-X archive of DLR.and (b) Wildfire monitoring and burnt scar mapping based on satellite images and relevant geospatial data. In this use case, the National Observatory of Athens used TELEIOS technologies to reengineer its real-time wildfire monitoring and burnt scar mapping services. Among the outcomes of the project are Strabon spatiotemporal RDF store and Sextant visualization tool.

LEO[26] builds on and continues the work of TELEIOS in order to develop software tools that support the whole life cycle of reuse of linked open EO data and related linked geospatial data. This includes publishing, interlinking, searching, browsing, visualization, tools. The project's use case consists in the development of a precision farming application that is heavily based on such data.

SmartOpenData[27] aims at creating a Linked Open Data infrastructure (including software tools and data) fed by public and freely available data resources, existing sources for biodiversity and environment protection and research in rural and European protected areas and its National Parks. It will focus on how Linked Open Data can be applied generally to spatial data resources and specifically to public open data portals, GEOSS Data-CORE, GMES, INSPIRE and voluntary data (OpenStreetMap, GEPWIKI, etc.).

The goal of SemaGrow[28] project is to develop a framework for querying distributed triple stores containing large, live, constantly updated datasets and streams that are published in heterogeneous formats. The project focuses on the following key challenges: (a) Develop novel algorithms and methods for querying distributed triple stores (b) Develop scalable and robust semantic indexing algorithms (c) Investigate how to optimize the effectiveness of schema translations.

DIACHRON[29] takes on the challenges of evolution, archiving, provenance, annotation, citation, and data quality in the context of Linked Open Data and modern database systems. DIACHRON intends to automate the collection of metadata, provenance and all forms of contextual information so that data are accessible and usable at the point of creation and remain so indefinitely. The results of DIACHRON are evaluated in three large-scale use cases: open governmental data life-cycles, large enterprise data intranets and scientific data ecosystems in the life sciences.

10.2. Related Work in regard to the Geospatial Linked Data Life-cycle

In this section, we review relevant related work for each phase of the Linked Data Life-cycle.

[25] http://www.earthobservatory.eu/
[26] http://linkedeodata.eu/
[27] http://www.smartopendata.eu/
[28] http://www.semagrow.eu/
[29] http://www.diachron-fp7.eu/

10.2.1. Extraction

There are various approaches for transforming/extracting conventional data to RDF. Indicatively, some approaches are presented next. Sparqlify[30] is a SPARQL-SQL query rewriter that allows the definition of RDF views using a Sparqlification Mapping Language. This way, it enables SPARQL queries on relational databases. Similarly, D2RQ[31] allows querying relational database with SPARQL, by creating virtual RDF graphs and exploiting a mapping language for mapping relational database schemas to RDF vocabularies and OWL ontologies. In [5], the authors present SPARQL2XQuery, a framework that provides a mapping model for the expression of OWL-RDF/S to XML Schema map-pings as well as a method for SPARQL to XQuery translation. Through the framework, XML datasets can be turned into SPARQL endpoints. TripleGeo [49] is an ETL utility that can extract geospatial features from various sources (shapefiles and DBMSs) and transform them into Basic Geo or

GeoSPARQL compatible RDF triples. Apart from approaches that can robustly handle large volumes of data, there are also tools that focus on simplicity and graphical user interfaces, such as OpenRefine[32].

10.2.2. Storage

There is a series of RDF store implementations, varying on the supported querying facilities, the indexing schemes used and the performance [48, 13, 10, 59], as well as benchmarking initiatives that evaluate and compare these approaches [60, 6, 48, 16]. [48] provides a thorough presentation of RDF stores with geospatial support, such as Virtuoso[33], Parliament[34], Strabon [27], AllegroGraph[35], OWLIM[36], uSeekM[37], while [10] compares NoSQL approaches for storing RDF data. [59] presents stores that have proven to handle large volumes of RDF data and [60] gathers several benchmarks, as well as benchmarking results for several stores.

10.2.3. Authoring

On authoring of Linked Data, OntoWiki [15] facilitates the visual presentation of RDF data as an information map and enables intuitive authoring of semantic content. RDFauthor allows users to edit information on arbitrary RDFaannotated web pages, extending RDFa with representations for provenance and update endpoint information. PoolParty [52] allows the enrichment of resources utilizing several Linked Data sources, such as DBpedia, WordNet, etc.

---

[30] https://github.com/AKSW/Sparqlify
[31] http://d2rq.org/
[32] http://openrefine.org/index.html
[33] http://virtuoso.openlinksw.com/
[34] http://parliament.semwebcentral.org/
[35] http://www.franz.com/agraph/allegrograph/
[36] http://www.ontotext.com/owlim
[37] https://dev.opensahara.com/projects/useekm/

10.2.4. Interlinking/Fusion

The studies of [55, 14] present and compare interlinking tools on various factors: degree of automation, matching method and algorithm logic, input/output format and access methods, etc. Currently, there are two prominent approaches for interlinking: Silk [58][38] and LIMES [38][39]. The former allows users to define types of RDF links to be discovered between data sources and to combine various similarity metrics through a graphical user interface. LIMES applies space tiling and approximation techniques to compute estimates of the similarity between entities, reducing the number of comparisons and, thus, runtime, by orders of magnitude. Further, it applies indexing and bounding techniques to efficiently interlink entities based on their spatial distance. On the other hand, fusion approaches on Linked Data are less sophisticated and mostly adopt state of art techniques for data fusion. A brief overview of RDF-specific fusion tools is provided in [17]. Indicatively, these include Sieve [34], ODCleanStore [24], and KnoFuss [46]

10.2.5. Classification/Enrichment

There are several directions w.r.t. Linked Data enrichment and classification. Tools such as DL-Learner [28] aim at learning concepts in Description Logics from user-provided examples, so as to support users in constructing knowledge and learning about the data they created. GeoLift[40] aims at extracting implicit geographical information and making it explicit, through dereferencing, interlinking and Natural Language Processing. Plato [21] identifies partonomic relations (e.g. part of, member of, located in) between entities utilizing WordNet and linguistic patterns identified in web corpora, enriching both the schema and the data. PoolParty [52] allows the enrichment of resources utilizing several Linked Data sources, such as DBpedia, Geonames, WordNet, etc. An extensive overview of research works on ontology enrichment is given in [9].

10.2.6. Quality Analysis

The authors of [62] present a thorough review of Quality Assessment methods on Linked Data. 26 quality dimensions are identified and categorized into six classes: Contextual, Trust, Intrinsic, Accessibility, Representational and Dataset dynamicity dimensions. Indicatively, some of the latest approaches are briefly presented next. Sieve[41] [34] is a plat-form that focuses on quality assessment and fusion of Linked Data, and constitutes part of a larger framework for Linked Data integration. LODRefine[42] is a LOD-enabled version of Google Refine, which is an open-source tool for refining messy data. Although this tool is not focused on data quality assessment per se, it is powerful in performing preliminary cleaning of raw data. ODCleanStore[43] [24] is another framework that supports linking, cleaning,

[38] http://wifo5-03.informatik.uni-mannheim.de/bizer/silk/
[39] http://aksw.org/Projects/LIMES.html
[40] http://aksw.org/Projects/GeoLift.html
[41] http://sieve.wbsg.de/
[42] http://code.zemanta.com/sparkica/index.html
[43] http://www.ksi.mff.cuni.cz/~knap/odcs/

transformation, and quality assessment operations on Linked Data. Finally, RDFUnit[44] [26] is a test driven data-debugging framework that can run automatically and manually generated test cases against a SPARQL endpoint.

10.2.7. Evolution/Repair

An overview of several ontology repair approaches is given in [30]. The same paper also presents ORE, a tool that supports the detection of ontology modelling problems and allows users to improve OWL ontologies by fixing inconsistencies and making suggestions for adding further axioms in a semi-automatic way. LODRefine, a LOD-enabled version of Google Refine, enables cleaning, reconciling and augmenting Linked Open Data with data from Freebase and other registered services. The surveys presented in [22, 61] provide an overview of several ontology evolution approaches. [33, 47] present a linked data approach for the preservation and archiving of open heterogeneous datasets that evolve through time both at the structural and the semantic layer.

10.2.8. Browsing/Exploration

There are numerous approaches for visualization and exploration of Linked Data. An overview of several of them is given in [11]. Indicatively, some recent tools are presented next. Facete[45] offers advanced faceted search techniques and visualization of geospatial RDF data. Sextant[46] [45] allows the visualization and exploration of time-evolving linked geospatial data and the creation, sharing, and collaborative editing of temporally enriched thematic maps. Mappify[47] facilitates the creation of simple map applications based on RDF data retrieved from a SPARQL endpoint. rdf:synopsViz[48] [4] provides facilities for hierarchical charting and visual exploration of Linked Open Data, as well as on the fly statistic computations, using aggregations over the ontology hierarchy levels. CubeViz[49] utilizes the RDF Data Cube vocabulary to visualize statistical data in RDF in charts.

10.3. Linked Data Platforms

In the past years several Linked Data platforms were developed as part of funded research projects. The most prominent ones are the LOD2 Linked Data stack[50], DataLift[51] and the commercial solution TasorONE[52]. The LOD2 stack can be entitled as the predecessor of the GeoKnow workbench covering the fill linked data life-cycle. The French funded project DataLift focuses mainly on the transformation to RDF, interlinking and publishing the data. TasorONE is a cloud-based solution supporting

[44] http://aksw.org/Projects/RDFUnit.html
[45] http://aksw.org/Projects/Facete.html
[46] http://sextant.di.uoa.gr/
[47] http://mappify.aksw.org/
[48] http://83.212.125.131:8084/synopsViz/
[49] http://cubeviz.aksw.org/
[50] http://stack.linkeddata.org/
[51] http://datalift.org/
[52] http://tasorone.com/

the collaborative development of ontology, triplifying the data and publishing it as SPARQL endpoint. LEO builds on and extends the results of TELEIOS project, aiming to develop a stack of tools[53] handling the complete life-cycle of Linked Earth Observation data.

11. Conclusion and Future Work

After 1.5 years in the project, there have been several advancements of the state of the art in geospatial Linked Data through the GeoKnow project. GeoSPARQL compliance and performance in Virtuoso has been significantly improved with full support for OGC geometries and the GeoSPARQL standard in the near future. The performance of link discovery frameworks has been improved by at least an order of magnitude on large datasets. FAGI has been developed to support fusion of thematic and geospatial metadata of resources, either manually or automatically. The RDFUnit quality assessment framework has been created and applied to several large datasets and ontologies. Existing standards such as GeoSPARQL have been extensively evaluated to identify shortcomings and challenges. Facet-based browsing techniques have been refined and the Mappify tool for lightweight geospatial web application development created. All of those components will be refined and further mature within the project. The major focus of future work will be the validation of those technologies in the project and third party use cases as well as the further establishment of the Linked Data Stack as a community tool repository.

Acknowledgements

Work on GeoKnow is founded by the European Commission within the FP7 Information and Communication Technologies Work Programme (Grant Agreement No. 318159). The consortium consists of the following partners: Institute of Applied Computer Science / University of Leipzig (Germany), Institute for the Management of Information Sys-tems/Athena Research and Innovation Center (Greece), OpenLink Software Ltd (United Kingdom), Unister GmbH (Germany), Brox (Germany), Ontos AG (Switzerland), and Institute Mihailo Pupin (Serbia).

References

[1] Albrecht, Jochen, Derman, Brandon, Ramasubramanian, and Laxmi. Geo-ontology tools: The missing link. Transactions in GIS, 12(4):409–424, 2008.
[2] Sören Auer and Jens Lehmann. Making the web a data washing machine - creating knowledge out of interlinked data. Semantic Web Journal, 2010.
[3] Sören Auer, Jens Lehmann, and Sebastian Hellmann. Linked geodata: Adding a spatial dimension to the web of data. In Proceedings of the 8th International Semantic Web Conference, ISWC ’09, pages 731–746, Berlin, Heidelberg, 2009. Springer-Verlag.
[4] Nikos Bikakis, Melina Skourla, and George Papastefanatos. rdf:synopsviz - a framework for hierarchical linked data visual exploration and analysis. In Proceedings of ESWC’14 (Demo), 2014.

[53] http://linkedeodata.eu/misc/LEO-D1.1.pdf

[5] Nikos Bikakis, Chrisa Tsinaraki, Ioannis Stavrakantonakis, Nektarios Gioldasis, and Stavros Christodoulakis. The sparql2xquery interoperability framework. World Wide Web, pages 1–88, 2014.
[6] C. Bizer and A. Schultz. The berlin sparql benchmark. International Journal On Semantic Web and Information Systems, 5:1–24, 2009.
[7] Andreas Brodt, Daniela Nicklas, and Bernhard Mitschang. Deep integration of spatial query process-ing into native rdf triple stores. In Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS '10, pages 33–42, New York, NY, USA, 2010. ACM.
[8] Agustina Buccella, Alejandra Cechich, and Pablo R. Fillottrani. Ontology-driven geographic informa-tion integration: A survey of current approaches. Computers and Geosciences, 35(4):710–723, 2009.
[9] Lorenz Bühmann and Jens Lehmann. Pattern based knowledge base enrichment. In Harith Alani, Lalana Kagal, Achille Fokoue, Paul Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha Noy, Chris Welty, and Krzysztof Janowicz, editors, The Semantic Web – ISWC 2013, volume 8218 of Lecture Notes in Computer Science, pages 33–48. Springer Berlin Heidelberg, 2013.
[10] Philippe Cudré-Mauroux, Iliya Enchev, Sever Fundatureanu, Paul Groth, Albert Haque, Andreas Harth, Felix Leif Keppmann, Daniel Miranker, JuanF. Sequeda, and Marcin Wylot. Nosql databases for rdf: An empirical evaluation. In Harith Alani, Lalana Kagal, Achille Fokoue, Paul Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha Noy, Chris Welty, and Krzysztof Janowicz, editors, The Semantic Web – ISWC 2013, volume 8219 of Lecture Notes in Computer Science, pages 310–325. Springer Berlin Heidelberg, 2013.
[11] Aba-Sah Dadzie and Matthew Rowe. Approaches to visualising linked data: A survey. Semant. web, 2(2):89–124, April 2011.
[12] Catherine Dolbear and Glen Hart. Ontological bridge building – using ontologies to merge spatial datasets. In AAAI Spring Symposium: Semantic Scientific Knowledge Integration, pages 15–20. AAAI, 2008.
[13] D. C. Faye, O. Cure, and G. Blin. A survey of RDF storage approaches. ARIMA Journal, 15:11–35, 2012.
[14] Alfio Ferrara, Andriy Nikolov, and François Scharffe. Data linking for the semantic web. Int. J. Semantic Web Inf. Syst., 7(3):46–76, 2011.
[15] Philipp Frischmuth, Michael Martin, Sebastian Tramp, Thomas Riechert, and Sören Auer. OntoWiki – An Authoring, Publication and Visualization Interface for the Data Web. Semantic Web Journal, 2014.
[16] George Garbis, Kostis Kyzirakos, and Manolis Koubarakis. Geographica: A benchmark for geospatial rdf stores (long version). In Harith Alani, Lalana Kagal, Achille Fokoue, Paul Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha Noy, Chris Welty, and Krzysztof Janowicz, editors,
[17] The Semantic Web – ISWC 2013, volume 8219 of Lecture Notes in Computer Science, pages 343–359. Springer Berlin Heidelberg, 2013. G. Giannopoulos et al. Geoknow eu/fp7 project. fusing of geographic features., 2013.
[18] He Hu and Xiaoyong Du. Linking open spatiotemporal data in the data clouds. In Jian Yu, Salvatore Greco, Pawan Lingras, Guoyin Wang, and Andrzej Skowron, editors, Rough Set and Knowledge Technology, volume 6401 of Lecture Notes in Computer Science, pages 304–309. Springer Berlin Heidelberg, 2010.
[19] Open Geospatial Consortium Inc. Opengis implementation specification for geographic information – simple feature access- part 2: Sql option. opengis implementation standard. version 1.2.1, 04/08/2010, 2010.
[20] Open Geospatial Consortium Inc. Ogc geosparql standard - a geographic query language for rdf data. version 1.0, 27/04/2012, 2012.
[21] Prateek Jain, Pascal Hitzler, Kunal Verma, Peter Z. Yeh, and Amit P. Sheth. Moving beyond sameas with plato: Partonomy detection for linked data. In Proceedings of the 23rd ACM Conference on Hypertext and Social Media, HT '12, pages 33–42, New York, NY, USA, 2012. ACM.
[22] Asad Masood Khattak, Khalid Latif, Songyoung Lee, and Young-Koo Lee. Ontology evolution: A survey and future challenges. In Dominik Sle¸zak, Tai-hoon Kim, Jianhua Ma, Wai-Chi Fang, Frode Eika Sandnes, Byeong-Ho Kang, and Bongen Gu, editors, U- and E-Service, Science and Technology, vol-ume 62 of Communications in Computer and Information Science, pages 68–75. Springer Berlin Hei-delberg, 2009.
[23] Eva Klien and Florian Probst. Requirements for geospatial ontology engineering. In 8th Conference on Geographic Information Science (AGILE 2005), pages 251–260. Citeseer, 2005.
[24] Tomáš Knap, Jan Michelfeit, and Martin Necaský. Linked open data aggregation: Conflict resolution and aggregate quality. In Proceedings of the 2012 IEEE 36th Annual Computer Software and Applications Conference Workshops, COMPSACW '12, pages 106–111, Washington, DC, USA, 2012. IEEE Computer Society.
[25] Dimitris Kontokostas, Patrick Westphal, Sören Auer, Sebastian Hellmann, Jens Lehmann, and Roland

Cornelissen. Databugger: A test-driven framework for debugging the web of data. In Proceedings of the 23rd international conference on World Wide Web Companion, 2014.
[26] Dimitris Kontokostas, Patrick Westphal, Sören Auer, Sebastian Hellmann, Jens Lehmann, and Roland Cornelissen. Test-driven evaluation of linked data quality. In WWW, 2014.
[27] Kostis Kyzirakos, Manos Karpathiotakis, and Manolis Koubarakis. Strabon: A semantic geospatial dbms. In Philippe Cudré-Mauroux, Jeff Heflin, Evren Sirin, Tania Tudorache, Jérôme Euzenat, Manfred Hauswirth, Josiane Xavier Parreira, Jim Hendler, Guus Schreiber, Abraham Bernstein, and Eva Blomqvist, editors, The Semantic Web ISWC 2012, volume 7649 of Lecture Notes in Computer Science, pages 295–311. Springer Berlin Heidelberg, 2012.
[28] Jens Lehmann. Dl-learner: Learning concepts in description logics. J. Mach. Learn. Res., 10:2639–2642, December 2009.
[29] Jens Lehmann, Sören Auer, Lorenz Bühmann, and Sebastian Tramp. Class expression learning for ontology engineering. Journal of Web Semantics, 9:71 – 81, 2011.
[30] Jens Lehmann and Lorenz Bühmann. Ore - a tool for repairing and enriching knowledge bases. In Proceedings of the 9th International Semantic Web Conference on The Semantic Web - Volume Part II, ISWC'10, pages 177–193, Berlin, Heidelberg, 2010. Springer-Verlag.
[31] Jens Lehmann and Pascal Hitzler. Concept learning in description logics using refinement operators. Machine Learning, 78:203–250, 2010.
[32] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. DBpedia - a large-scale, multilingual knowledge base extracted from wikipedia. Semantic Web Journal, 2014.
[33] M. Meimaris, G. Papastefanatos, C. Pateritsas, T. Galani, and Y. Stavrakas. Towards a framework for managing evolving information resources on the data web. In Proceedings of Profiles Workshop of ESWC'14, 2014.
[34] Pablo N. Mendes, Hannes Mühleisen, and Christian Bizer. Sieve: Linked data quality assessment and fusion. In Proceedings of the 2012 Joint EDBT/ICDT Workshops, EDBT-ICDT '12, pages 116–123, New York, NY, USA, 2012. ACM.
[35] Mohamed Morsey, Jens Lehmann, Sören Auer, Claus Stadler, and Sebastian Hellmann. Dbpedia and the live extraction of structured data from wikipedia. Program: electronic library and information systems, 46:27, 2012.
[36] Ian Murdock. Overview of the debian gnu/linux system. Linux Journal, 1994(6es), October 1994.
[37] D. Nebert. Developing spatial data infrastructures: The sdi cookbook. technical report, global spatial data infrastructure, 2004.
[38] Axel-Cyrille Ngonga Ngomo. Link discovery with guaranteed reduction ratio in affine spaces with minkowski measures. In Proceedings of the 11th International Conference on The Semantic Web – Volume Part I, ISWC'12, pages 378–393, Berlin, Heidelberg, 2012. Springer-Verlag.
[39] Axel-Cyrille Ngonga Ngomo. On link discovery using a hybrid approach. J. Data Semantics, 1(4):203–217, 2012.
[40] Axel-Cyrille Ngonga Ngomo. Orchid - reduction-ratio-optimal computation of geo-spatial distances for link discovery. In International Semantic Web Conference (1), pages 395–410, 2013.
[41] Axel-Cyrille Ngonga Ngomo and Klaus Lyko. Eagle: Efficient active learning of link specifications using genetic programming. In ESWC, pages 149–163, 2012.
[42] Axel-Cyrille Ngonga Ngomo, Klaus Lyko, and Victor Christen. Coala - correlation-aware active learning of link specifications. In ESWC, pages 442–456, 2013.
[43] Axel-Cyrille Ngonga Ngomo, Mohamed Ahmed Sherif, and Klaus Lyko. Unsupervised link discovery through knowledge base repair. In ESWC, pages 380–394, 2014.
[44] Axel-Cyrille Ngonga Ngomo and Sören Auer. Limes - a time-efficient approach for large-scale link discovery on the web of data. In Proceedings of IJCAI, 2011. Charalampos Nikolaou, Kallirroi Dogani, Kostis Kyzirakos, and Manolis Koubarakis. Sextant: Browsing and mapping the ocean of linked geospatial data. In Philipp Cimiano, Miriam Fernández, Vanessa Lopez, Stefan Schlobach, and Johanna Völker, editors, The Semantic Web: ESWC 2013 Satellite Events,
[45] volume 7955 of Lecture Notes in Computer Science, pages 209–213. Springer Berlin Heidelberg, 2013.
[46] Andriy Nikolov, Victoria Uren, Enrico Motta, and Anne Roeck. Integration of semantically annotated data by the knofuss architecture. In Proceedings of the 16th International Conference on Knowledge Engineering: Practice and Patterns, EKAW '08, pages 265–274, Berlin, Heidelberg, 2008. Springer-Verlag.
[47] George Papastefanatos and Yannis Stavrakas. Diachronic linked data: Capturing the evolution of structured interrelated information on the web. ERCIM News, 2014(96), 2014.
[48] K. Patroumpas et al. Geoknow eu/fp7 project. market and research overview., 2013.
[49] Kostas Patroumpas, Michalis Alexakis, Giorgos Giannopoulos, and Spiros Athanasiou. Triplegeo: an etl tool for transforming geospatial data into rdf triples. In EDBT/ICDT Workshops, pages 275–278, 2014.

[50] Matthew S. Perry. A Framework to Support Spatial, Temporal and Thematic Analytics over Semantic Web Data. PhD thesis, Dayton, OH, USA, 2008. AAI3324256.
[51] David A. Randell, Zhan Cui, and Anthony G. Cohn. A spatial logic based on regions and connection. In Proceedings 3rd International Conference on Knowledge Representation and Reasoning, 1992.
[52] Thomas Schandl and Andreas Blumauer. Poolparty: Skos thesaurus management utilizing linked data. In The Semantic Web: Research and Applications, volume 6089 of Lecture Notes in Computer Science, pages 421–425. Springer Berlin Heidelberg, 2010.
[53] Claus Stadler, Jens Lehmann, Konrad Höffner, and Sören Auer. Linked geodata: A core for a web of spatial open data. Semantic Web Journal, 3(4):333–354, 2012.
[54] Claus Stadler, Michael Martin, and Sören Auer. Exploring the web of spatial data with facete. In Proceedings of the companion publication of the 23rd international conference on World wide web companion, pages 175–178. International World Wide Web Conferences Steering Committee, April 2014.
[55] STI. A survey on data interlinking methods. Technical report, 2011.
[56] Sebastian Tramp, Philipp Frischmuth, Timofey Ermilov, and Sören Auer. Weaving a Social Data Web with Semantic Pingback. In P. Cimiano and H.S. Pinto, editors, Proceedings of the EKAW 2010 – Knowledge Engineering and Knowledge Management by the Masses; 11th October-15th October 2010 - Lisbon, Portugal, volume 6317 of Lecture Notes in Artificial Intelligence (LNAI), pages 135–149, Berlin/ Heidelberg, October 2010. Springer.
[57] Sebastian Tramp, Norman Heino, Sören Auer, and Philipp Frischmuth. RDFauthor: Employing RDFa for collaborative Knowledge Engineering. In P. Cimiano and H.S. Pinto, editors, Proceedings of the EKAW 2010 - Knowledge Engineering and Knowledge Management by the Masses; 11th October-15th October 2010 - Lisbon, Portugal, volume 6317 of Lecture Notes in Artificial Intelligence (LNAI), pages 90–104, Berlin / Heidelberg, October 2010. Springer.
[58] Julius Volz, Christian Bizer, Martin Gaedke, and Georgi Kobilarov. Discovering and maintaining links on the web of data. In Proceedings of the 8th International Semantic Web Conference, ISWC '09, pages 650–665, Berlin, Heidelberg, 2009. Springer-Verlag.
[59] World Wide Web Consortium. Largetriplestores - w3c wiki. http://www.w3.org/wiki/LargeTripleStores.
[60] World Wide Web Consortium. Rdfstorebenchmarking - w3c wiki. http://www.w3.org/wiki/RdfStoreBenchmarking.
[61] Fouad Zablith, Grigoris Antoniou, Mathieu d'Aquin, Giorgos Flouris, Haridimos Kondylakis, and Enrico Motta. Ontology evolution: a process-centric survey. The Knowledge Engineering Review, page FirstView, 2013.
[62] Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. Quality assessment methodologies for linked open data (under review). Semantic Web Journal, 2014. This article is still under review.
[63] Xiaofang Zhai, Lei Huang, and Zhifeng Xiao. Geo-spatial query based on extended sparql. In Geoinformatics, pages 1–4. IEEE, 2010.
[64] Tian Zhao, Chuanrong Zhang, Mingzhen Wei, and Zhong-Ren Peng. Ontology-based geospatial data query and integration. In GIScience, volume 5266 of Lecture Notes in Computer Science, pages 370– 392. Springer