FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

Titel der Arbeit

Optionaler Untertitel der Arbeit

BACHELORARBEIT

zur Erlangung des akademischen Grades

Bachelor of Science

im Rahmen des Studiums

Software und Information Engineering

eingereicht von

Kevin Haller
Matrikelnummer 1325694

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Mag.rer.soc.oec Stefan Biffl, Univ.Doz.
Mitwirkung: Reka Marta Sabou, Project Ass.

Wien, 1. August 2016

Kevin Haller	Stefan Biffl

FAKULTÄT FÜR INFORMATIK

Faculty of Informatics

Title

Optional Subtitle of the Thesis

BACHELOR'S THESIS

submitted in partial fulfillment of the requirements for the degree of

Bachelor of Science

in

Software and Information Engineering

by

Kevin Haller
Registration Number 1325694

to the Faculty of Informatics

at the TU Wien

Advisor: Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Mag.rer.soc.oec Stefan Biffl, Univ.Doz.
Assistance: Reka Marta Sabou, Project Ass.

Vienna, 1st August, 2016

Kevin Haller	Stefan Biffl

Erklärung zur Verfassung der Arbeit

Kevin Haller
Address

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 1. August 2016

Kevin Haller

Danksagung

TODO: Ihr Text hier.

Acknowledgements

TODO: Enter your text here.

Kurzfassung

TODO: Ihr Text hier.

Abstract

TODO: Enter your text here.

Contents

CHAPTER 1

Introduction

1.1 Motivation

Today‘s universities usually have to manage a significant amount of information and have to provide systems to handle common services like finding and booking courses for students. As the current time demands it, those services are often provided over web sites participating in the Web of Documents (see figure 1.1). The domain of a university is quite complex and as a consequence it is likely that different isolated information systems handling a particular part of the domain evolve; resulting in an environment, where information is distributed over multiple disconnected data silos that may have different formats and/or data owners. Such a situation prevents universities of fully exploiting their data [18].

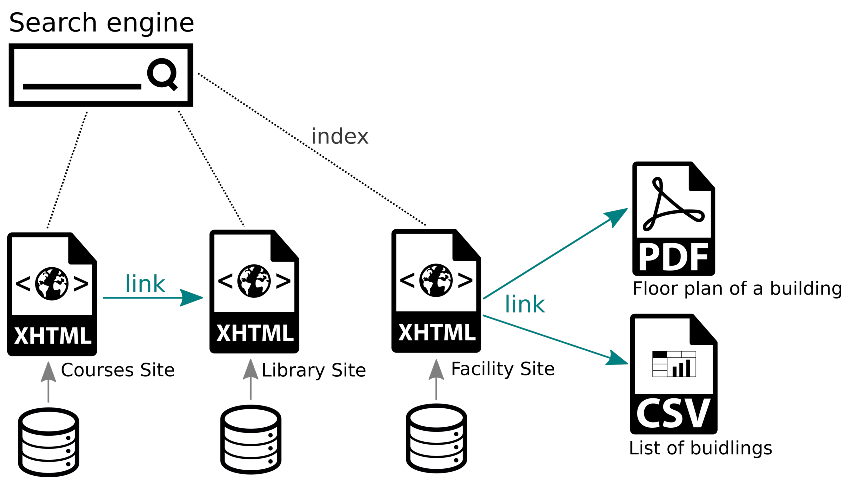

Figure 1.1: Illustration of the Web of Documents

The same applies at the moment to the Vienna University of Technology (TU Vienna). In order to find a special location like a certain lecture hall and how it can be accessed without obstacles like stairways for persons with mobility-impairments, one has to search for information on different web sites, scan floor plans and eventually construct a convenient route to the location based on the gathered knowledge. For humans this procedure does not constitute a problem, but think of machines. The process of information retrieval through data mining or harvesting is quite difficult and/or time consuming. As a consequence application developers that may have innovative ideas, which would be a benefit for the information environment of the university, face a barrier that is hard to overcome. Linked Data (LD) is one way to transform this information published on multiple web sites into a university-wide data space (see figure 1.2).

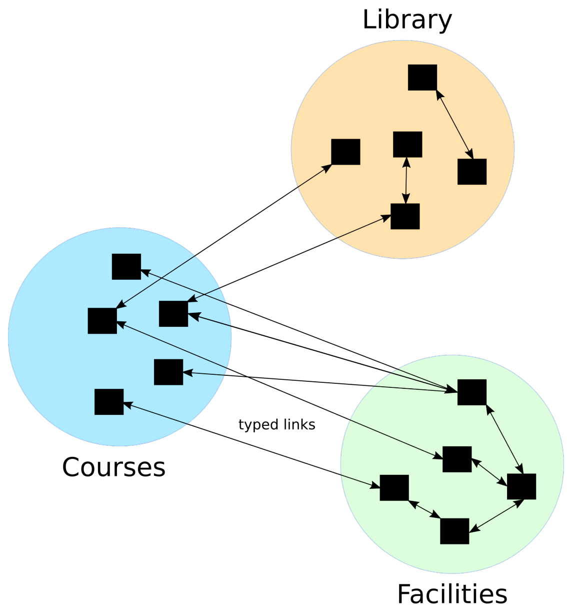

Figure 1.2: Illustration of the Web of Data

> LD describes a method of publishing structured data so that it can be interlinked and become more useful. It builds upon standard Web technologies, but rather than using them to serve web pages for human readers, it extends them to share information in a way that can be read automatically by computers. This enables data from different sources to be connected and queried [4].

This thesis aims to show the potential that could evolve, if spatial data about TU Vienna is transformed into a machine-readable form such as LD, by proposing a prototype of a map application based on a subset of spatial data about TU Vienna. This subset is intended to comply with the principles of LD [2] and to fulfil the submission criteria to the Linking Open Data cloud[1], which are as follows [6]:

- There must be resolvable `http://` (or `https://`) URIs.
- They must resolve, with or without content negotiation, to RDF data in one of the popular RDF formats (RDFa, RDF/XML, Turtle, N-Triples).
- The dataset must contain at least 1000 triples.
- The dataset must be connected via RDF links to a dataset that is already in the diagram. This means, either your dataset must use URIs from the other dataset, or vice versam. We arbitrarily require at least 50 links.
- Access of the entire dataset must be possible via RDF crawling, via an RDF dump, or via a SPARQL endpoint.

1.2 Problem description

As already mentioned does the TU Vienna distribute spatial information over different sources. Few of this spatial data is actually in a machine-readable format, so that it can be transformed into LD automatically. The ***1st problem*** to solve is therefore the transformation process of the unstructured part of the data into a machine-readable format. Section 4.1 is suggesting a solution for this problem.

The ***2nd problem*** is how the structured spatial data shall be described so that it can be easily used by application developers as well as integrated into the Web of Data. The developed ontology shall be compliant to the best practises for publishing ontologies [3]. Section 4.2 is suggesting a prototype of an ontology by taking already existing approaches (see section 3.1 and 3.3) into consideration.

After the spatial data was transformed into LD, a system must be designed to expose this data on the Web in order to qualify as Linked Open Data (LOD), representing the ***3rd problem*** to solve. The solution shall take best practises for publishing LD[9] and best practises for spatial data on the Web[16] into consideration. Section 4.3 is proposing an architectural prototype of such a system.

The ***4th problem*** is the development of a map application that is based on the resulting LOD. The application shall answer the following question: *"Give me all learning rooms that are nearby that are free in a given time range and are accessible without obstacles for person with mobility-impairments"* to show the potential of the data. Section 4.4 presents a solution to this problem.

[1] `http://lod-cloud.net/`

1.3 Structure of the thesis

This chapter outlined the motivation for writing this thesis and gave a description of the problems for which a solution will be suggested. The rest of this thesis is structured as follows: Chapter 2 discusses the basic concepts, principles and technologies that build the foundation of LD and the Semantic Web. It is intended to be a brief introduction for readers that are not familiar with this topic. Chapter 3 provides a summary of past efforts in research and works related to the focus of this thesis. In the subsequent chapter 4 a solution for the given problems will be suggested and discussed. This includes the prototype of an ontology that models the problem domain as well as an architectural prototype of a system that manages a subset of spatial data about the TU Vienna in form of LD and provides a human- and machine-friendly interface to it. Finally, the implemented solution is evaluated and conclusions are drawn in the last chapter 5. It provides furthermore an outlook to future work and potential improvements.

CHAPTER 2

Background

2.1 Web of Data

TODO: Enter your text here.

2.2 RDF

Resource description framework (RDF) **TODO: Enter your text here.**

2.3 Ontology

TODO: Enter your text here.

2.3.1 RDFS

TODO: Enter your text here.

2.3.2 OWL

TODO: Enter your text here.

2.4 SPARQL

SPARQL protocol and RDF query language (SPARQL) **TODO: Enter your text here.**

CHAPTER 3

Related Work

In this chapter important efforts and research related to this thesis are outlined. Section 3.1 describes and evaluates geospatial ontologies that were suggested by [16] or discovered in the dataset of Linked Open Vocabulary (LOV) through a systematic search. Section 3.2 lists already existing map applications that are based on LOD and describes their capabilities. The final section 3.3 describes done research and designed ontologies for modelling indoor environments.

3.1 Geospatial ontologies

This section is going to outline common ontologies for describing geospatial data and to evaluate them from the perspective of the given problem description (see 1.2).The ontologies were either be suggested by [16] or discovered in the dataset of LOV by common spatial search terms like *'feature'*, *'geometry'* or *'location'*. *Feature* and *Geometry* are widely used terms in these ontologies and describe two different concepts of geospatial science. A *feature* is simply a spatial entity that can be everything from a building with fixed position to a movable food truck as long as it has a spatial extent. *Geometry* as the name suggests is a certain geometric shape from points, lines to polygons. Geometric shapes can be used to describe the spatial extent of *features*.

The ontologies are visualized using the VOWL2 notation[11]. Classes are modelled as circles, properties in form of rectangles; data type properties have a green colour and object properties a blue one. Literals are presented in a yellow rectangle.

3.1.1 Basic Geo Vocabulary (WGS84)

Basic Geo (WGS84 long/lat) Vocabulary (GEO)[5] is a lightweight ontology published by W3C to describe the position of a spatial *feature* using the WGS84 geodetic reference datum[1] with the properties `longitude`, `latitude` and `attitude`. The general class `SpatialThing` is the domain of all these properties. Figure 3.1 shows a visualization of this ontology.

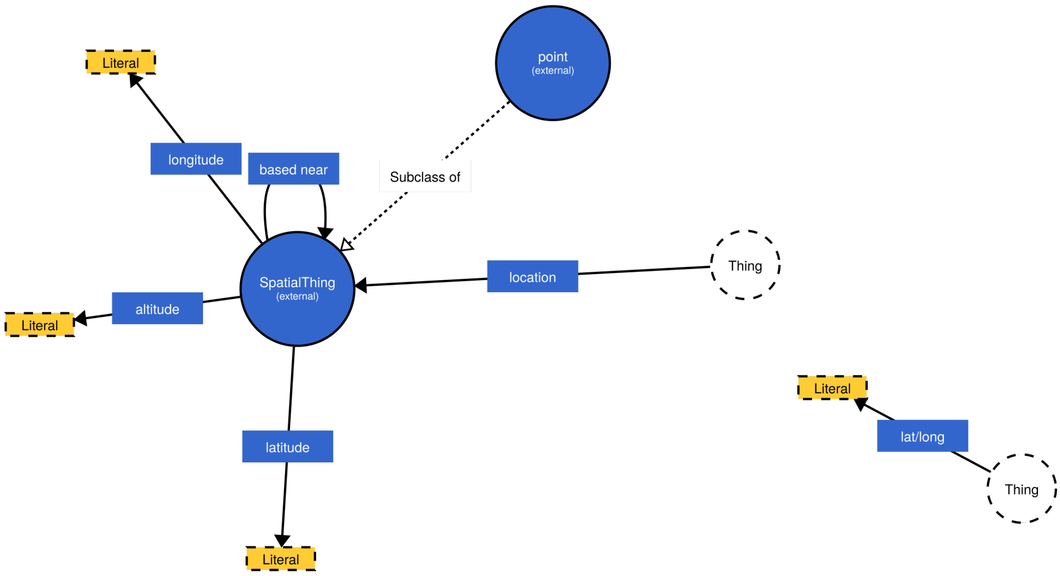

Figure 3.1: Visualization of GEO

Advantages are it‘s simplicity and popularity as an analysis of the *Comprehensive Knowledge Archive* (CKAN, "The Data Hub"[2]) indicates. This analysis shows that GEO is used in **538** different datasets in the archive and is thereby the 4th most used ontology regarding usage in different datasets [15]. It is used by big players like DBpedia[3], LinkedGeoData[4] and GeoNames[5]. Listing 3.1 shows a data snippet of DBpedia describing the town square 'Karlsplatz' in Vienna with GEO.

Listing 3.1: Snippet of DBpedia

```
@prefix dbr:  <http://dbpedia.org/resource/> .
@prefix geo:  <http://www.w3.org/2003/01/geo/wgs84_pos#> .

dbr:Karlsplatz  rdf:type  geo:SpatialThing ;
  rdfs:label  "Karlsplatz"@en , "Karlsplatz (Wien)"@de ;
  geo:lat     "48.19916534423828125"^^xsd:float ;
  geo:long    "16.37000839233398438"^^xsd:float .
```

[1] http://en.wikipedia.org/wiki/World_Geodetic_System
[2] http://thedatahub.org
[3] http://dbpedia.org
[4] http://linkedgeodata.org/About
[5] http://linkedgeodata.org/About

GEO evolved into a standard for presenting the location of points of interest and as described by [9] in an article about best practises, standardized ontologies shall be used wherever possible to facilitate inclusion into the Web of Data.

However, this ontology is not sufficient for describing spatial data more complex than a single point in a coordinate reference system.

3.1.2 NeoGeo Geometry/Spatial Ontology

NeoGeo Geometry Ontology (NGEO)[6] is an ontology that aims to provide a comprehensive descriptive power for modelling geographic regions, whereas NeoGeo Spatial Ontology (SPATIAL)[7] aims to describe topological relationships between *features*. Both ontologies were designed to have a strict distinction between *features* and *geometries* [12].

NGEO

NGEO follows the principle of modelling geometries in pure RDF, which leads to a number of classes and properties for covering all shapes suggested by Simple Features Profile[8]. This includes points, lines and polygons. Shapes that go beyond a simple point are represented by a RDF collection of `point` instances, whereby `point` is a external class of GEO. Figure 3.2 shows a visualization of this ontology.

One advantage of this approach is that triple stores do not have to meet special requirements in order to enable the querying over geometric data like it is the case for GeoSPARQL. However, this shifts the burden of writing geometric queries to developers. A further problem is the high demand for resource identifiers or blank nodes (at least for each point of a shape), as the example of describing the geometry of Iceland shows[9]; this significantly increases the verbosity of the data without adding particular gains of expressitivity to the data[1]. NGEO has also a generic property `asWKT` for pointing to the Well-known text (WKT) serialization of a geometry, which has at the moment of writing the status 'deprecated'.

LinkedGeoData and GeoNames use both the `geometry` property of this ontology to point to the *geometry* of a *feature*, but use other approaches to represent it.

SPATIAL

SPATIAL is an ontology that aims to provide properties to make topological relationships of `features` explicit, which else would only exist implicit in the geometric data. Included properties cover all topological relationships described by RCC-8 like does a *feature* overlap with another *feature*. One problem of making the relationships explicit is that these relationships have to be adopted if the *geometry* of *features* changes.

[6] `http://geovocab.org/geometry`
[7] `http://geovocab.org/spatial`
[8] `http://www.ogcnetwork.net/gml-sf`
[9] `http://nuts.geovocab.org/id/IS_geometry.ttl`

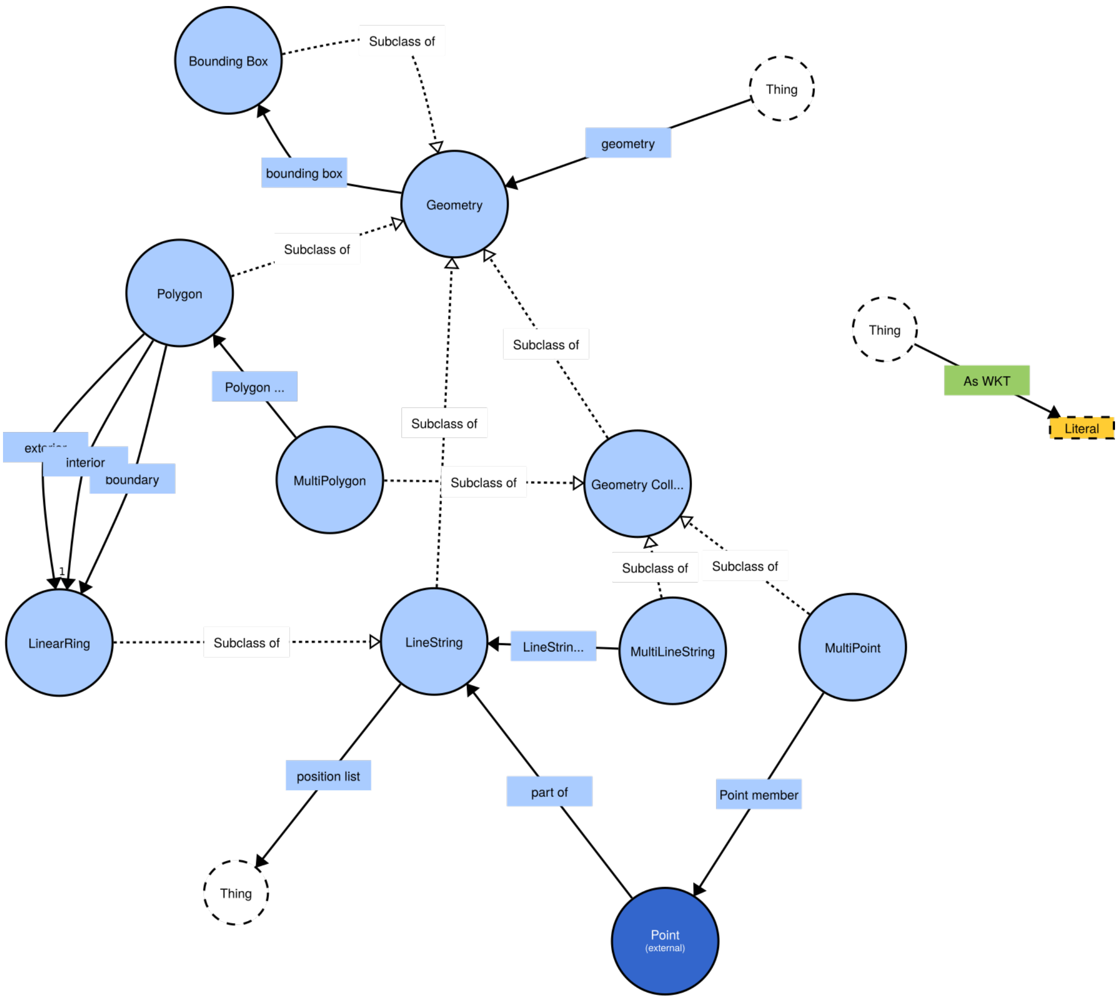

Figure 3.2: Visualization of NGEO

3.1.3 GeoSPARQL

GeoSPARQL defines a set of SPARQL extension functions, a set of Rule Interchange Format (RIF) rules, and a core ontology for geographic information[14].

3.1.4 schema.org

schema.org[10] is an initiative that was started by Google, Bing and Yahoo! (subsequently joined by Yandex) to promote a common ontology for publishing structured data mark-up on web pages [7]. This machine-readable information can then be extracted from web pages by search engines to improve search results.

[10] http://schema.org/

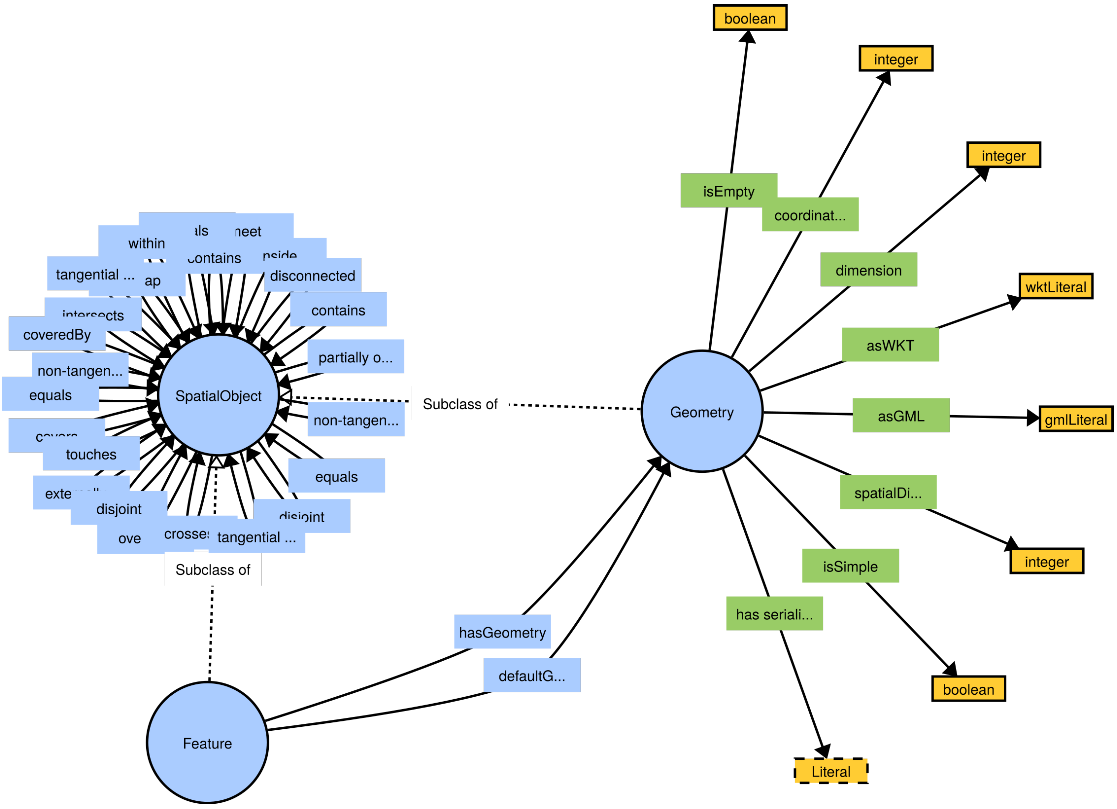

Figure 3.3: Visualization of GeoSPARQL

schema.org is a quite broad ontology covering a set of different domains from events, organizations to places. For this thesis especially the class `Place` is of interest describing entities with a fixed, physical extent; it has a set of more specific subclasses. This class has properties to assign an address to it‘s instances (`address`) as well as to provide geo-information (`geo`). The value of the property `geo` can either be instances of the class `GeoCoordinates` or `GeoShapes`. `GeoCoordinates` represents a point in a georeference system similar to GEO. `GeoShapes` on the other hand provide the ability to describe more complex geometric shapes including cirlces, boxes, polygons and simple lines. All those shapes are expressed in textual form following an own specific pattern, in contrast to GeoSparql, which declares the use of Geography Markup Language (GML) or WKT. However, such literals could be transformed into WKT by string operations (supported in SPARQL) except for the circle, which has no direct representation in WKT.

One advantage of this ontology is the visibility to search engines when exposed on web pages in one of the supported formats (Microdata[11], RDF in Attributes (RDFa)[12] and

[11] `https://www.w3.org/TR/microdata/`
[12] `https://www.w3.org/TR/rdfa-syntax/`

JSON-LD[13]). As mentioned earlier, the embedded data can then be extracted and in fact major search engines crawl for it to enhance search results[8]. **TODO: mention function**. There are furthermore odd ascriptions of properties like that instances of the class `Beach` (subclass of `Place`) can have the property `faxNumber`; it must be considered that schema.org is meant for web masters to add structured metadata to their web sites [17] and not to describe a specific domain precisely.

3.1.5 ISA Programme Location Core Vocabulary

ISA Programme Location Core Vocabulary (LOCN) is an ontology that provides a set of properties and classes to describe the geometry, address and location of a *feature*. Figure 3.6 shows a visualization of this ontology. It provides a comprehensive set of properties to describe the address of a *feature*, but a mininal set for geometries and locations. For geometries this ontology suggests the use of external classes or a simple literal representing the serialization of the geometry in formats such as WKT and GML. Those suggested classes are *Geometry* of GeoSPARQL with it‘s more specific subclasses and `GeoCoordinates` as well as `GeoShapes` from schema.org. In case of representing simple points also `point` of GEO is mentioned. [13] Howerver, this quite broad range of possibilities does not make it easy for the consumer of such LD.

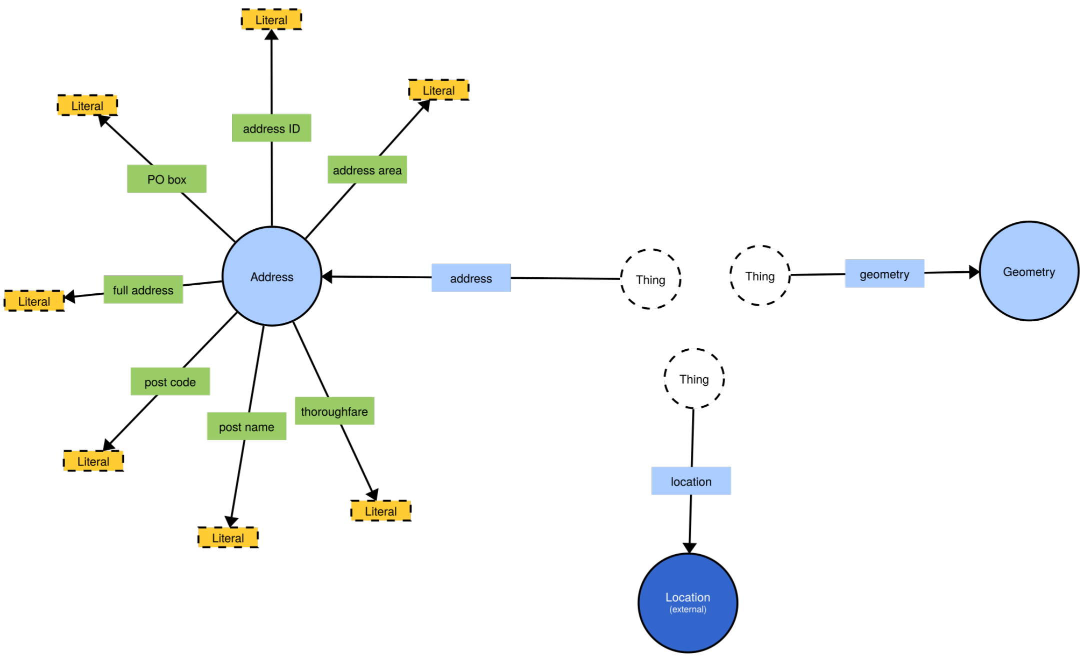

Figure 3.4: Visualization of LOCN

[13]`http://json-ld.org/`

3.1.6 Other ontologies

3.2 Linked Open Data based map applications

Multiple universities (e.g. Linked Universities[14]) exposed their public university data and map applications played out to be a common use case for this data.

University of Münster started an Open Data initiative named LODUM[**?**] from which a map application[15] evolved that shows the location of buildings and which departments are located in the selected building, but there is no deeper insight into the buildings. However, there is a publication of the initiative that deals with the issue of indoor navigation [10]. Furthermore an indoor map ontology was designed (see section 3.3.1).

University of Southampton is a further university that exposed spatial information about their campus as LOD. In contrast to the map of the LODUM initiative Southampton‘s map application[16] (see figure 3.5) gives insight into buildings and shows the location of certain rooms with the ability to switch the floor. It shows also computer rooms with their current estimated capacity and opening hours.

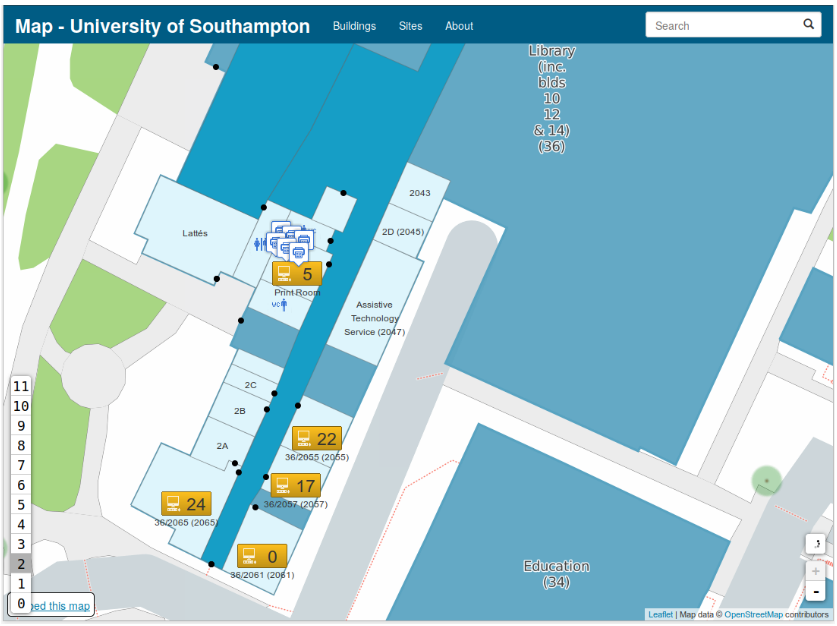

Figure 3.5: Map application of University of Southampton

University of Oxford has also exposed spatial information in form of LOD on their Open Data portal and developed an application named *University Science Area Map* [17]. It

[14] http://linkeduniversities.org
[15] http://app.uni-muenster.de/Karte/
[16] http://maps.southampton.ac.uk/
[17] https://data.ox.ac.uk/explore/science-area/

highlights buildings, where departments of a certain field are located, on a map, but it gives no insight into the buildings.

3.3 Indoor modelling

Section 3.1 outlined ontologies to describe the spatial extent of *features* and topological relationships between them, but those ontologies are not intended for expressing the characteristics of *features* in an indoor environment; like for example that a *feature* is a learning room with certain opening hours or an elevator to get from one floor to another. This section discusses ontologies to model indoor environments that have been discovered in the repository of LOV by searching for terms such as *'room'*, *'building'* and *'toilette'* or explored in datasets of map applications based on LOD (see section 3.2).

3.3.1 LIMAP

3.3.2 Other ontologies

Buildings and Rooms Vocabulary (ROOMS) provides a minimal set of classes and properties to describe the basic structure of a building. It has 6 classes `Building`, `Floor`, `FloorSection`, which is a named (identifiable) section of a floor,`Room` and `Desk` as well as `Site`, which describes an area of land like a campus.

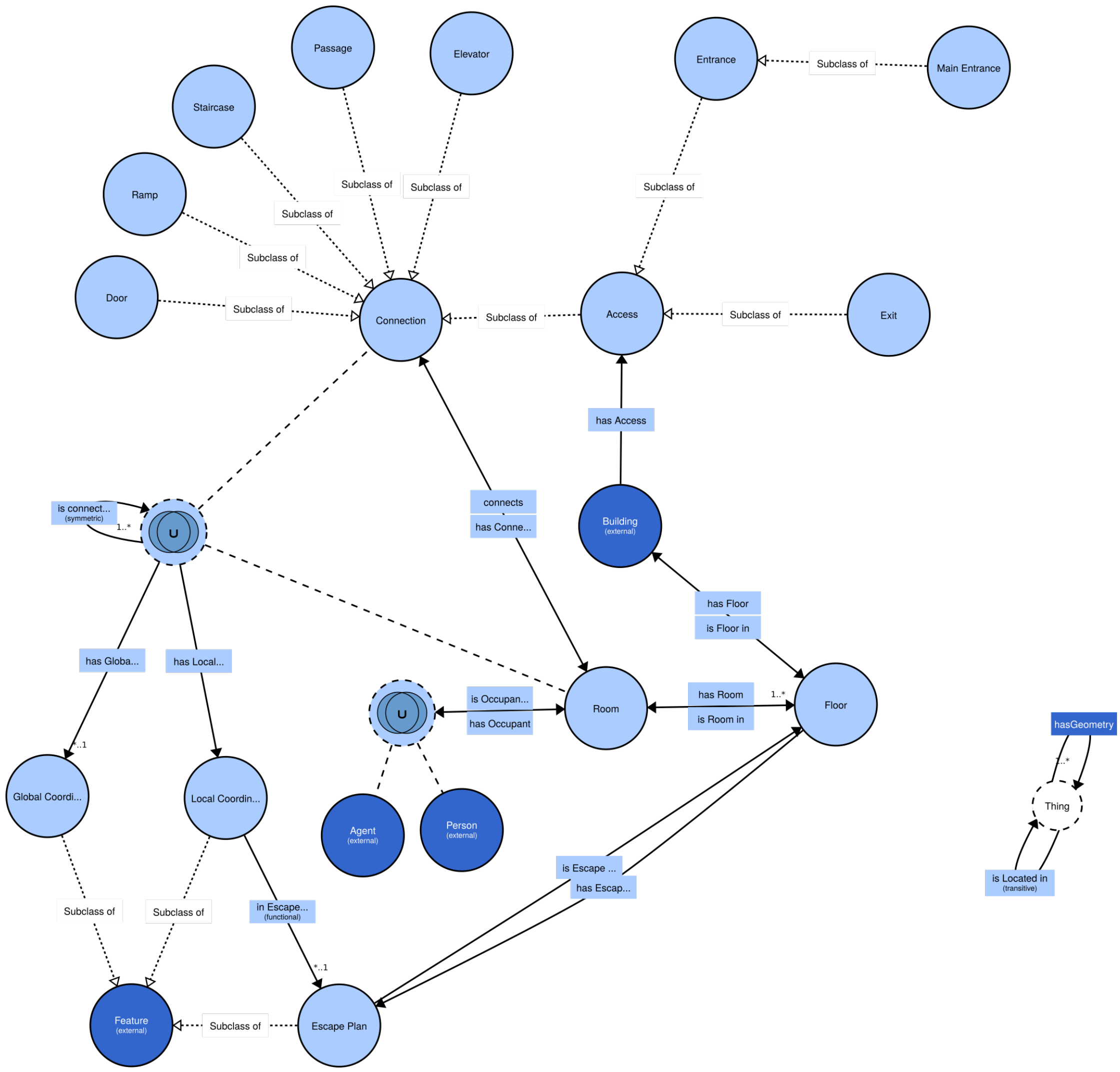

Figure 3.6: Visualization of LODUM Indoor Mapping Ontology (LIMAP)

CHAPTER 4

Solution

This chapter suggests solutions for the given problem description (see 1.2). Section 4.1 is dealing with the problem of transforming unstructured spatial data about the TU Vienna into a machine-readable format. The question of how this structured data shall be described is answered in section 4.2, where a prototype of an ontology is discussed. The subsequent section 4.3 is proposing an architectural prototype of a system for integrating and publishing LD consistent with the LD principles [2]. The final section 4.4 is presenting a map application to show the potential of the generated spatial LOD.

4.1 Data acquisition

Spatial data about the TU Vienna is distributed over multiple sources with different formats and data owners. In order to be eventually transformed into LD, this data must be extracted and prepared for the integration. Figure 4.1 visualizes the different data sources.

The major source for building information is the web site of the facility management unit of the TU Vienna (GUT[1]). This site lists all buildings and an enumeration of their floors in form of a (X)HTML table. It also contains links to the floor plans that are provided as Portable Document Format (PDF) files as well as to a PDF file for each building consisting of a table of all contained rooms with their intended function (e.g. office room, sanitary room). Section 4.1.3 deals with the problem of extracting and transforming tables from PDFs and web pages. The approach for processing floor plans is discussed in section 4.1.2.

The central information system TISS[2] provides a RESTful API that gives access to organizational information and the public address book. For this thesis, this API is useful

[1] `http://www.gut.tuwien.ac.at/`
[2] `https://tiss.tuwien.ac.at/`

for linking persons and organizations to their offices. TISS has also a web page dedicated to presenting the event schedule of certain rooms like lecture halls, but this data is not available over the mentioned RESTful API; only in a human-readable form.

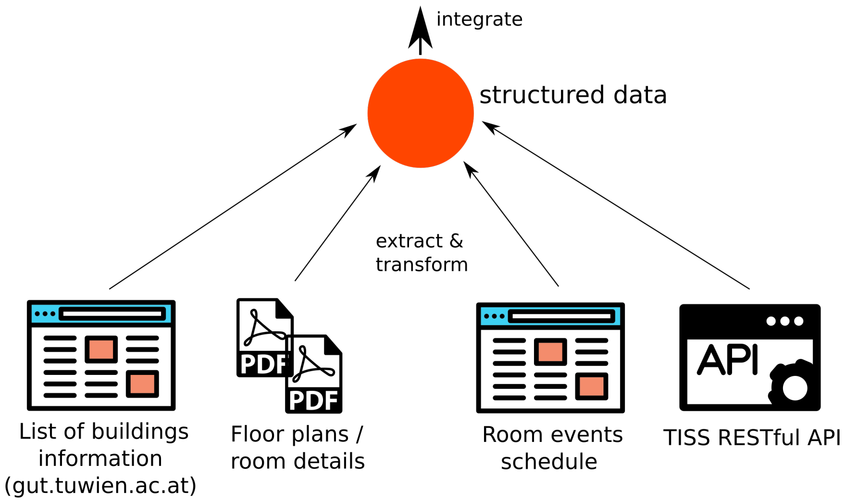

Figure 4.1: Visualization of the data sources

4.1.1 (X)HTML and PDF tables

As mentioned earlier, some of the data is presented to users only in form of (X)HTML tables and must therefore be extracted in order to be transformed into LD. Web scrapers/Web harvester are an useful tool to extract such tables from web pages manually or by using XPATH expressions and to store them in an intermediate representation. For this thesis a chrome extension named 'Web Scraper' [3] was used. This intermediate representation was then exported into a Comma-separated values (CSV) file.

For tables in PDF files a different approach is required and the tool named 'Tabula'[4] is qualified for this kind of problem. It can detect tables automatically and transforms them into an intermediate representation, but this does not always work appropriately. In this case, the user can support the program by selecting the relevant parts manually. This intermediate representation was then also exported into a CSV file.

After the information was extracted from web pages and PDF files and transformed into a structured format, it still was messy data that is not convenient for a further transformation into LD. In this case the tool OpenRefine[5] is quite useful. It provides a set of abilities to manipulate messy, tabular data including an own expression language named GREL. LODRefine and RDF extension are extensions for OpenRefine, which enable the transformation of tabular data into RDF by mapping columns to classes and

[3] `http://webscraper.io/`
[4] `http://tabula.technology/`
[5] `http://openrefine.org/`

relationships between columns to properties. This mapping is then computed for each row.

4.1.2 Floor plans

Floor plans are ideal for describing the internal structure of a building, but they are mostly available in form of images and vector graphics. This representation is fine for humans, but not for machines; because the floor plan should not only be available as single image, but rather be the knowledge base for applications to find paths and locate rooms in the corresponding building.

Google Maps is a map application that also gives insight into certain buildings. It provides a tool [6], where any user can submit floor plans and this floor plan then will be automatically transformed into an internal representation that can thenceforward be looked at in the application (if accepted). The problem is that the user cannot demand access to this internal representation of the plan, although the user may be the owner. Figure 4.3 shows the indoor plan of TU Vienna‘s main library in Google Maps.

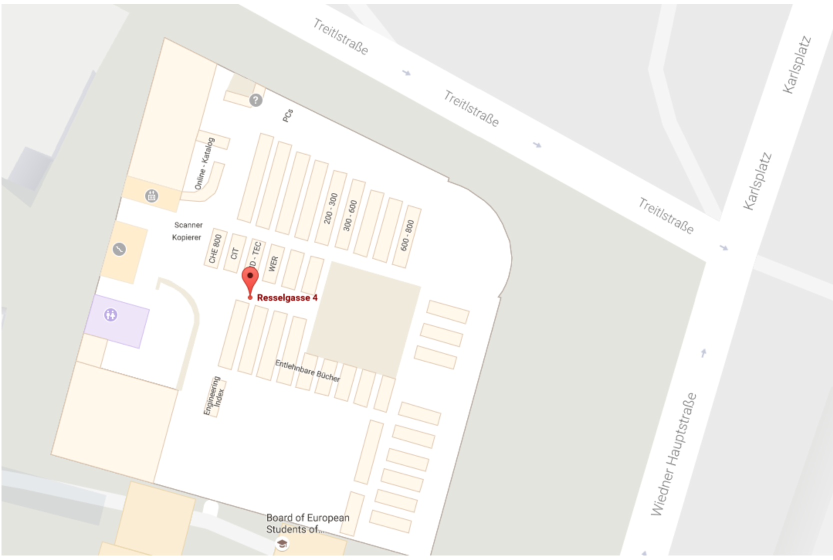

Figure 4.2: Google Maps: Indoor plan of TU Vienna‘s main library

Due to the lack of alternatives, the approach used for this thesis was to manually transform two floor plans of one building into shape files. The tool used for this approach was the Open Source software QGIS[7]. It has two plugins that are useful for the transformation, the 'Open Layers' plug-in to enable the use of OpenStreetMaps among others and the GDAL georeferencing plug-in to position a raster image on a map. Especially the last plug-in is important for positioning the floor plan to the correct position on the map, whereby

[6] https://maps.google.com/floorplans/find
[7] http://www.qgis.org/

the map should already contain the boundary of the building. After georeferencing the floor plan, a shape file of the rooms and other *features* of interest can be created. Each of the created shapes has a table of attributes with an unique ID that can be set by the user. The unique ID of each shape is an Internationalized Resource Identifier (IRI), so that the shape is ready to be transformed into LD. The resulting shape file can then be exported in multiple formats including GeoJSON[8]. For this thesis, a CSV file with the WKT serialization of each shape of the file and the corresponding ID per row showed to be the easiest way to integrate this data. Figure 4.3 shows a shape file of rooms on the basement floor of the informatics institute‘s building at the TU Vienna.

Figure 4.3: Room shapes file of the basement floor of the informatics institute‘s building

4.1.3 XML

TODO: Describe ...

4.2 Prototype of ontology

TODO: Description of the developed ontology with newly created terms and reused ones, and why using GeoSPARQL and what are the disadvantages

4.3 Architectural prototype

TODO: A description of the system, the different layers, integration, linking and web framework, REST API, used technologies

[8]http://geojson.org/

4.3.1 System architecture

TODO: Description of the system architecture

4.3.2 RDF Framework

TODO: Reasoning why using Sesame, not Apache Jena

4.3.3 Linked Data management

TODO: Description of the data management and compliance to best practise

4.3.4 Linking

TODO: Description of strategies for linking with external sources and to enrich the data

4.3.5 Linked Data publishing

TODO: Description of the data publishing and compliance to best practise

4.4 Map application

TODO: A more comprehensive presentation of the user interface for humans to show the targeted use case.

CHAPTER 5

Discussion

5.1 Evaluation

TODO: Enter your text here.

5.2 Further work

TODO: Enter your text here.

5.3 Conclusion

TODO: Enter your text here.

List of Figures

List of Tables

List of Algorithms

Acronyms

GEO Basic Geo (WGS84 long/lat) Vocabulary. 7–9, 11, 12, 21

. 9, 10, 21

GML Geography Markup Language. 11, 12

LD Linked Data. 2, 12

LIMAP LODUM Indoor Mapping Ontology. 14, 21

LOCN ISA Programme Location Core Vocabulary. 12, 21

LOD Linked Open Data. 2, 7, 13, 14

LOV Linked Open Vocabulary. 7, 14

RDF Resource description framework. 5, 9, 11, 29

RDFa RDF in Attributes. 11

ROOMS Buildings and Rooms Vocabulary. 14

SPARQL SPARQL protocol and RDF query language. 5, 11

SPATIAL NeoGeo Spatial Ontology. 9, 11, 21

TU Vienna Vienna University of Technology. 2, 3

WKT Well-known text. 9, 11, 12

Bibliography

[1] Robert Battle and Dave Kolas. Geosparql: enabling a geospatial semantic web. *Semantic Web Journal*, 3(4):355–370, 2011.

[2] Tim Berners-Lee. Linked data-design issues (2006). *URL http://www. w3. org/DesignIssues/LinkedData. html*, 2009.

[3] Diego Berrueta, Jon Phipps, Alistair Miles, Thomas Baker, and Ralph Swick. Best practice recipes for publishing RDF vocabularies. *Working draft, W3C*, 2008.

[4] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 2009.

[5] D Brickley. Basic geo vocabulary. *W3C Semantic Web Interest Group. Available at: http://www.w3.org/2003/01/geo*, 2003. Accessed: 2016-09-26.

[6] Richard Cyganiak and Anja Jentzsch. Linking open data cloud diagram. *LOD Community (http://lod-cloud. net/)*, 12, 2011.

[7] Ramanathan Guha. Introducing schema.org: Search engines come together for a richer web. *Google Official Blog*, 2011.

[8] Kevin Haas, Peter Mika, Paul Tarjan, and Roi Blanco. Enhanced results for web search. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 725–734. ACM, 2011.

[9] Bernadette Hyland, Ghislain Atemezing, and Boris Villazón-Terrazas. Best practices for publishing linked data. *W3C Working Group Note*, 2014.

[10] Nemanja Kostic and Simon Scheider. Automated generation of indoor accessibility information for mobility-impaired individuals. In *AGILE 2015*, pages 235–252. Springer, 2015.

[11] Steffen Lohmann, Stefan Negru, Florian Haag, and Thomas Ertl. VOWL 2: user-oriented visualization of ontologies. In *International Conference on Knowledge Engineering and Knowledge Management*, pages 266–281. Springer, 2014.

[12] Barry Norton, Luis M. Vilches, Alexander De León, John Goodwin, Claus Stadler, Suchith Anand, and Dominic Harries. NeoGeo Vocabulary Specification - Madrid Edition. 2012.

[13] Andrea Perego and Michael Lutz. ISA Programme Location Core Vocabulary. 2015.

[14] Matthew Perry and John Herring. OGC GeoSPARQL-A geographic query language for RDF data. *OGC Implementation Standard. Sept*, 2012.

[15] Research Group: AKWS. LODStats - Basic Statistics of the LOD cloud. Accessed: 2016-09-26.

[16] Jeremy Tandy, Payam Barnaghi, and Linda van den Brink. Spatial Data on the Web Best Practices, January 2016. Accessed: 2016-09-26.

[17] Csaba Veres and Eivind Elseth. Schema.org for the Semantic Web with MaDaME. In *I-SEMANTICS (Posters & Demos)*, pages 11–15. Citeseer, 2013.

[18] Fouad Zablith, Mathieu d'Aquin, Stuart Brown, and Liam Green-Hughes. Consuming linked data within a large educational organization. In *Proceedings of the Second International Conference on Consuming Linked Data-Volume 782*, pages 85–96. CEUR-WS. org, 2011.