Implementing and Developing Geo-visualization techniques showing the large amount of historical and geospatial and temporal data available in the Linked Open Data Cloud on a specific example

Abderrahmen Sdiri

June 27, 2016

Abstract

The project aims to investigate geo-visualization techniques showing a large amount of historical ,geospatial and temporal data available in the Linked Open Data Cloud: http://lod-cloud.net/ and other open data sources . We selected the example of Belgium, we focused on :Population ,external immigration and number of crimes in the 3 regions of Belgium(Flemish,Walloon,Brussels-Capital) and we tried to find a correlation between those datasets to understand the causes of the crimes in Belgium .The data are gathered, processed ,converted to linked data and visualized .The data can be accessed by authorized users over the Internet using an intuitive graphical user interface (GUI) to be implemented. The visualization can be beneficial for understanding the impact of the different factors affecting changes.

Dedication

Acknowledgements

Contents

List of Figures

List of Tables

General introduction

Nowadays, there are many techniques of processing data. Unfortunately, there are many different data formats one can work with. It makes the processing a lotmore difficult. The task becomes even harder when one wants to connect two different datasets in order to benefit from the connection. The connection allows us to get some additional information about entities from each of the standalone datasets. Therefore, a lot of computation time is spent on converting, formatting and transforming data into another form. But transforming datasets into a matching format is not enough. One needs to specify how the data should be linked together.There are many ways of doing that. Starting with implementing the logic into a simple conversion script according to a specific dataset to introducing a more complex metadata description framework for purposes of generic data processing. Since one of the most attractive tasks in this area is to be able to connect any of the datasets available on the Internet, we are interested in the generic description frameworks. We would like to have a tool, which enables us to work with any data on the Internet formatted according to some kind of rules. We would like to link them together, analyze them and visualize them. One of the most used description frameworks is the Resource Description Framework . It is a standard model for data interchange on the Web. It tells us how to describe resources on the Internet in order to allow other people, applications and tools to understand such a description. That gives us a potentialto link any data on the Internet. Based on the framework, a new model named Linked Data was introduced. The model has been brought up to make data interconnecting easier. The result of interconnecting data while utilizing the principles of the Linked Data model and Resource Description Framework is a directed graph. Its vertices represent resources we have information about. The edges stand for relations between such entities. From this point on, it is up to us, how we look at the data. We can either explore them in a plain graph or apply some more semantics and make domain specific visualizations while using ontologies and other advanced techniques.One of the specific domains are statistical data, which are one of the most interesting kind of data. They

are produced and processed by many stakeholders. In the context of Linked Open Data, the most interesting are, of course,governments and scientific groups. But we would like to work with such data in the usual way make tables, charts or more interesting visualizations.

In section 2.1, we look at the semantic web.

Chapter 1

Data science

Introduction

In this chapter, we make an overview of data science by explaining the main idea behind it and its main purpose. We will also present its activities .Besides ,we will focus on data analysis part which is among basic concepts of this work. The benefits of data analysis will be set in the end of this chapter.

1.1 Data science

1.1.1 Overview

Data Science is the art of turning data into actions. This is accomplished through the creation of data products, which provide actionable information without exposing decision makers to the underlying data or analytics (e.g. buy/sell strategies for financial instruments, a set of actions to improve product yield, or steps to improve product marketing). A data product is produced from a statistical analysis. Data products automate complex analysis tasks or use technology to extend the usefulness of informal data model, algorithmic or inference. Performing Data Science requires the extraction of timely, actionable information from diverse data sources to drive data products. Examples of data products include answers to questions such as: Which of my products should I advertise more heavily to increase profit? How can I improve my compliance program, while reducing costs? What manufacturing process change will allow me to build a better product? The key to answering these questions is: understand the data you have and what the data inductively tells you. Data scientists use their data and analytical ability to find and interpret rich data sources, manage large amounts of data despite hard-

ware, software, and bandwidth constraints.They merge also data sources and ensure consistency of datasets, moreover data scientists create visualizations to aid in understanding data.In addition they build mathematical models using the data and present and communicate the data insights/findings. They are often expected to produce answers in days rather than months, work by exploratory analysis and rapid iteration, and to produce and present results with dashboards (displays of current values) rather than papers/reports, as statisticians normally do.

1.1.2 Data Science Activities

Data Science is a complex field. It is difficult, intellectually taxing work, which requires the sophisticated integration of talent, tools and techniques. But we need to cut through the complexity and provide a clear, yet effective way to understand this new world. To do this, we will transform the field Data Science into a set of simplified activities a, The Four Key Activities of a Data Science Endeavor.

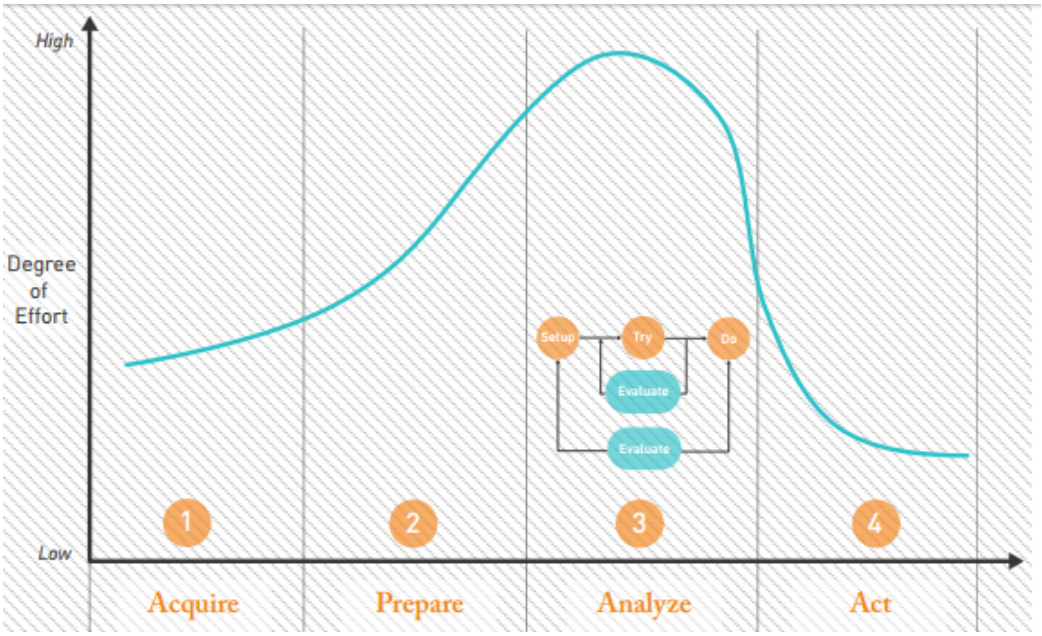

Figure 1.1: Data science activities

1. **To acquire:** All analysis starts with access to data, and for the Data Scientist this axiom holds true. But there are some signifiant diffrences particularly with respect to the question of who stores, maintains and

owns the data in an organization.But before we go there, lets look at what is changing. Traditionally,rigid data silos artifiially defie the data to be acquired. Stated another way, the silos create a fiter that lets in a very small amount of data and ignores the rest. Thse fitered processes give us an artifiial view of the world based on the surviving data, rather than one that shows full reality and meaning. Without a broad and expansive dataset, we can never immerse ourselves in the diversity of the data. We instead make decisions based on limited and constrained information. Eliminating the need for silos gives us access to all the data at once including data from multiple outside sources. It embraces the reality that diversity is good and complexity is okay. Ths mindset creates a completely diffrent way of thinking about data in an organization by giving it a new and diffrentiated role. Data represents a signifiant new profi and mission-enhancement opportunity for organizations.
But as mentioned earlier, this fist activity is heavily dependent upon the situation and circumstances. We cant leave you with anything more than general guidance to help ensure maximum value:

- Look inside fist:What data do you have current access to that you are not using? Ths is in large part the data being left behind by the fitering process, and may be incredibly valuable
- Remove the format constraints:Stop limiting your data acquisition mindset to the realm of structured databases. Instead, think about unstructured and semi-structured data as viable sources
- Figure out whats missing:Ask yourself what data would make a big diffrence to your processes if you had access to it,then go find it!
- Embrace diversity: Try to engage and connect to publicly available sources of data that may have relevance to your domain area

2. **To prepare:** Once you have the data, you need to prepare it for analysis. Organizations often make decisions based on inexact data. Data stovepipes mean that organizations may have blind spots. Thy are not able to see the whole picture and fail to look at their data and challenges holistically. Th end result is that valuable information is withheld from decision makers. Research has shown almost 33decisions are made without good data or information. When Data Scientists are able to explore and analyze all the data, new opportunities arise for analysis and data-driven decision making. Th insights gained from these new

opportunities will signifiantly change the course of action and decisions within an organization. Gaining access to an organizations complete repository of data, however, requires preparation.
Our experience shows time and time again that the best tool for Data Scientists to prepare for analysis is a lake specifially, the Data Lake.This is a new approach to collecting, storing and integrating data that helps organizations maximize the utility of their data. Instead of storing information in discrete data structures, the Data Lake consolidates an organizations complete repository of data in a single, large view. It eliminates the expensive and cumbersome data-preparation process, known as Extract/Transform/Load (ETL), necessary with data silos. Th entire body of information in the Data Lake is available for every inquiry and all at once..

3. **To analyse:** Th Analyze activity requires the greatest effort of all the activities in a Data Science endeavor. Th Data Scientist actually builds the analytics that create value from data. Analytics in this context is an iterative application of specialized and scalable computational resources and tools to provide relevant insights from exponentially growing data. Ths type of analysis enables real-time understanding of risks and opportunities by evaluating situational, operational and behavioral data. With the totality of data fully accessible in the Data Lake, organizations can use analytics to fid the kinds of connections and patterns that point to promising opportunities. Ths high-speed analytic connection is done within the Data Lake, as opposed to older style sampling methods that could only make use of a narrow slice of the data. In order to understand what was in the lake, you had to bring the data out and study it. Now you can dive into the lake, bringing your analytics to the data. The figure, Analytic Connection in the Data Lake, highlights the concept of diving into the Data Lake to discover new connections and patterns.
Data Scientists work across the spectrum of analytic goals Describe, Discover, Predict and Advise. Th maturity of an analytic capability determines the analytic goals encompassed. Many variables play key roles in determining the diffilty and suitability of each goal for an organization. Some of these variables are the size and budget of an organization and the type of data products needed by the decision makers. A detailed discussion on analytic maturity can be found in Data Science Maturity within an Organization. In addition to consuming the greatest effrt, the Analyze activity is by far the most complex. Th tradecraft of Data Science is an art. While we cannot teach you how to

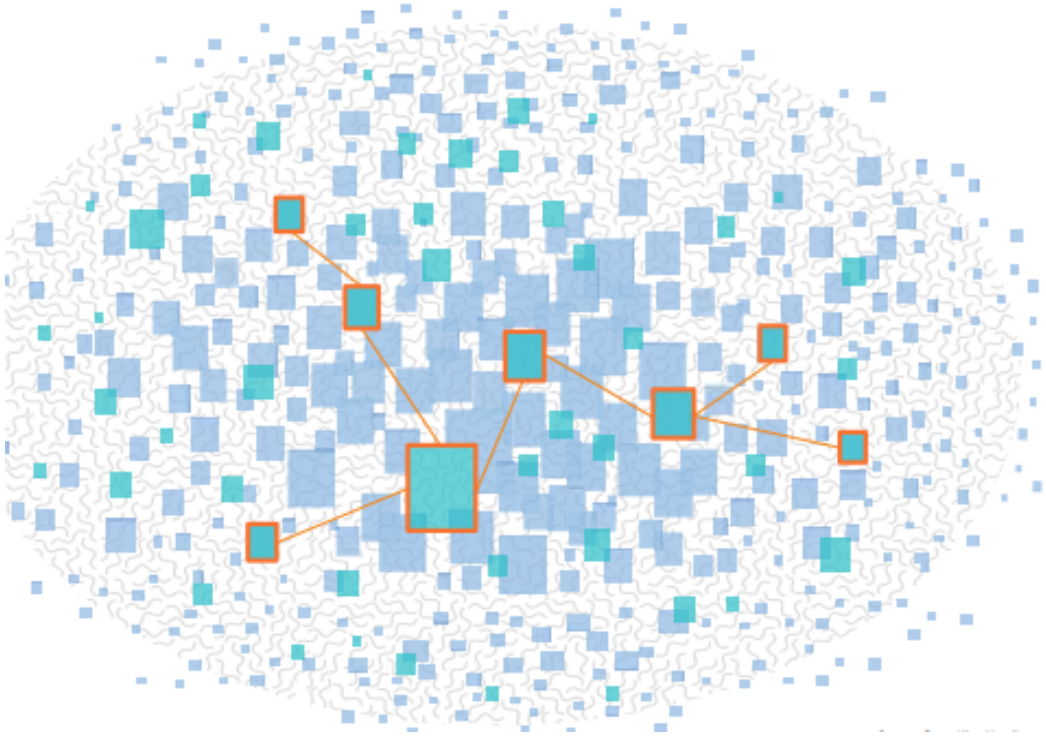

Figure 1.2: Analytic Connection in the Data Lake

be an artist, we can share foundational tools and techniques that can help you be successful. Th entirety of Take Of the Training Wheels is dedicated to sharing insights we have learned over time while serving countless clients. This includes descriptions of a Data Science product lifecycle and the Fractal Analytic Model (FAM). The Analytic Selection Process and accompanying Guide to Analytic Selection provide key insights into one of the most challenging tasks in all of Data Science selecting the right technique for the job

4. **To Act:** The ability to make use of the analysis is critical. It is also very situational. Like the Acquire activity, the best we can hope for is to provide some guiding principles to help you frame the output for maximum impact. Here are some key points to keep in mind when presenting your results:

   (a) The finding must make sense with relatively little up-front training or preparation on the part of the decision maker.

   (b) The findings must make the most meaningful patterns, trends and

exceptions easy to see and interpret.

(c) Every effort must be made to encode quantitative data accurately so the decision maker can accurately interpret and compare the data.

(d) The logic used to arrive at the fiding must be clear and compelling as well as traceable back through the data.

(e) Th fidings must answer real business questions.

1.2 Data analysis

1.2.1 Definition

The term data analysis refers to the process by which large amounts of raw data is reviewed in order to determine conclusions based on that data. It is the process of bringing order, structure and meaning to the mass of collected data. The data is often unorganized, and may come from different sources.Analysis of data is a process of inspecting, cleaning, transforming, and modeling data with the goal of discovering useful information, suggesting conclusions, and supporting decision-making. Data analysis has multiple facets and approaches, encompassing diverse techniques under a variety of names, in different business, science, and social science domains.The nature of data analysis varies, and correlates to the type of data being examined. For example, a business may concentrate on things such as determining employee performance, sales performance by department or sales person, etc. An economist, however, might look for identifiable patterns that explain the spending habits of various consumers.

1.2.2 Types of data analysis

There are many different types of data analysis, all geared towards the nature of the data being analyzed. Generally speaking there are two broad categories: quantitative analysis and qualitative analysis

Qualitative Analysis

Qualitative analysis deals with the analysis of data that is categorical in nature. In other words, data is not described through numerical values, but rather by some sort of descriptive context such as text. Data can be gathered by many methods such as interviews, videos and audio recordings, field notes, etc. Once data is gathered it then needs to be interpreted. Often times

Qualitative Data	Quantitative Data
Data is observed	Data is measured
Involves descriptions	Involves numbers
Emphasis is on quality	Emphasis is on quantity
Examples are color, smell, taste	Examples are volume, weight, etc.

Table 1.1: Comparison of Qualitative and Quantitative Analysis

this involves coding, which refers to the grouping of data into identifiable themes. Themes are then given a unique label, and each label can then be quickly grouped and contrasted to each other.Of course data must also be interpreted. Interpretation can be a part of the coding process, but this is not always the case.Qualitative analysis can be summarized by three basic principles (Seidel, 1998): Notice things, Collect things, Think about things

Quantitative Analysis

Quantitative analysis refers to the process by which numerical data is analyzed, and often involves descriptive statistics such as mean, media, standard deviation, etc. An in-depth discussion of quantitative analysis is beyond the scope of this article. Generally speaking, however, the following are often involved with quantitative analysis: Statistical models, Analysis of variables, Data dispersion, Analysis of relationships between variables, Contingence and correlation, Regression analysis, Statistical significance, Precision Error limits

1.2.3 The process of data analysis

Analysis refers to breaking a whole into its separate components for individual examination. Data analysis is a process for obtaining raw data and converting it into information useful for decision-making by users. Data is collected and analyzed to answer questions, test hypotheses or disprove theories. Statistician John Tukey defined data analysis in 1961 as: "Procedures for analyzing data, techniques for interpreting the results of such procedures, ways of planning the gathering of data to make its analysis easier, more precise or more accurate, and all the machinery and results of (mathematical) statistics which apply to analyzing data. " There are several phases that can be distinguished, described below. The phases are iterative, in that feedback from later phases may result in additional work in earlier phases.

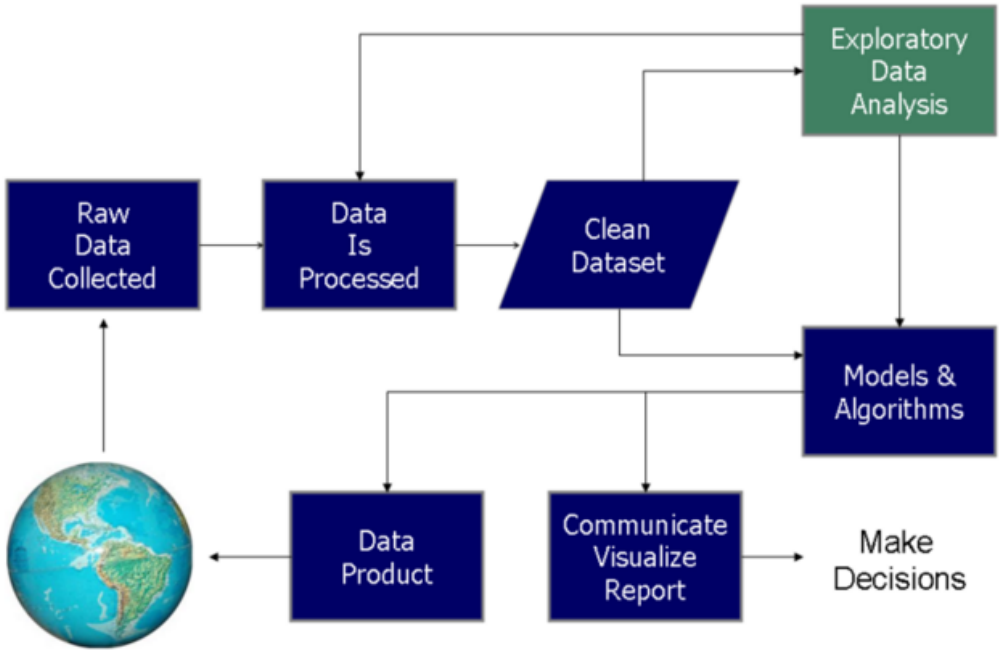

Figure 1.3: The process of data science

Data requirements

The data necessary as inputs to the analysis are specified based upon the requirements of those directing the analysis or customers who will use the finished product of the analysis. The general type of entity upon which the data will be collected is referred to as an experimental unit (e.g. a person or population of people). Specific variables regarding a population (e.g., age and income) may be specified and obtained. Data may be numerical or categorical (i.e., a text label for numbers).

Data collection

Data is collected from a variety of sources. The requirements may be communicated by analysts to custodians of the data, such as information technology personnel within an organization. The data may also be collected from sensors in the environment, such as traffic cameras, satellites, recording devices, etc. It may also be obtained through interviews, downloads from online sources, or reading documentation.

Data processing

The phases of the intelligence cycle used to convert raw information into actionable intelligence or knowledge are conceptually similar to the phases in data analysis. Data initially obtained must be processed or organized for analysis. For instance, this may involve placing data into rows and columns in a table format for further analysis, such as within a spreadsheet or statistical software.

Data cleaning

Once processed and organized, the data may be incomplete, contain duplicates, or contain errors. The need for data cleaning will arise from problems in the way that data is entered and stored. Data cleaning is the process of preventing and correcting these errors. Common tasks include record matching, deduplication, and column segmentation. Such data problems can also be identified through a variety of analytical techniques. For example, with financial information, the totals for particular variables may be compared against separately published numbers believed to be reliable. Unusual amounts above or below pre-determined thresholds may also be reviewed. There are several types of data cleaning that depend on the type of data. Quantitative data methods for outlier detection can be used to get rid of likely incorrectly entered data. Textual data spellcheckers can be used to lessen the amount of mistyped words, but it is harder to tell if the words themselves are correct.

Exploratory data analysis

Once the data is cleaned, it can be analyzed. Analysts may apply a variety of techniques referred to as exploratory data analysis to begin understanding the messages contained in the data.The process of exploration may result in additional data cleaning or additional requests for data, so these activities may be iterative in nature. Descriptive statistics such as the average or median may be generated to help understand the data. Data visualization may also be used to examine the data in graphical format, to obtain additional insight regarding the messages within the data.

Modeling and algorithms

Mathematical formulas or models called algorithms may be applied to the data to identify relationships among the variables, such as correlation or causation. In general terms, models may be developed to evaluate a particular

variable in the data based on other variable(s) in the data, with some residual error depending on model accuracy (i.e., Data = Model + Error). Inferential statistics includes techniques to measure relationships between particular variables. For example, regression analysis may be used to model whether a change in advertising (independent variable X) explains the variation in sales (dependent variable Y). In mathematical terms, Y (sales) is a function of X (advertising). It may be described as Y = aX + b + error, where the model is designed such that a and b minimize the error when the model predicts Y for a given range of values of X. Analysts may attempt to build models that are descriptive of the data to simplify analysis and communicate results.

Data product

A data product is a computer application that takes data inputs and generates outputs, feeding them back into the environment. It may be based on a model or algorithm. An example is an application that analyzes data about customer purchasing history and recommends other purchases the customer might enjoy.

Communication

Data visualization is used to understand the results of a data analysis. Once the data is analyzed, it may be reported in many formats to the users of the analysis to support their requirements. The users may have feedback, which results in additional analysis. As such, much of the analytical cycle is iterative.When determining how to communicate the results, the analyst may consider data visualization techniques to help clearly and efficiently communicate the message to the audience. Data visualization uses information displays such as tables and charts to help communicate key messages contained in the data. Tables are helpful to a user who might lookup specific numbers, while charts (e.g., bar charts or line charts) may help explain the quantitative messages contained in the data.

1.2.4 Benefits of Data Analysis

The main benefits of data analysis are rather self-evident. How can someone improve their processes and identify problematic issues if they are not willing to look at the data? The answer, of course, is that they cannot make reliable improvements without data analysis. The key word here is reliable! Most people have a general idea about possible changes that should or could improve their processes. However, when it comes to these sorts of changes

there is the inherent risk that the change does not have the desired result. There can also be unexpected consequences that impact some other aspect of that organization in a negative manner. Having said that, the following are just some of the benefits of proper data analysis:

- Allows for the identification of important (and often mission-critical) trends
- Helps businesses identify performance problems that require some sort of action Can be viewed in a visual manner, which leads to faster and better decisions
- Better awareness regarding the habits of potential customers
- It can provide a company with an edge over their competitors

The process of evaluating data using analytical and logical reasoning to examine each component of the data provided. This form of analysis is just one of the many steps that must be completed when conducting a research experiment. Data from various sources is gathered, reviewed, and then analyzed to form some sort of finding or conclusion. There are a variety of specific data analysis method, some of which include data mining, text analytics, business intelligence, and data visualizations.

Conclusion

According to this chapter, we can classify our work as a quantitative data analysis project in which we aim to get some conclusion based on the data gathered, processed and visualized ,so lets treat now the data part in the next chapter called semantic technologies.

Chapter 2

Semantic technologies

Introduction

This chapter contains the basis of the semantic web .It presents also the well known Resource Description Framework and its query language SPARQ.

2.1 Semantic web

2.1.1 Overview

The current web represents information using natural languages, graphics and multimedia objects which can be easily understood and processed by an average user. Some tasks on the web require combining data on the web from different sources e.g. travel and hotel information may come from different web sites when booking for a trip. Humans can merge this information and process them quite easily. However, machines can not combine such information and process it. Most of the Webs content today is designed for humans to read, not for computer programs to manipulate meaningfully. Computers can adeptly parse Web pages for layout and routine processing here a header, there a link to another page but in general, computers have no reliable way to process the semantics. The Semantic Web will bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users.The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation. The Semantic Web is a mesh of information linked up in such a way as to be easily processable by machines, on a global scale. You can think of it as being an efficient

way of representing data on the World Wide Web, or as a globally linked database. The Semantic Web was thought up by Tim Berners-Lee, inventor of the WWW, URIs, HTTP, and HTML. There is a dedicated team of people at the World Wide Web consortium (W3C) working to improve, extend and standardize the system, and many languages, publications, tools and so on have already been developed. However, Semantic Web technologies are still very much in their infancies, and although the future of the project in general appears to be bright, there seems to be little consensus about the likely direction and characteristics of the early Semantic Web In addition to the classic Web of documents W3C is helping to build a technology stack to support a Web of data, the sort of data you find in databases. The ultimate goal of the Web of data is to enable computers to do more useful work and to develop systems that can support trusted interactions over the network. The term Semantic Web refers to W3Cs vision of the Web of linked data. Semantic Web technologies enable people to create data stores on the Web, build vocabularies, and write rules for handling data. Linked data are empowered by technologies such as RDF, SPARQL, OWL, and SKOS.

2.1.2 Semantic web architecture

The first layer, URI and Unicode, follows the important features of the existing WWW. Unicode is a standard of encoding international character sets and it allows that all human languages can be used (written and read) on the web using one standardized form. Uniform Resource Identifier (URI) is a string of a standardized form that allows to uniquely identify resources (e.g., documents). A subset of URI is Uniform Resource Locator (URL), which contains access mechanism and a (network) location of a document - such as http://www.example.org/. Another subset of URI is URN that allows to identify a resource without implying its location and means of dereferencing it - an example is urn:isbn:0-123-45678-9. The usage of URI is important for a distributed internet system as it provides understandable identification of all resources. An international variant to URI is Internationalized Resource Identifier (IRI) that allows usage of Unicode characters in identifier and for which a mapping to URI is defined. In the rest of this text, whenever URI is used, IRI can be used as well as a more general concept.Extensible Markup Language (XML) layer with XML namespace and XML schema definitions makes sure that there is a common syntax used in the semantic web. XML is a general purpose markup language for documents containing structured information. A XML document contains elements that can be nested and that may have attributes and content. XML namespaces allow to specify different markup vocabularies in one XML document. XML schema serves

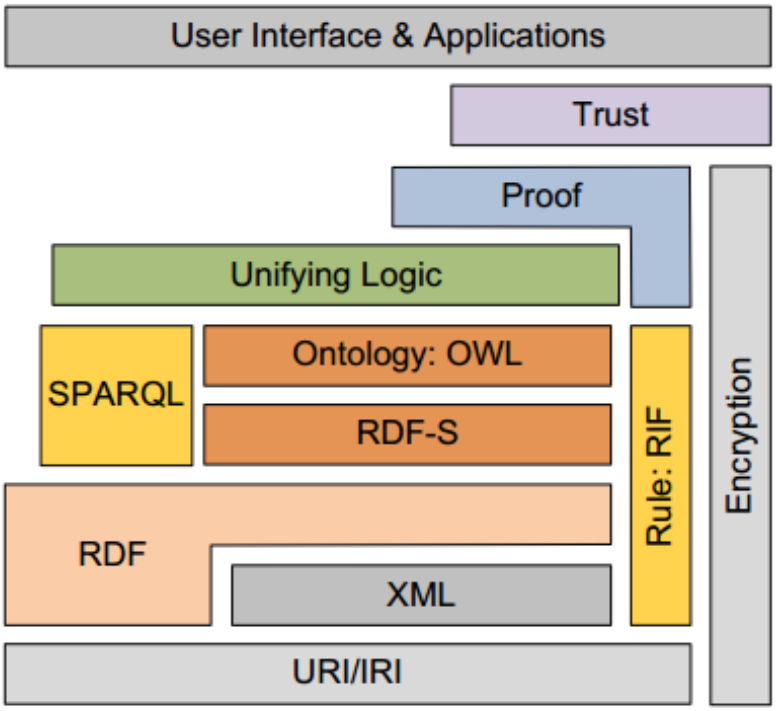

Figure 2.1: Semantic web architecture

for expressing schema of a particular set of XML documents.A core data representation format for semantic web is Resource Description Framework (RDF). RDF is a framework for representing information about resources in a graph form. It was primarily intended for representing metadata about WWW resources, such as the title, author, and modification date of a Web page, but it can be used for storing any other data. It is based on triples subject-predicate-object that form graph of data. All data in the semantic web use RDF as the primary representation language. The normative syntax for serializing RDF is XML in the RDF/XML form. Formal semantics of RDF is defined as well.RDF itself serves as a description of a graph formed by triples. Anyone can define vocabulary of terms used for more detailed description. To allow standardized description of taxonomies and other ontological constructs, a RDF Schema (RDFS) was created together with its formal semantics within RDF. RDFS can be used to describe taxonomies of classes and properties and use them to create lightweight ontologies. More detailed ontologies can be created with Web Ontology Language OWL. The OWL is a language derived from description logics, and offers more constructs over RDFS. It is syntactically embedded into RDF, so like RDFS, it

provides additional standardized vocabulary. OWL comes in three species - OWL Lite for taxonomies and simple constrains, OWL DL for full description logic support, and OWL Full for maximum expressiveness and syntactic freedom of RDF. Since OWL is based on description logic, it is not surprising that a formal semantics is defined for this language.RDFS and OWL have semantics defined and this semantics can be used for reasoning within ontologies and knowledge bases described using these languages. To provide rules beyond the constructs available from these languages, rule languages are being standardized for the semantic web as well. Two standards are emerging - RIF and SWRL.For querying RDF data as well as RDFS and OWL ontologies with knowledge bases, a Simple Protocol and RDF Query Language (SPARQL) is available. SPARQL is SQL-like language, but uses RDF triples and resources for both matching part of the query and for returning results of the query. Since both RDFS and OWL are built on RDF, SPARQL can be used for querying ontologies and knowledge bases directly as well. Note that SPARQL is not only query language, it is also a protocol for accessing RDF data.

2.2 Resource Description framework

The Resource Description Framework (RDF) is a language for representing information about resources in the World Wide Web. A resource is a physical or virtual entity, such as a personor an IP packet. RDF describes those resources in a subject-predicate-object structure.

2.2.1 Concept

RDF represents information by means of statements in a Subject-Predicate-Object structure:

- Subject: a resource that is described by the statement
- Predicate: a property of the resource that is described
- Object: the value of the property of the resource that is described

Take for example the statement Abderrahmen's emailAdress is abderrahmen.sdiri@supcom.tn. The parts are:

- Subject:Abderrahmen
- Predicate:emailAdress

- Object: abderrahmen.sdiri@supcom.tn

Since RDF is meant to be machine-processable, evey resource has to be unique to avoid confusion. The Web offers URI references to deal with this problem. Subjects and predicates are resources, and thus are to be represented by a URI reference. Objects can be resources, though they also can be a literal, which is a non-decomposable object, like a string or a number

2.2.2 Representation

RDF has in fact an inherent graph based structure, which can be serialized using turte, RDF/XML and others. The next paragraphs introduce representations for a group of statements. Forsome representations, an example is given for Abderrahmen is called Abderrahmen S and is 24 years old. Ahmed is his friend. The FOAF ontology is used to describe these statements. Note that since URIs can be long, they can be shortened in a more clear notation, using prefixes. Therefore a URI reference is split up in a namespace and local name. This namespace is represented by a prefix. The notation is shortened as `prefix:local_name`. For instance,`http://xmlns.com/foaf/0.1/name` is equivalent to `foaf:name`, using foaf as a prefix for `http://xmlns.com/foaf/0.1/`.

2.2.3 Graph model notation

RDF statements lend themselves easily to be represented as a graph. Subjects and objects are the equivalent of the nodes and predicates are labeled edges. Conceptually this means that a subject is connected to an object by means of a predicate. The following graph is the representation of the earlier given example. Note that every resource, used as a subject or object, is unique; per resource only one node exists.

2.2.4 RDF Turtle

Turtle (Terse RDF Triple Language) is a format for expressing data in the Resource Description Framework (RDF) data model with a syntax similar to SPARQL. RDF, in turn, represents information using "triples", each of which consists of a subject, a predicate, and an object. Each of those items is expressed as a Web URI.Turtle provides a way to group three URIs to make a triple, and provides ways to abbreviate such information, for example by factoring out common portions of URIs. For example:
`<http://example.org/person/Abderrahmen> <http://example.org/relation/student> <http://example.org/engineeringSchools/SUP'COM>` .

2.2.5 RDF XML

RDF uses XML as a structure to guarantee the machine-pro cess ability and interchangeability.XML is a markup language that allows creating a custom do cument format. Hence, RDF/XMLis a representation for state ments.

2.2.6 RDFS

RDF is used to structure and represent statements about resources. It does not specify what the meaning of those statements is. In other words, the statements lack semantics. RDFS (RDF Schema) deals with that problem and offers the possibility to create vocabularies, also written in RDF. Therefore RDFS uses the notion of classes and properties. The vocabularies describe the semantics of these classes and properties. By using them in RDF, reuse of semantics is possible. RDFS has many other features, including domain and range definition for a property. The next figure is an example of a simple vocabulary of an animal. Pet has two subclasses Mammal and Bird. Dog is a subclass of Mammal. Due to the subclassing, all mammels (including dogs) and birds also have the property name, which is defined for Pet and is a literal Therefore, RDF Schema language, provides the syntax for

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
 <rdfs:Class rdf:ID="Pet" />
 <rdfs:Class rdf:ID="Mammal">
  <rdfs:subClassOf rdf:resource="#Pet" />
 </rdfs:Class>
 <rdfs:Class rdf:ID="Bird">
  <rdfs:subClassOf rdf:resource="#Pet" />
 </rdfs:Class>
 <rdfs:Class rdf:ID="Dog">
  <rdfs:subClassOf rdf:resource="#Mammal" />
 </rdfs:Class>
 <rdf:Property rdf:ID="name">
  <rdfs:domain rdf:resource="#Pet" />
  <rdfs:range rdf:resource="rdfs;Literal" />
 </rdf:Property>
</rdf:RDF>
```

Figure 2.2: RDF Schema Example

defining the general RDF vocabulary as well as domain-specific vocabularies. RDFS is built on top of RDF, so that RDFS data is also valid RDF data. RDFS introduces the concepts of classes and their properties. In particular, RDFS allows for specifying classification and generalisation hierarchies for both metadata properties and values. Using RDFS it is possible to distinguish between RDF instance data and its schema.

2.3 OWL

OWL (Web Ontology Language) is, like RDFS, a language to define vocabularies. It facilitates greater machine interpretability by providing more vocabulary, thus is more expressive thanRDFS. OWL adds, among others, cardinality, relations between classes and more properties. OWL comes in three flavours, which are ordered by increasing expressiveness:

- OWL Lite only defines a classification hierarchy and simple cardinality constraints.
- OWL DL supports maximum expressiveness and guarantees computational completeness (computable). Every computation also is decidable (finishes in a finite time).
- OWL Full handles every aspect of OWL DL without any computational guarantees.

While it is possible to define your own vocabularies, reusing existing ones facilitates understandability of your data. A commmonly used vocabulary to describe persons is Friend-Of-A-Friend (FOAF).
Thus ,ontologies are used to capture knowledge about some domain of interest. An ontology describes the concepts in the domain and also the relationships that hold between those concepts. Different ontology languages provide different facilities. The most recent development in standard ontology languages is OWL from the World Wide Web Consortium (W3C) . Like Protg, OWL makes it possible to describe concepts but it also provides new facilities. It has a richer set of operators - e.g. intersection, union and negation. It is based on a different logical model which makes it possible for concepts to be defined as well as described. Complex concepts can therefore be built up in definitions out of simpler concepts. Furthermore, the logical model allows the use of a reasoner which can check whether or not all of the statements and definitions in the ontology are mutually consistent and can also recognise which concepts fit under which definitions. The reasoner can therefore help to maintain the hierarchy correctly. This is particularly useful when dealing with cases where classes can have more than one parent.

2.4 SPARQL and SPARQL Endpoints

SPARQL (SPARQL Protocol and RDF Query Language) is a query language for RDF. It also anchors the protocol which clients use to access a SPARQL endpoint.

2.4.1 Syntax

In this section, the basic syntax of the SELECT feature is highlighted. However, many other query forms are supported by SPARQL (CONSTRUCT, ASK and DESCRIBE), but they are not described in detail, as they are not used in this project. SPARQL does not support commands that alter data, such as UPDATE, etc.
The next figure wants to retrieve the names of all persons in the dataset. The SELECT keyword asks the endpoint to bind every matching RDF resource or literal to the defined variables. The WHERE clausule restricts the possible bindings by making a subset of the dataset, based on the statements that are specified. Variables are represented by the prefix ?. Literals are placed between double quotes. The PREFIX command implements the concept of shortened notation of resources. The query wants to retrieve the names of all persons in the dataset that know another person. The name of that person is also returned.

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT ?name
WHERE {
 ?person  rdf:type   foaf:Person  .
 ?person  foaf:name  ?name        .
}
```

Figure 2.3: SPARQL SELECT query

2.4.2 Working of SPARQL

The results of a SPARQL query depend on the conditions specified in the WHERE-clause, since they filter the queried RDF dataset. As stated before, RDF statements have an inherent graph based structure. This also applies for a WHERE-clause, as that is a collection of statements. Each graph contains one or more patterns. There are various types of such graph patterns, proportional to the complexity of the WHERE-clause. . In order to filter a dataset, SPARQL uses the notion of graph patterns to apply graph pattern matching between the datasets graph and the graph of its WHEREclause. This approach is completely different from SQLs, which acts in a procedural way for querying relational datasets, without an implicit join feature. That is, joins have to be specified explicitly in order to combine data, whereas in SPARQL, this is already provided by the structure of RDF

2.4.3 SPARQL endpoints

SPARQL endpoints are web applications that offer an interface to an RDF dataset in the form of a plain HTTP GET-request. The SPARQL protocol is hidden for the end users. There exist many SPARQL endpoints on the Web 6. Users can set up their own endpoint using some software like OpenLink Virtuoso or Apache Jena Fuseki. In this project, Fuseki is used to set up multiple sources with distributed data

Conclusion

Chapter 3

Linking open data project

Introduction

3.1 Open Data

Even though the term Open Data is currently in frequent use, there is no commonly agreed definition. We will use the one provided by the Open Knowledge Definition (OKD) project. OKD considers data as any kind of content from sonnets to statistics, genes to geodata. According to the OKD definition, data is open if anyone is free to use, reuse, and redistribute it - subject only, at most, to the requirement to attribute and sharealike. This definition considers three aspects of data openness: social, technological and legal. Social openness means that the data must be accessible as a whole, and not only few items of it at a time (e.g., by downloading). By technological openness OKD considers absence of any technological obstacles to access and reuse the data (e.g., no,access control and open formats). Legal openness is established by open data licenses.

Open Data licensing Open licenses meet the requirements of Open Data and grant permissions to access, reuse and redistribute data with few or no restrictions. In order to enable people to use the data on the Web on a secure legal basis, one needs to explicitly state which license applies to your data. It is important to apply open data licenses simply for the sake of clarity. Without a license it is not clear if the data can be used, reused and distributed by others. Examples of open licenses include Public Domain Dedication and License (PDDL) and Attribution License by the Open Data Commons project, the GNU Free Documentation License and the licenses

prepared by the Creative Commons Attribution project, such as Creative Commons AttributionShare-Alike (cc-by-sa).
Open data sets There are various interesting open data sets available on the Web. A well-known example of open data is Wikipedia. Most of the Wikipedias text and many of its images are under open licenses. Other examples of open data are Wikibooks, Geonames, MusicBrainz, WordNet and the DBLP bibliography. Google Maps is an example of data that is not open, since the geodata is currently proprietary (copyrighted or protected by DB rights).In general, Open Data can come from anywhere. One of the biggest source of Open Data is the government domain. Open Government Data is a global movement of governments starting to open their information from public sector. The pioneers were the governments of the U.S. and the U.K, and many more governments have already joined the movement, including Australia, Netherlands, Spain, Austria, Denmark

3.2 Linked Data

The term Linked Data was coined by Tim Berners-Lee in 2006 in his design note . He outlined a set of recommendations, referred to as Linked Data principles, on how to complement the current Web of human-oriented documents with a Web of machineenabled data. Tim Berners-Lee proposed to apply the same ideas that are successfully used for making the current Web, to publish and interlink data in such a way that machines can also process it and extract its meaning, i.e., build a Web of Data.

3.2.1 The Linked Data principles

The Linked Data principles in its original reading are as follows :

1. Use URIs as names for things
2. Use HTTP URIs so that people can look up those names
3. When someone looks up a URI, provide useful information, using the standards (RDF,SPARQL)
4. Include links to other URIs. so that they can discover more things

The Linked Data principles in a nutshell The classic Web provides humans with data. It can be any kind of data, e.g., air temperature, somebodys

personal profile, official government reports, etc. The data is represented in the form of documents (e.g., HTML Web pages). Documents can be processed by machines that can understand the structure of the documents (e.g., paragraphs, titles, tables, etc.) and present them in a more convenient way for humans to read and comprehend the data carried out by the documents. However, the meaning of the data is mostly given as a plain text and hidden from the machines. It is hard for software applications to extract semantics from HTML pages. To allow machines to understand the meaning of the data, we need to be able to describe not only the structure of the documents, but the data itself. At its most basic, data is made up of any kind of things that exists in the world, i.e., real-world objects (e.g., people, countries, building, etc.) or abstract concepts (e.g., air temperature, the fact of knowing somebody, etc.). Thus, the Web of Data extends the scope of the traditional Web from documents to encompass real-world objects and abstract concepts. The first and second Linked Data principles define a mechanism to name any thing that exists in the world using the existing Web standards, such as URI and HTTP , and make them accessible on the Web. For example, by using the Linked Data principles one can give a name the concept of city of Lisbon and publish it on the Web. Note that the city of Lisbon and its homepage are not the same concept. For this reason they must be named with two different URIs.
The conventional Web has its standard way to describe documents on the Web, i.e., HTML is used to create Web pages. The third Linked Data principle recommends to use the Resource Description Framework (RDF) to describe things in the world in a machine-redable manner . In RDF one can provide descriptions of real-world concepts in the form of sentences. For example, one may describe the Lisbon city as follows: Lisbon is the capital of Portugal; Lisbon has population 545,245 people, etc. Finally, links are an integral part of a Web. Documents on the classic Web are connected by means of the hypertext links. Similarly, the fourth Linked Data principle claims to connect data on the Web by setting links to data from other data sources. Links on the Web of Data are defined using RDF and referred to as RDF links. Unlike the hypertext links, the links on the Web of Data not only connect two pieces of data together, they also provide semantics for this connection. The RDF links look like sentences as well, just involve concepts that were defined by different people. For example, if you define the concept of the city of Lisbon and discover that somebody else in another dataset defined the concept of Portugal, then the following link can be set Lisbon is the capital of Portugal (where Portugal is the concept from that external dataset). Thus, you can connect your data with this external dataset.
Linked Data vs Linked Open Data The fact that Linked Data is defined

in a personal note of Tim Berners-Lee and is not formally endorsed by W3C contributes to the ambiguity of the definition of the concept of Linked Data. The discussions regarding this topic generally come down to which technology is used to represent data. Some people argue that RDF is integral to Linked Data, others suggest that, while it may be desirable, use of RDF is optional rather than mandatory. Some reserve the capitalized term Linked Data for data that is based on RDF, preferring lower case linked data, or linkable data, for data that uses other technologies. In our work we will use the term Linked Data to refer to data published on the Web in accordance with the Linked Data principles. We will also stick to the most common opinion that RDF is a standard for representing Linked Data. When we want to emphasize that the Linked Data principles were applied to Open Data we will use the term Linked Open Data.

3.3 Linking Open Data project

The LOD project began in 2007 with the support and sponsorship of the W3C Semantic Web Education and Outreach Group (SWEO) . The goal of the project is to bootstrap the Web of Data. Initially, the project was driven mainly by researchers in university research labs and Web enthusiasts, whose aim was to identify Open Data and serve it as Linked Data on the Web. Since 2007 the project has grown considerably due to the involvement from large organizations from different domains. According to the latest statistics of September 2011, the scope of the Linking Open Data project included 295 datasets.

The next Figure demonstrates the range and scale of the Web of Data originating from the LOD project. Each node in the LOD cloud diagram represents a distinct Linked Data set. The arcs indicate the connections between datasets. The heavier the arcs are, the bigger number of links exist. Bidirectional arcs represent links in both directions. For keeping the LOD cloud diagram up to date, the Linking Open Data community effort maintains acatalog of known Linked Data sources, the LOD Cloud Data Catalog

3.4 The LOD project activities

The next step of promoting Linked Data concerns demonstration how Linked Open Data can actually be used. The Linked Data standards represent the data in a structured machine-readable way with explicitly defined semantics. This gives new opportunities to work with data. Compare, for example,

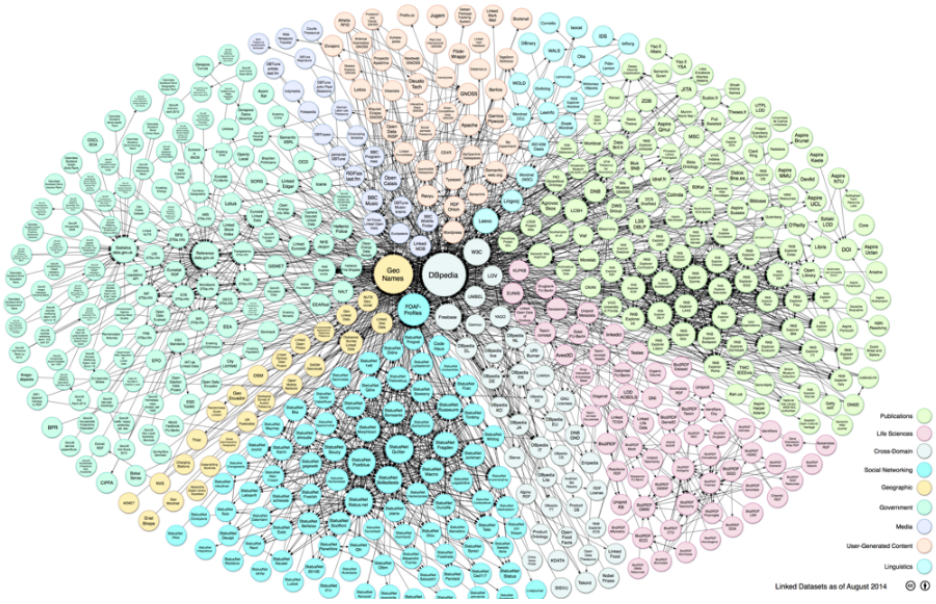

Figure 3.1: LOD cloud diagram as of September 2014

the way the number 2.3 is represented in an xls file and as Linked Data. In xls this number is not of a huge value for machines, they operate with it as with another cell, without differentiating it from the number 3.1 for example. With Linked Data we can provide a meaning to these numbers. For example, The number 2.3 is the inflation rate in Portugal in 2005, when the president of Portugal was Jorge Sampaio, from the Socialist Party. The number 3.1 is the inflation rate in 2006, when the president Anibal Cavaco Silva from the Social Democratic Party was elected. Now machines can use these numbers to analyse inflation rate in relation to the political situation in the country. The Web of Data contains a big collections of such structured machine-readable data interlinked to form a single global informational space. We can use the Web of Data to connect disparate data sources on the Web and develop new kinds of applications that operate upon such data, so called Linked Data driven Web applications. As Linked Data is a relatively novel technology, the existing applications are mostly prototypes and will likely undergo significant evolution as lessons are learnt from their development and deployment. Nevertheless, they already give an idea of what will be possible in the future. The present Linked Data driven applications can be classified into generic applications, such as Linked Data browsers and search engines, and domain specific applications that cover the needs of specific user communities.

3.4.1 Linked Data browsers

In the classical Web of documents, browsers allow users to navigate between HTML pages by following untyped hypertext links, that can be understood by humans but are meaningless for machines, and, thus, can not be used to assist people in finding information and guide them through the Web in a smarter way. The Linked Data browsers are Web applications that provides interactive support for navigating through or exploring Linked Data. Examples of LD browsers include the LinkSailor , Tabulator , Marbles and URI Burner. They can process the semantic links established between different data sources and facilitate users in exploring the Web of Data. For example, when a user is looking for the description of the city of Lisbon in DBpedia, Linked Data browsers can interpret the data available about Lisbon and present it nicely using conventional data presentation methods.Thus, the browsers can understand that Lisbon is a city and represent it on a map, as well as other notable locations in the city such as theatres, castles, shops, etc., can be recognized by the browsers as having geographical characteristics and represented on the map. Tabulator and Marbles can also merge data about the same concept from different data sources. They can discover that there are other data sources on the Web that also describe Lisbon, combine their data with the DBpedia data and present the aggregated view to the user.

3.4.2 Linked Data search engines

The Linked Data search engines also take advantage of the ability of machines to process and extract the meaning of information on the Web of Data. The existing LD search engines provide richer interaction capabilities to a user and ensure more accurate search results, than classic search engines with a simple keyword-based search implemented. For example, DBpedia implements the faceted search paradigm that allows users to filter search results according to specific criteria (facets), e.g., people who were born in a certain country or who have a specific profession. Thus, if a user searches for information about Armstrong, the bicyclist, the results can be narrowed to contain only data relevant to bicyclists, excluding those with data about Armstrong, the astronaut, or Armstrong, the jazzman, or other Armstrongs who were not outstanding bicyclists. While DBpedia implements an enhanced search over the DBpedia dataset together with information from interlinked datasets such as Geonames, Freebase and DBLP bibliography, the Falcons search engine provides the same functionality at a Web scale

3.4.3 Domain specific Linked Data applications

Domain specific Linked Data driven applications are those that reuse content of existing LODsets to fulfil different purposes. Numerous examples of such applications exist. The U.K. and U.S. governments are among the key institutions that have recognised the advantages in converting legacy data stores into Linked Data and making explicit links between these heterogeneous data sources. One direct benefit of Linked Data is richer government transparency: citizens can now participate in collaborative government data access, including mashing up distributed government data from different agencies, discovering interesting patterns, customizing applications, and providing feedback to enhance the quality of published government data. The Tetherless World Constellation (TWC) investigates the role of Linked Data in producing, enhancing and utilising government data published on data.gov. For this, TWC develops visualisations and mashups of different government data, including financial data, spending, energy usage and public healthcare. Their works for consuming Linked government data demonstrate how the value of the data is increased in combination with other datasets. For example, one application utilise the U.S. government data spending on fire fighting to integrate it with data from DBpedia about number of fires and burned area in different years. The application shows correlations between the government spending and the the actual fires. Another example combines the smokerrates statistics with data about population and cigarette taxes5. More works done by TWC can be found in.
The U.K. government gave rise to plenty of mashups and visualisations that show immediate benefits of the LD standards for citizens. Among them are applications that provide information about local services, help managing finance and environmental issues. For example, the Walkonomics6 application rates and maps the pedestrian friendliness of streets and urban areas combining government data with real people reviews. It allows to check a street by a post code and helps to understand how walkable it is. Another example is the BUSit London application that reuses information about London buses and allows to plan a bus journey with several changes by indicating which buses to take, where to catch them and where to change. An interesting interactive visualisation of the U.K. government spending is developed by the Open Knowledge Foundation . Where Does My Money Go?represents spending by area and helps to understand where the money of the UK taxpayers goes. BBC is utilising the benefits of Linked Data as means for storing and sharing news. The BBC Programmes site reuses information from other LODsets (e.g., DBpedia and Freebase) to identify and link semantically related information owned by the BBC to increase usability of their web pages

and other applications that make use of it.
The BBC Music site is enriched with artists information from MusicBrainz and artists biographies fetched from DBpedia to compose introductory texts. Talis Aspire is a Linked Data driven application that helps educators to create and manage lists of learning resources, e.g., books, journal articles, Web pages. Users interact with the application via a conventional Web interface, while the data they create is stored as Linked Data. Aspire then uses the Linked Data principles to connect the learning resources with related data elsewhere on the Web and enrich the range of material available to support the educational process. This resource list management system is currently used by thousands of students at the University of Plymouth and the University of Sussex . Another interesting application that reuses the DBpedia data is a generic reviewing and rating site, Revyu 9. For example, when a film is reviewed on Revyu, the site attempts to provide more information about the film (e.g., the directors name and the film poster) by reusing the data from DBpedia.An interesting mobile application was developed upon the DBpedia dataset, DB-pedia Mobile . This application helps tourists to explore a city by identifying their location based on the current GPS signal of the mobile device and rendering a map with indications of nearby interesting places. One of the examples of Linked Data driven applications in the life science domain is the NCBO Resource Index [18]. It relies on LOD from different biomedical datasets on the Web of Data and supports researchers in exploring them. Zemanta is a content recommendation tool that reuses data from DBpedia and Freebase in order to help users to better organize their blogging activities. It suggests relevant links, articles or images while users write their blogs to make them more interesting and attractive for readers. Google uses Linked Data describing people, products, businesses, organizations, reviews, recipes and events in its search results to provide them in the form of rich snippets . It also uses the extracted Linked Data to directly answer simple factual questions such as the birth date or place of somebody. Google answers such queries not only with a list of relevant links, but it provides the actual answer.

.1 Source Code

```
print('Hello World!')
    use indentation and {special &characters}
```