LAPPEENRANTA UNIVERSITY OF TECHNOLOGY
School of Industrial Engineering and Management
Department of Software Engineering and Information Management

Poorang Vosough

IMPLEMENTING AN OPEN DATA SYSTEM AND SHOWING ITS BENEFITS

Supervisors: Professor, Ph.D. Kari Smolander,

Associate Professor, D.Sc. (Tech.) Uolevi Nikula

Abstract

LAPPEENRANTA UNIVERSITY OF TECHNOLOGY
School of Industrial Engineering and Management
Department of Software Engineering and Information Management

Poorang Vosough

IMPLEMENTING AN OPEN DATA SYSTEM AND SHOWING ITS BENEFITS

Master's Thesis

2013

70 pages, 28 figures, 1 appendix

Supervisors: Professor, Ph.D. Kari Smolander,
Associate Professor, D.Sc. (Tech.) Uolevi Nikula

Keywords: Open data, Linked open data, Open government data, JSON, Android

Open data refers to publishing data on the web in machine-readable formats for public access. Using open data, innovative applications can be developed to facilitate people's lives. In this thesis, based on the open data cases (discussed in the literature review), Open Data Lappeenranta is suggested, which publishes open data related to opening hours of shops and stores in Lappeenranta City. To prove the possibility of creating Open Data Lappeenranta, the implementation of an open data system is presented in this thesis, which publishes specific data related to shops and stores (including their opening hours) on the web in standard format (JSON). The published open data is used to develop web and mobile applications to demonstrate the benefits of open data in practice. Also, the open data system provides manual and automatic interfaces which make it possible for shops and stores to maintain their own data in the system. Finally in this thesis, the completed version of Open Data Lappeenranta is proposed, which publishes open data related to other fields and businesses in Lappeenranta beyond only stores' data.

Acknowledgments

I would like to express my profound gratitude to my supervisors Professor Kari Smolander and Associate professor Uolevi Nikula, who gave me the opportunity to do my Master's Thesis on the topic related to a university project, CrossCom. Your trust, support, guidance and advice helped me during this thesis.

I would like to express my deepest gratitude to my Granny and my Uncle Cyrus who always supported me in my life. They are always with me in my heart and memory, although they are not in this world anymore.

I extend my kind gratitude to my Mom and Dad who encouraged me to continue my studying towards Master's Degree in Finland. They have always motivated me in my life to do my best to achieve success.

Lappeenranta, 26 May 2013

Poorang Vosough

Table of Content

Abbreviations

API	Application Programming Interface
CSV	Comma Separated Value(s)
EU	European Union
GUI	Graphical User Interface
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
ICT	Information and Communication Technology
IT	Information Technology
JSON	Java Script Object Notation
KML	Keyhole Markup Language
LOD	Linked Open Data
MVC	Model View Controller
ODA	Open Data Albania
ODL	Open Data Lappeenranta
OGD	Open Government Data
OGDI	Open Government Data Initiative
PDF	Portable Document Format
PHP	PHP: Hypertext Preprocessor
RDF	Recourse Description Framework
REST	Representational State Transfer

SDK	Software Development Kit
SQL	Structured Query Language
SparQL	SPARQL Protocol and RDF Query Language
TSV	Tab Separated Value
URI	Universal Resource Identifier
W3C	World Wide Web Consortium
XML	EXtensible Markup Language

1 Introduction

The term open data appeared for the first time in 1995, when democratic countries started to publish government data in standard formats on the web to increase transparency and trust in the society. Providing data in standard format which can be processed by machines and software agents facilitated reusing data for developing innovative software applications. Nowadays (after 2007) embracing open data paradigm is not limited only to government data since other organizations, companies and business owners across the globe are increasingly interested in the idea of open data and publish their data in standard formats for public access based on open data principles. [1, 8, 41]

1.1 Background

Russians represent 40% of all travelers to Finland since 50000 Russian lived and 2.6 million visited Finland in 2010. Almost half of the Russian travelers travel to Finland for leisure time activities, one third of them come for shopping and the rest have other interests such as business [47]. However, when crossing the Finnish border, Russian travelers may face several problems such as process of getting visa, language barriers, car parking regulations and shops' opening hours.

Therefore, Improving Social Service (ISS) project, co-funded by the European Union, the Russian Federation and the Republic of Finland, created a social web-portal, CrossBorderTravel.eu, to ease travelling for Finnish and Russian travelers and to iron out mentioned problems. The web portal not only provides information about differences between two countries but also makes it possible for portal visitors to help each other in real-time using social features provided by the portal.

One of the big issues Russian travelers have in Finland is related to differences in stores' opening hours between the two countries. In this case, one of the services provided by

CrossBorderTravel.eu web portal is providing information about shops and stores in Lappeenranta and their opening hours. However, shops and stores may change their opening or closing hours in different seasons or specific days during a year. But there is not any automation implemented in CrossBorderTravel.eu web portal to automatically update data related to opening hours in the case that shops and stores in Lappeenranta changed their opening hours.

However, if data related to stores' opening hours is kept in an open data system adhering to open data principles, which provides standard interfaces that publish data in machine-readable format for public access, CrossBorderTravel.eu web portal and other services will be able to receive stores' opening hours data directly from the open data system. Basically, if business owners provide their data to such an open data system which publishes the data in machine-readable format for public access, developers will be able to use published data for creating innovative services. In this case, not only business owners will benefit but also other people will be provided new services and applications.

1.2 Objectives

In this development project, the benefits of open data are presented in practice. For this purpose, an open data system is created to keep basic information about shops and stores in Lappeenranta. The open data system provides both manual and automatic interfaces for local shops and stores to update their own data in the system. Also, stores' data in the open data system are supposed to be presented in a machine-readable format. In this case, web and mobile applications are developed using published data in standard format to demonstrate the benefits of open data in practice. Moreover, other services like CrossBorderTravel.eu web portal will be able to update their database using machine-readable format of data provided by the open data system. The outcome of this development project will be presented to stores in Lappeenranta to observe the benefits of open data in practice. In this case, they will realize the benefits of such an open data system, which are the following: secure enough, easy to use and effective in improving their business, so they can update their data in the open data system using provided interfaces.

1.3 Research questions

To implement the open data system discussed in Section 1.2 (Objectives), there are three research questions defined, which are answered in different chapters of this thesis.

RQ 1: Why open data is the appropriate solution?

RQ 2: What are the existing solutions for making an open data system?

RQ 3: How to implement an open data system in practice?

Research questions 1 and 2 will be answered in Chapter 2 (Literature review), and research question 3 will be answered in Chapter 4 (Proof of concept).

1.4 Structure

This thesis is composed of seven chapters. Chapter 2 describes the basic concepts of open data based on background literature about open data. Chapter 3 defines the research process of the thesis. Chapter 4 (Proof of concept) presents the constructive part of the development project. Chapter 5 conducts an evaluation about the outcomes of the constructive part. Chapter 6 presents the discussion based on supporting literature and constructive part. Finally, chapter 7 gives the conclusion of the development project of the thesis.

2 Literature review and related research

In this chapter, a background research is conducted to track the state of open data around the world. First, open data definition and principles will be described in this chapter. Then, different levels of data openness will be presented based on Five-star open data model. Furthermore, benefits of open data will be discussed, in addition to open data challenges. Later on, open data projects developed by different solutions will be discussed. Then, the concept of five-star open data will be presented in the projects developed based on linked open data principles. Finally in this chapter, Open Data Lappeenranta is suggested in summary section.

2.1 What is open data?

The idea of open data can be seen as closely related to the notation of open source software that makes the user of a software program able to freely access the source code of the program, study it, change it and redistribute it [18, 19]. Similarly, open data refers to publishing any sets of data in a machine-readable formats which everybody is free to use, reuse and redistribute with no licensing or patent restrictions [1]. Opening up the data, enables developers to create new services and applications to facilitate peoples' lives [15].

Open data is often considered as data which is related to government and it is supposed to be open to increase transparency in the society. However, other organizations, companies and citizens can publish open data too. Open Government Data (OGD) means publication of government data in raw open format (Open data) [2]. Many democratic countries such as the United States and the United Kingdom have supported open data practice to facilitate free access to their government data to lay foundations for transparency and decision making [8, 20].

In the next section, most important features of open data, known as open data principles, will be discussed in more detail.

2.2 Open data principles

Open data principles can be classified into four major categories: legal issues, accessibility, value and technical perspective. Each category includes two or more principles which are prerequisites for publishing open data.

From the legal point of view, open data should be license free, non-discriminatory and free of charge for anybody who is using the data.

- **License-free:** Data is not subject to any trademark, copyright or patent regulation. However, reasonable security and privacy may be allowed [2].
- **Non-Discriminatory:** Data is available for anybody without any need for registration for accessing data [2].
- **No charge:** Using, reusing or redistributing the data is free of charge for anybody [29].

From the accessibility point of view, open data should be findable and accessible.

- **Findable:** Data should be published on the web in a way that users can easily find the location of the data [29].
- **Accessible:** Data is supposed to be published for the widest range of users and for the widest range of purposes. To access the data, users do not need to accept any agreement on purposes of data usage [2].

From the value point of view, open data should be primary (original), complete and timely.

- **Primary:** Data is supposed to be published as what exists in the primary source. In fact, privacy or security concerns do not restrict users from having access to the data in its original form [29].
- **Complete:** Data is not supposed to be published in a partial format. In other words, the whole data existing in the original source should be published [29].
- **Timely:** Data is supposed to be made available as quickly as possible so the value of the data will be preserved. Moreover, data which is out of date and does not have real use should not be published [2, 29].

From the technical point of view, open data should be machine-readable, documented and follow open standards.

- **Machine-readable:** Data is supposed to be published in a structured format that allows automated processing of the data [2].
- **Open standard:** User should be able to receive data using open standards without any need to buy vendor's product or obtaining their permission. It is suggested that data providers use the same open standards as users [29].
- **Documented:** Published data should have appropriate documentation to provide effective use of data for users. Lacking an appropriate documentation, data will lose its understandability and will remain useless [29].

2.3 Different levels of openness

Primary data is defined as data in the original format, which maintains the semantic context in which they were defined in their original sources. On the other hand, data can be classified into three categories: unstructured data, semi-structured data and structured data [2].

- When there is no scheme defined for the data, it is called **unstructured data**. This type of data contains only the content and a means of presenting it. Text available in a PDF file or an HTML page is an example of unstructured data.
- In **semi-structured data**, some of its general attributes can be known in advance, however, one cannot always predict all aspects of a given piece of data. Harvard-style referencing for journal articles is an example of semi-structure data, which contains fairy similar items related to articles (such as author name, publication year, title of the article and journal name) in a specific order style.
- **Structured data** are much easier to understand when relevant values of data can be identified clearly according to corresponding concepts. For example, tables of relational databases have self explanatory data in columns with each row containing a different value.

World Wide Web Consortium (W3C) has developed Five-star open data model consists of five different levels of data openness based on the characteristics and usefulness of data in each level. This model is used globally for assessing data readiness for reuse. Primary data regardless what structure they have, can move toward openness by passing the prerequisites defined for each level of Five-star open data model. The first three levels of this model define the primitive standards for publishing open data. On the other hand, the last two levels present creating linkable datasets which can be easily joined together and create new datasets. [25]

In the first level, data should be available on the web in a way that anybody can access the data and reuse it. However, it requires considerable effort to reuse data in this level. In fact, lack of standard data format in this level makes it difficult to identify where the data starts and ends. PDF documents and data in HTML tables are examples of one-star open data. [25]

In the second level, data goes one step further toward being in structured and predictable format. In fact, in this level data is converted to a basic machine-readable format which can be queried and consumed using software program. However, a two-star data is not machine-readable enough to be accessed by any software. For example, data in Microsoft Excel files are two-star data which are only accessible using Microsoft Excel software, or few other compatible applications. [25]

In level three of this model, data is converted to a non-proprietary format that can be accessed by any software. In this case, data does not require any specific software or systems to access it. For instance, data in Comma Separated Values (CSV) files can be opened in any spreadsheet software, whether it is Microsoft Excel, Open Office and so on. In CSV files, each row contains one record of information and each record can contain multiple pieces of data, each separated by a comma [25]. Figure 1 presents a piece of a CSV file presenting simplicity and predictability of this format of data.

```
company,model,year,color
Toyota,Yaris,2010,white
Toyota,Camry,2011,blue
BMW,X6,2012,black
```

Figure 1. Data in CSV format

In the fourth level, not only data include previous features of openness such as machine-readability and availability in the web but also it should be described in a standard fashion like RDF triples [25]. RDF or Resource Description Framework is a W3C standard for the definition and use of metadata description on the web. This standard expresses data on the web as RDF triples in the following form: subject, predicate and object. The subject describes what the data is about, the property shows an attribute of the subject and the object presents the actual value of subject [23, 44]. There are open source tools like CSV2RDF4LOD for converting data in CSV format to RDF. Figure 2 presents the last record of CSV data in Figure 1, converted to RDF in code and graph.

```
<rdf: description
rdf: about=BMW>
  <model>X6</model>
  <year>2012</year>
  <color>Black</color>
</rdf: description>
```

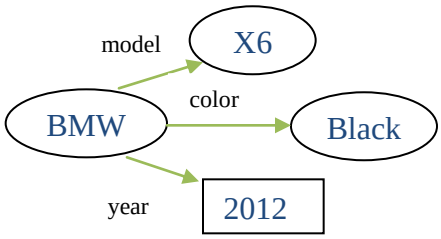

Figure 2. Simple RDF code and graph

Finally in the fifth level, open datasets defined with RDF triples will be linked together to produce new datasets on the web. For this purpose, each RDF triple is linked to a common definition on the web using Uniform Resource Identifiers (URIs) [25]. A URI is a string of characters for identifying a name or a resource on the Internet [2]. In other words, RDF triples consist of subject, predicate and object will be defined by only one unique identifier, as a form of hyperlink, regardless the dataset the RDF triple is used in. Figure 3 shows how two RDF graphs are linked together with a mutual triple “Black”, which is defined by a unique URI on the web.

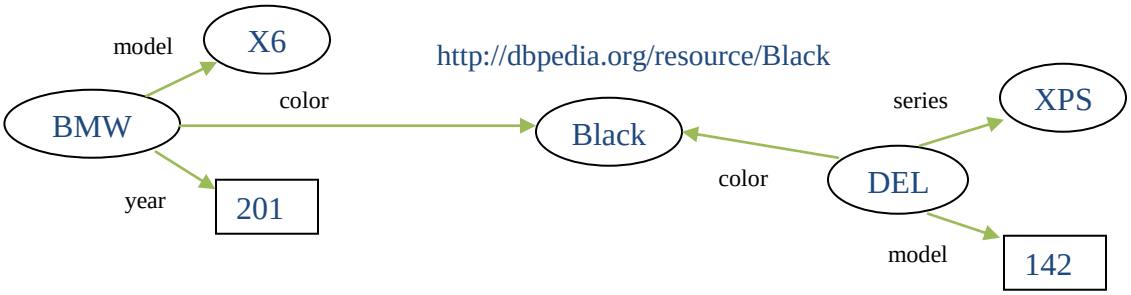

Figure 3. Linked RDF graphs

Basically, five-star open data is the base for creating Linked Open Data (LOD). Linked open data refers to publishing machine-readable data on the web and connecting related data across multiple data sources [2, 24]. However, publishing linked open data needs adhering to the linked data principles [14, 26]:

- In a dataset, all items should be identified by using URIs.
- HTTP URIs should be used, so that people can look up an item using its URI.
- When user is looking up a URI, it leads him or her to more data (Using RDF standard).
- Links to URI in other datasets should be made in a way that enables the discovery of data.

The linking open data community project[1] has published a Linked Open Data cloud diagram based on LOD principles. By September 2011 the project had grown to 31 billion RDF triples, interlinked by around 504 million RDF links [27]. The diagram of the linking open data community project is illustrated in the Figure 4.

[1] http://linkeddata.org

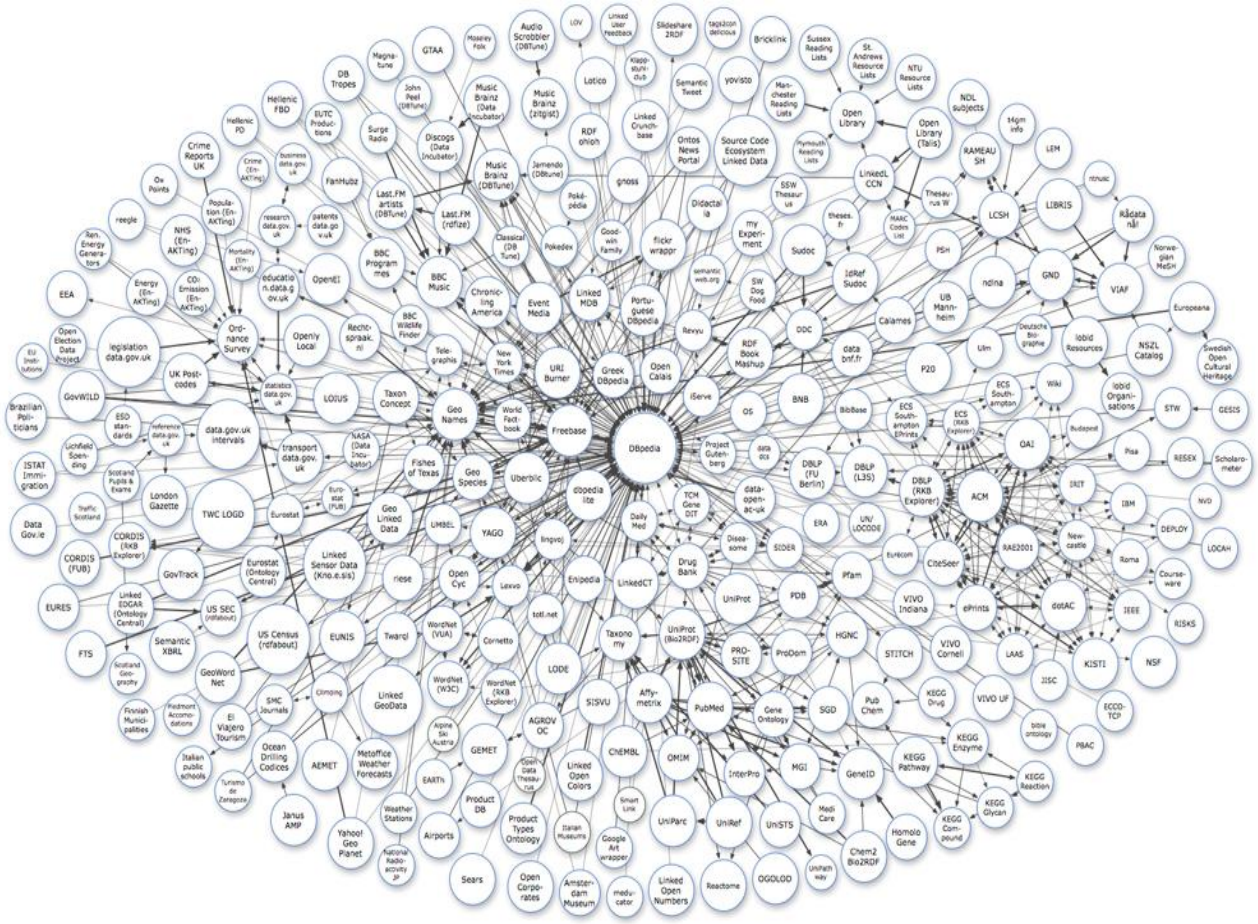

Figure 4. The Linking Open Data cloud diagram

2.4 The benefits of open data

Open data is used by wide variety of people, companies and organizations ranging from citizens to software developers to researchers in both academic and private fields. Moreover, open data can be used directly (e.g. in carrying out research), or indirectly (e.g. for developing mobile application) [45]. Generally, people who have the most benefit from open data can be classified into three major categories: citizens, developers and governments. In this section, the impact of open data will be discussed on each category.

Citizens are the first group who benefit from implementing open data systems in cities. Having an easy access to published open data, they will find economic development opportunities which can lead them to job creation. Also, opportunities for community engagement will be increased in the society and people will have collaboration in meeting social needs. However, the most important benefit that using open data can bring for members of the city is software applications and services which can be developed based on open data. [5]

Developers are the second group who benefit from published open data. They will have easy and free of charge access to the provided data by governmental or non-governmental companies and organizations for making new applications. It means open data helps them to have economic software development opportunities to develop initiative applications. [5]

Governments are the third group who benefit from publishing their data in open formats. In fact, "*offering government data in a more useful format to enable citizens, the private sector and non-government organizations to leverage it in innovative and value-added ways*" [28] will increase citizens' engagement and collaboration with business and community groups and improve government transparency and trust in the society. It means, using the democratization of information in the city will increase government transparency and will increase citizens' trust in city government. [5, 45]

Figure 5 illustrates open data impacts on different aspects of the city. City governments provide their information for public use to increase transparency and community level in the city and to facilitate citizens' lives by creating useful services based on their needs. Therefore, development companies obtain free and easy access to required data for developing innovative and powerful applications. In this case, citizens will enjoy new applications and services created by developers and transparency provided by city government. [5]

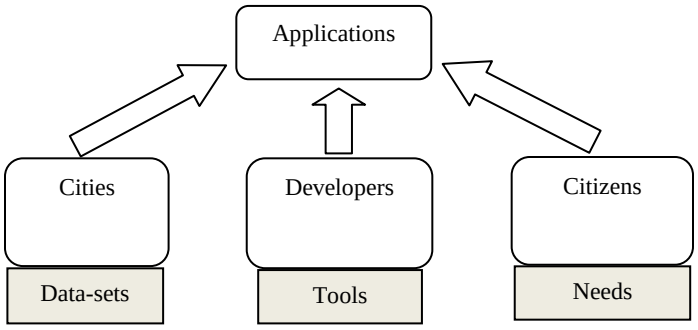

Figure 5. Impact of open data on different aspects of the city [5]

2.5 Open data challenges

Similar to open data benefits, the challenges of open data are also abound, spanning from cultural and organizational to legal, skills and technological aspects [21]. From the technical point of view, there are challenging questions which should be answered before implementing the open data system. The main questions can be summarized as follows:

- What platform is needed for publishing data? An infrastructure or a cloud platform?
- What are the best framework and technologies for developing the open data system considering selected platform?
- What solution will be selected for publishing data out of existing open data techniques and methods?
- What is the appropriate interface that users interact with data? What are convenient formats that users receive the data?
- What is the level of data openness? Is it enough to publish data in a standard format according to open data principles? Or is it necessary to provide linkable datasets adhering to linked open data principles?
- What is the size of the data that the open data system is going to publish?

Therefore, data publishers should have a deep understanding about open data objectives, size of data which is going to be published, available resources considering potential limitations such as time and budget before starting to implement an open data system.

Moreover, there are other challenges related to community awareness about open data. As it was discussed in Section 2.1 (What is open data?), open data provides data in standard formats that can be processed by machine. However, being machine readable does not equate to being easy-to-read for human. Therefore, majority of people may prefer a closed version of data since only limited group of people are technical specialists who can interpret and use open data. For example, they may prefer information in PDF files rather than structured data in XML or JSON formats, or business owner may prefer to put their business information in simple HTML pages rather than publishing them as open data adhering to structured linked open data principles. Thus, it can be challenging to convince people about open data benefits, when they are not even familiar with primitive open data principles. [48]

2.6 Open data cases developed with different solutions

2.6.1 Using W3C standards

Adoption of World Wide Web Consortium (W3C) standards like URI, RDF and SparQL, is one way to implement open data. URI and RDF standards were already discussed in Section 2.3 (Different levels of openness). SparQL standard will be discussed in this section, in addition to the way these standards work together for creating open data.

SparQL is a W3C recommendation for accessing and querying RDF data [13]. In fact, SparQL can be considered as the SQL on the web, which provides a possibility for querying RDF triples and graphs [2].

To publish open data using W3C standards, data should be published in dataset by a single provider, accessible by URI (Uniform Resource Identifier), available in Resource Description Framework (RDF triples) and queried by a query language such as SparQL [2].

Open Data Albania

Open Data Albania (ODA) is an initiative that embraces open data principles to increase openness of data offered by different government institutes in Albania. In fact, this project aims to represent these data in more understandable ways for solving the problems that the country is currently facing, such as cost analysis of national road construction, statistics about educational university problems and information about how people spend their loans in Albania. [8]

For this purpose, they collect data from different sources, which include websites of government institutions in Albania such as Ministry of Justice, Ministry of Economy, Ministry of Finance and Bank of Albania. [8]

However, a unified representation for data is needed since data is gathered from various sources. For this purpose, ODA ontology [1] was modeled based on linked open data principles to provide a common understanding of the domain knowledge by providing unified and formal semantic to gathered data.

On the other hand, collected data from government sources are mostly in unstructured or semi structured formats such as text files (like pdf and Excel). Thus, after collecting the raw data from their original sources, the next level is converting them to CSV format. In this case, the first version of datasets is created in a unified format. Later on, raw datasets are converted to RDF triples based on ODA ontology. [8]

For converting datasets to RDF, an automatic process was performed using XLWrap wrapper tool, which is a java implementation mapping between dataset's Excel files and ODA ontology. On the other hand, identifying datasets with well-formed URI links RDF in this project to other established data sources. This process makes it possible for the users to navigate the web of data. [8]

In order to assist the users to query RDF datasets, Open Data Albania has provided a web page named knowledge explorer, which allows the visitors to search their desired data by criteria like topics, indicators and date. Based on what visitors search for, the query is compiled and sent to a SparQL endpoint to process it and produce result based on the chosen criteria. [8]

When SparQL endpoint processed the queries, the result is displayed to the user in graphical representations using Google visualization API, which is an open source visual application interface [8]. As Figure 6 illustrated, both user community and source institutions can use the linked open data produced by Open Data Albania.

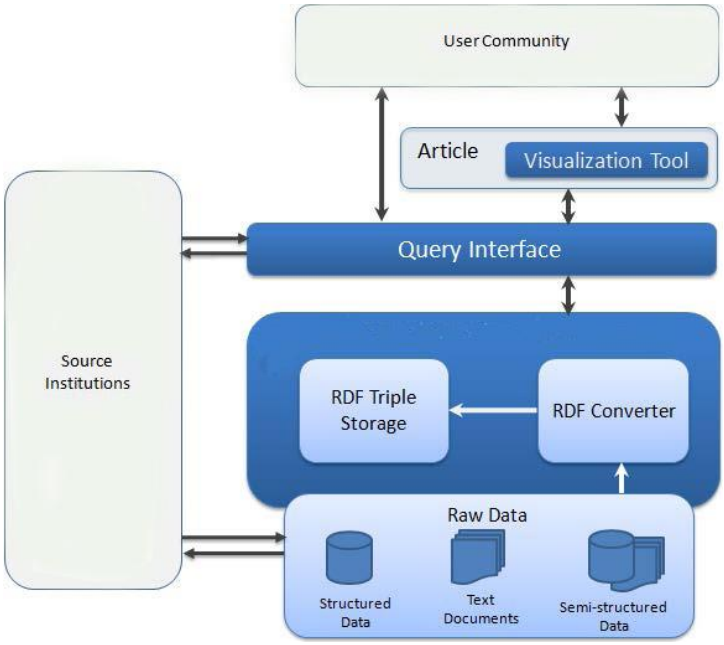

Figure 6. ODA architecture for publishing open government data [8]

US government Linked Open Data: Data.gov

United States is one the first courtiers which supported publishing government data for public access. During the recent years in US, Open Government Data (OGD) has become a vital communication channel between government and citizens. The largest open data web portal in US is Data.gov, which is deployed to release OGD datasets online. Data.gov project was launched on May 2009 with using only 47 datasets, however, today it offers access to more than 400,000 different datasets from 185 US government agencies and organization. During this time, the goal of Data.gov was always improving user's ability to discover and retrieve US

government datasets collected from various sources, using open data and linked open data principles. The outcome of the project helps people to make better decisions, and create smart phone apps, visualizations and data mash-ups. [11]

From the beginning of Data.gov project, open data principles, linked data and semantic web have been considered as the technological area for the project's development. Today, more than one thousand government datasets have been made available for public access in Resource Description Format (RDF) format, with totally six billion RDF triples based on W3C standards. Also, all the collected government datasets are at the fourth level of Five-star open data model discussed in Section 2.3 (Different levels of openness). It means all datasets are provided with URL; moreover, most of the datasets are linked to other data in other datasets, which means five-star data. [11]

In 2011, Data.gov project focused on providing open data related to health[2], energy[3], education, law, public safety, research and development. In the same year, the first two open data projects (health and energy) were launched by Data.gov and published on the web. [11]

Data provided by Health.data.gov project is available in popular machine-readable formats to facilitate application-specific processing. For example, users can get hospital data in Java Script Object Notation (JSON), Comma Specific Values (CSV), Atom Feeds (.atom) and many other available data formats. In this case, users interested in hospital data can either browse the linked data via the web portal or query the linked open data for developing their own applications using Health.data.gov as the platform. Using Health.data.gov, developers have created applications to improve citizens' health. These applications range from PatientLikeMe[4], which lets patients connect to other people who have similar symptoms and find treatment options, to Asthmapolis[5], which uses a GPS-enabled inhaler to make it possible for users to track where and when their asthma attacks are occurring. [11]

[2] www.health.data.gov
[3] www.energy.data.gov
[4] www.patientlikeme.com
[5] www.asthmapolis.com

The focus of Energy.data.gov project is on providing innovation and access to datasets around the energy field, from government to utilities and homeowners. One important requirement for providing open energy datasets in Energy.data.gov is to integrate search across multiple energy data sources beyond Data.gov project. Data.gov project provides its energy data in open datasets with SparQL endpoint. Therefore, it is possible to integrate Data.gov with other energy datasets with similar standard endpoints. For example, Open Energy Information[6] (OpenEI), launched by National Renewable Energy, also provides energy data sources in SparQL endpoint. In this case, because Data.gov and OpenEI both provides energy data through the same endpoint (SparQL), it was possible to create an efficient browser across both data sources to access wide range of open data around the energy field. Basically, when standards-based approach is used in different projects, future integrations and extended capabilities will be possible. [11]

Open Government Data in Brazil

In September 2011, Brazil became a member of Open Government Partnership to promote worldwide adoption of Open Government Data (OGD). In this case, Brazil was committed to public transparency and to action in securing open publication of official data. To meet this commitment, Brazil launched Brazilian Open Government Data (Brazilian OGD[7]) portal on the web. First efforts toward publishing Brazilian OGD can be traced back to when Committee of the Presidency of Brazil (COI) began to gather large amounts of government data for digital publication in 2009. The goal of COI team was to create a central information catalog of public data to monitor government activity and improve governance. Later on, because the project was so successful, reflecting open data principles, the data catalog was made available for public (re)use in 2010. [12]

Later on DadosGov database was created based on spreadsheets provided to COI team by 40 different Brazilian government agencies, totally approximately 2.5 million records in relational

[6] openei.org
[7] dados.gov.br

database. Moreover, DadosGov data was published in XML and JSON formats to facilitate reuse of the data for application developers. [12]

The office of World Wide Web Consortium (W3C) in Brazil played an important role in OGD publication process in this country by sponsoring training in OGD technologies for information technology professionals in public sector. In 2010, subset of the DadosGov database was converted to RDF triples using W3C standards. Describing Brazilian OGD with well-known RDF facilitated integration of Brazilian open data with other datasets in LOD cloud such as DBpedia. As the result, in 2011, Brazilian OGD was linked to DBpedia, containing information in details about Brazilian cities, states, population, areas and so on. [12]

2.6.2 OGDI

Open Government Data Initiative (OGDI) led by Microsoft, is an initiative solution for publishing government and other public data in a more quick and efficient way [40]. OGDI has been written using C# and .Net framework based on a cloud computing platform, Microsoft Windows Azure, "*an open and flexible cloud platform that enables users to quickly build, deploy and manage applications across a global network of Microsoft-managed datacenters.*" [39]. In general, OGDI consists of three major components: data loader, data browser and data service.

Data loader is a software utility which provides a user interface for data publishers to import data quickly and easily into the relevant catalogues in the cloud. Data can be received by data loader user interface in two different formats, CSV or KML files [34].

Data service is a RESTful Web service which is implemented using HTTP and the principles of REST (Representational State Transfer) to expose published datasets in the cloud for programmatic access [34].

Data browser or the Interactive SDK is an ASP.NET web application which provides an interface to the OGDI data service. In fact, data browser allows users to browse, query and interact with published data in the cloud. Also, the data browser will visualize data for the user in

recognizable formats such as tables, maps, bar graphs or pie charts. In this case, instead of downloading data, end-users will be able to interact with user-friendly visual tools which illustrate complex data in a more meaningful manner. In addition to browsing and querying data, developers can use published data exposed by OGDI in machine-readable formats such as XML, JSON, CSV and so on to develop their own applications using variety of languages and frameworks. [34, 38]

Niagara Region Open Data

Niagara Region Open Data[8] is an open data project developed using the integration of HTML/Java script, SharePoint and OGDI DataLab[9]. In the web page, data about Niagara region located in southern Ontario, Canada can be found in five major data catalogs: Academic & Cultural, Health, Land Planning, Recreation and Transportation for the user. Also, user can look for a specific data subject using the search textbox located on the page. Regardless of the search method used, in both ways open data result will be provided to the user. As Figure 7 presents, open data about "flu clinic in Niagara region" can be found in data table and map.

[8] http://www.niagararegion.ca/government/opendata/data-catalogue.aspx
[9] http://ogdisdk.cloudapp.net/

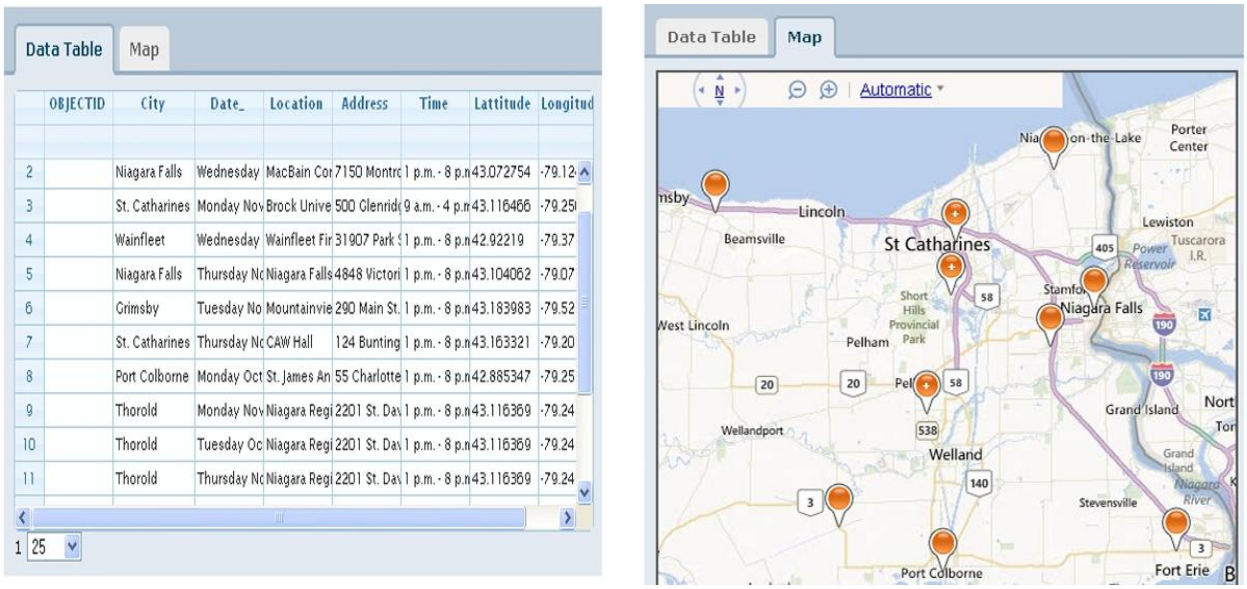

Figure 7. Niagara Region flu clinics open data

Moreover, it is possible for the user to download the open data in several standard formats such as HTML, XML, CSV and KML which can be used for different purposed based on the user's needs [15].

2.6.3 CKAN

Comprehensive Knowledge Archive Network (CKAN) is a powerful data management system which makes data accessible, discoverable and presentable on the web by providing tools for publishing, sharing, finding and using data. CKAN makes it possible for data publishers (national and regional governments, companies and organizations) to make their data open and available for public access. CKAN uses its internal model to store metadata about different datasets and presents the data on a web interface which allows users to search and browse published data in different categories. CKAN also offers a powerful Application Programming Interface (API) which allows third-party applications and services to be built using the published

data. CKAN is currently used by governments and user groups worldwide to provide both official and community data portals for publishing open data. [37]

Open Cities Project

Open Cities project[10], co-funded by the European Union, aims to use real-time open data to facilitate citizens' lives in seven major European cities: Helsinki, Berlin, Amsterdam, Paris, Rome, Barcelona and Bologna [35]. Real-time data is a type of open data, which its main characteristic is that the data need to be updated often (every minute, second or even fractions of seconds). In fact, real-time data requires a special tool in the metadata, which is called the update rate. Depends on type of the data, update rate varies in different real-time data cases. For example, economic and demographic data are updated once per year; however, traffic-related data are supposed to be updated in every second or even less. On the other hand, sensors are the most common data source when we talk about real-time data. In fact, sensors are responsible for updating real-time data according to the update rate defined for the data. [5]

To create a real-time open data platform, Open Cities project asked city councils for information from city datasets in standard formats. Cities responded that they have thousands of standard datasets in different categories, in addition to many sensor networks responsible for collecting the data (e.g. urban transport). In this case, the main objective of Open Cities project was to offer a platform to store structured data provided by different cities' sensor networks into high speed databases and to provide a web interface which gives the users the possibility of downloading gathered data in standard formats [5]. Figure 8 presents the real-time open data platform deployed under the Open Cities project.

[10]Http://www.opencities.net

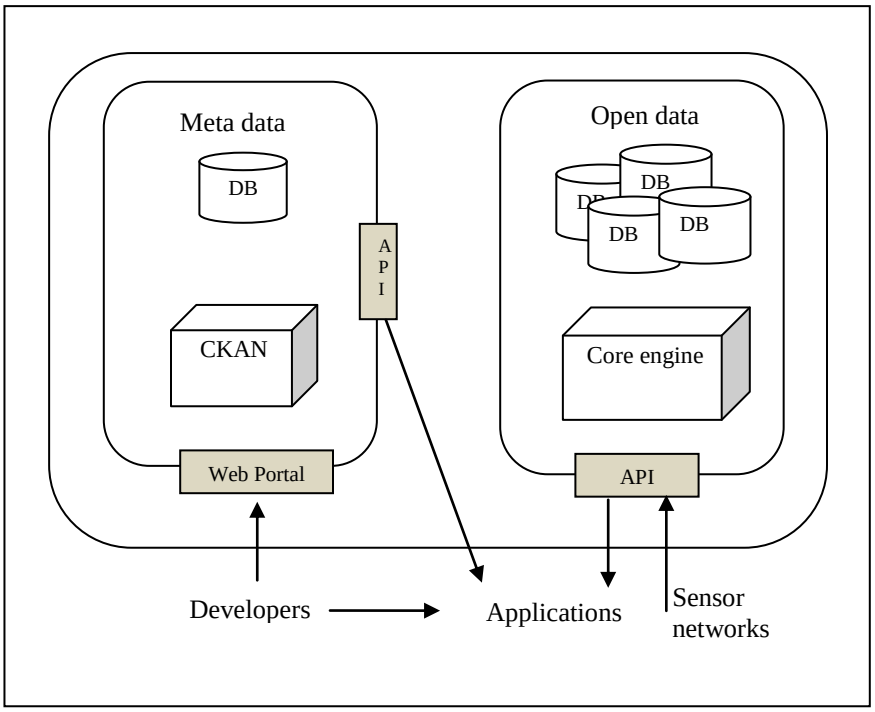

Figure 8. Real-time data platform provided by Open Cities project [5]

As it is illustrated in figure 8, the architecture provides interfaces for adding and consuming data to and from the platform. To add data into the platform, CKAN portal offers a database which stores metadata information containing links to open data stored in core engine databases. Also, the CKAN portal allows users to register into the system via a web portal and store their datasets using a REST API. Similarly, core engine provides a REST API which allows sensor networks to easily store their raw data into the platform. For consuming the data from the platform, the CKAN portal and core engine offer web interfaces which make it possible for users to search for the data in different categories (e.g. education, health and disability, emergency services and urban transport) and download the open data in standard formats such as XML and JSON. [5]

Real-time open data platform provided by Open Cities project gives the citizens the right to access data held by the government. In this case, city problems are tackled with the help and involvement of citizens [36]. In fact, Open cities project aims to publish real-time open data in

smart cities where citizens have become information workers with open access to the data of the city [17]. Real-time open data published by Open Cities project can be used in innovative applications development. Barcelona bike leasing system (Bicing[11]) and Helsinki tram application (Mobitransit Helsinki[12]) are mobile applications developed using standard data provided by Open Cities project. Bicing mobile application makes users aware of whether they can have a bike in the closet bike station out of 400 stations in Barcelona City and Mobitransit Helsinki application allows users to visualize public transport in Helsinki City in real time instantly from their mobile phones [5]. Figure 9 presents screenshots of Bicing and Mobitransit Helsinki mobile applications running on an Iphone device.

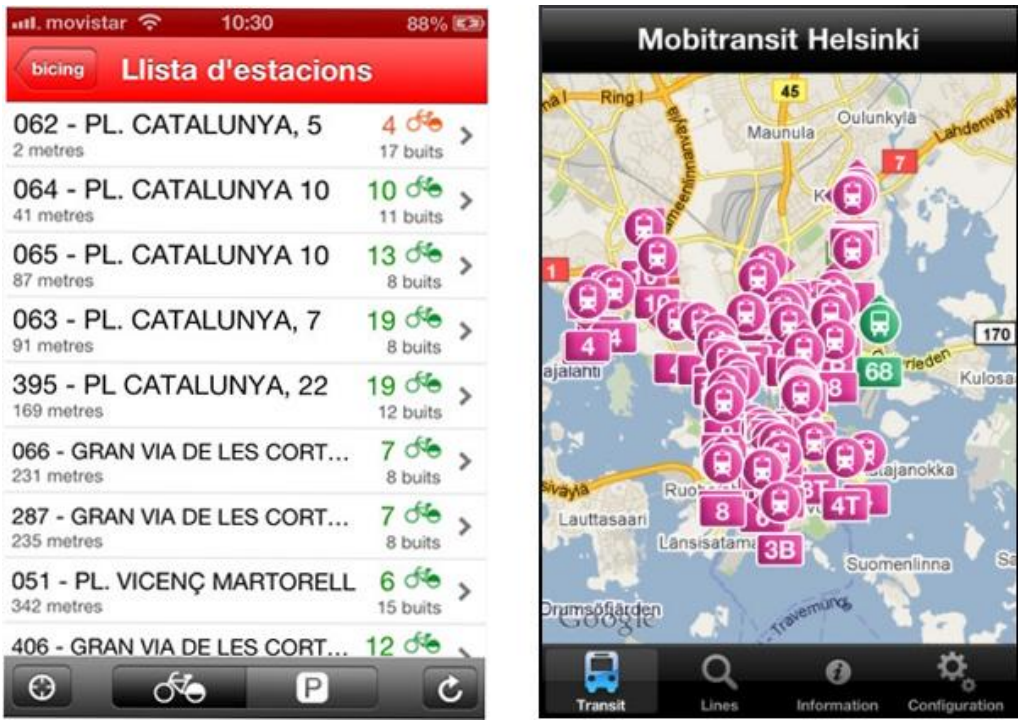

Figure 9. Bicing and Mobitransit Helsinki mobile applications

[11] https://www.bicing.cat/ca/content/app-bicing
[12] http://www.mobitransit.com/

2.7 Five-star open data on the web

2.7.1 Standard datasets for publishing data

As it was discussed in Section 2.3 (Different levels of openness), to produce five star open data, published open data from different sources should be linked together. For this purpose, data should be published in standard datasets regardless what the data describes, where the data is collected from, with what method the data is published and with what format. In other words, organizing collected data from data sources in unified representation (standard datasets) facilitates the process of publishing the data and makes it possible interlinking open datasets together [1]. In this section, two open data projects (GovWild and Transport Open Data in France) will be discussed to demonstrate the importance of defining well-structured datasets for publishing open data.

Government Web Data Integration for Linked Data (GovWild) is a project that integrates different government open data into clean, well-structured and concise datasets. GovWild integrates data sources from US and EU government agencies. The project selected US and EU as the main sources for collecting data, because there is major effort in these regions for publishing open data. Moreover, collecting data from two geographically distinct origins leads the project to define and implement a universal approach for integrating open data. [9]

GovWild collects open government data from multitude data sources such as US Spending, US Congress, EU Finance and EU Parliament Data in different data formats like XML, CSV, HTML and TSV. Most of the data is collected as unstructured format like HTML, or appears as raw text on the web, e.g. biographic information about famous political characteristics in US. To perform extraction of structured data, the project has created a generic JSON format with specific structure. After collecting the data from different sources in various formats, it is transferred into the JSON format to create a clean, consistent and duplicate-free dataset from the heterogeneous input [9]. Figure 10 presents the HTML data from ec.europe.eu, which is transformed to the generic JSON format with specific standard structure.

```
{ "_id" : "euFinance#28994",
"year" : 2008,
"nameOfBeneficiary" : "ROBERT BOSCH GMBH*",
"coordinator" : false,
"countryTerritory" : "Germany 70049 STUTTGART",
"coFinancingRate" : "67,51 %",
"amount" : 3199959.00,
"commitmentPositionKey" : "F13.A22622.1",
"subjectOfGrantOrContract" : "MULTISPECTAL TERAHERTZ, INFRARED, VISIBLE IMAGING",
"responsibleDepartment" : "Information Society and Media",
"budgetLineNameAndNumber" : "Support for research cooperation in the area of information and
communication technologies(ICTs-Cooperation)(09.04.01.01)."
}
```


Figure 10. Raw Data in JSON format with specific

So, in GovWild project, first in Mapping and Scrubbing phase, collected data from data sources are organized into JSON format (Similar to Figure 10). Then in Data Transformation Phase, the standard JSON objects are transformed as a whole. Finally in Deduplication phase, intra-source JSON duplicates which represent a single real-world object are identified and removed. The integration result is a set of specific JSON objects which represent distinct real objects in the world. As the result, GovWild tool[13] is a search engine-like web application that allows browsing and querying the collected open data interactively. [9]

Other project which demonstrates the role of standard datasets in open data, is building a framework for publishing and interlinking transport open data in France. Publishing transport open data in France makes it possible to develop applications which deliver functionalities related to public transportation in different cities of France. The project applies the workflow to two standard datasets: Passim, a directory which contains information on transport services in French cities, and Neptune, a standard for describing public transport routes in France. Each dataset keeps relevant transport data in specific fields. In fact, providing standard datasets with specific fields is the prerequisite for publishing and interlinking open data in this project. [10]

[13] http://govwild.hpi-web.de

Passim directory is a dataset which identifies and provides a list of information on French passenger transport services [10]. This dataset is published in CSV format which is presented in Figure 11.

Sheet number;Service Name;Coverage service;Region;Department;City;Modes of transport;Type of service;Network ccessibility for disabled person;Land informations;Website; Website accessibility for disabled person;Information points ;Re-mark; Comments; Sms;Mobile application;List of cities covered (Postal code);Sheet created;Sheet modi_ed

Figure 11. Passim dataset with specific fields in CSV

As it is presented, Passim organizes transport data in specific columns (fields) by the character ';'. The names of the columns are self explanatory. Figure 12 shows a CSV line in Passim dataset. A succession of ';' means that the column between ';' is empty [10].

1;05voyageurs;d_epartementale;Provence-Alpes-C^ote d'Azur;Hautes-Alpes;N/A;Autocar, Covoiturage ;Calcul d'itin_eraire,Description du r_eseau ,Horaires; Non;; http://www.05voyageurs.com ;Non;;;;;; 09/06/2010;04/08/2011modi_ed

Figure 12. A CSV line in Passim dataset [10]

The next dataset is Neptune in XML format, which describes a transport line in French cities. There is an XML file defined for each transport line, which describes all information about that line such as stops, schedules, latitude, longitude and etc. Figure 13 illustrates an example of this format which models a bus stop [10]:

```
<ChouettePTNetwork>
<ChouetteLineDescription>
<StopPoint>
<objectId>NINOXE: StopPoint :15577811
</objectId>
<objectVersion>0</objectVersion>
<creationTime >2007-12-16T14:2 6:1 9.000+01:00
</creationTime>
<longitude >5.7949447631835940</ longitude>
<latitude>46.5263907175936000</ latitude >
<longLatType>WGS84</longLatType>
<containedIn>NINOXE: StopArea :1557779
</containedIn>
<name>CimetieredesSauvages (A)</name>
</StopPoint>
</ChouetteLineDescription>
</ChouettePTNetwork>
```

Figure 13. Neptune dataset in XML format, modeling a bus stop

Basically, defining the transport data in well-structured datasets with specific fields facilitates publishing and interlinking open transport data in this project. As the result of this project, transport application can be developed using multiple datasets simultaneously. For example, application which displays restaurants and other activities around each transit stop, or to find tourist transportation routes based on a destination and interests of the user. [10]

2.7.2 Linking open datasets

In this section, it will be discussed how an open dataset such as Passim (discussed in previous section) can be linked to other open datasets available on the web. As it was mentioned in Section 2.3 (Different levels of openness), providing open data in RDF triples (subject, predicate and object) makes it possible to connect open datasets on the web to create linked open data.

Before converting open datasets to RDF triples, a standard ontology should be defined for the provided data available in the dataset, based on dataset fields. Generally, ontology is a branch of philosophy which clarifies the order and structure of reality [31]. However in semantic web, ontology is defined as a formal, explicit specification of a shared conceptualization. A standard ontology for specific dataset describes the semantics of items in the dataset and gives an explicit

meaning to the provided information [1]. Concerning Passim dataset, an ontology has been defined (Passim ontology) which contains 4 classes and 18 properties based on standard fields defined in Passim dataset. Figure 14 illustrated Passim ontology[14] (available on the web) containing classes and properties.

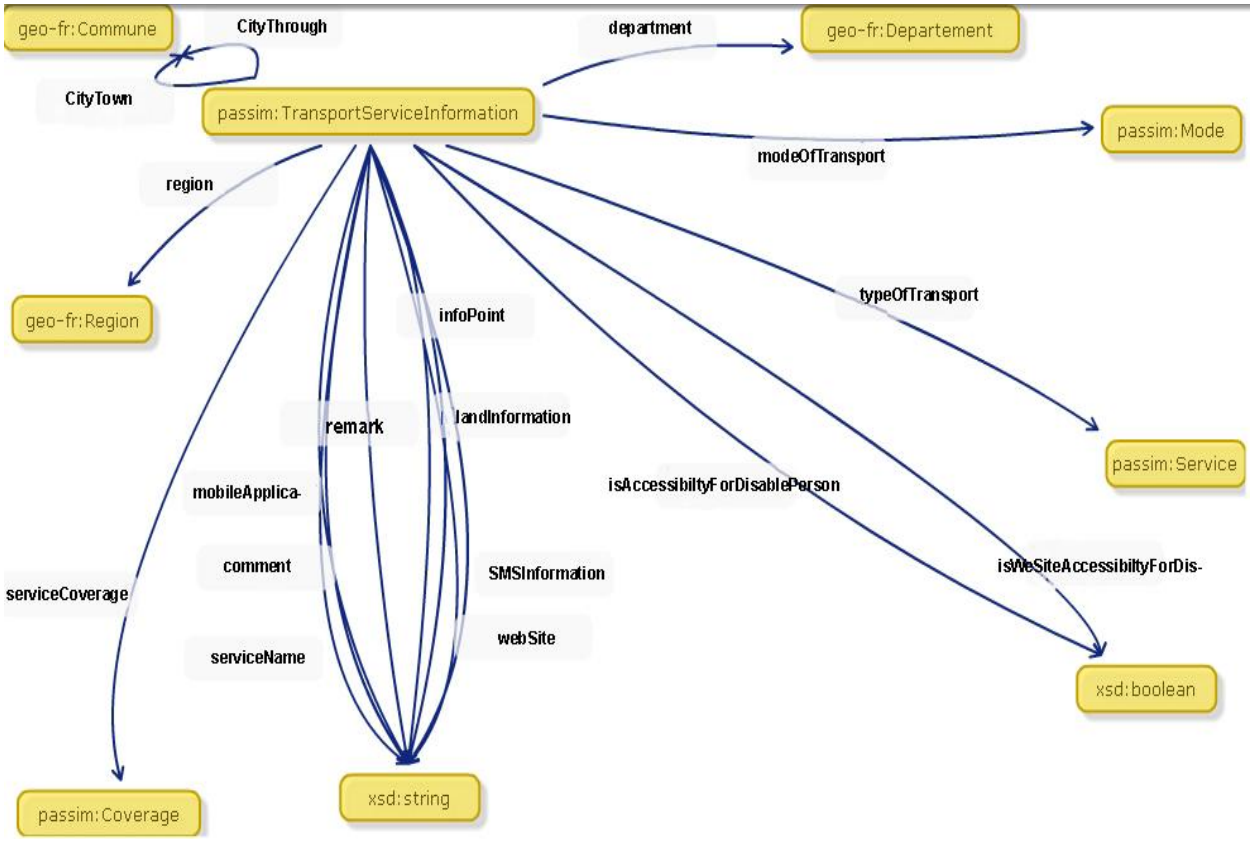

Figure 14. Diagram of the Passim ontology

In the first level of converting, without considering any ontology open data available in Passim dataset is converted to RDF using an open source CSV into RDF convertor tool provided by

[14] http://data.lirmm.fr/ontologies/passim

DataLift. In the next level of converting, RDF file should be converted into another RDF file in a way that data meets the Passim ontology presented in Figure 14.

After converting the open data to RDF triples, it can either be simply published in RDF files on the web, or publish data in SparQL endpoint. A SparQL endpoint allows agents (machines or human) to query published RDF data via the SparQL query language. In Passim dataset case, after converting CSV data to RDF triples, data were published on a SparQL endpoint which makes it possible to perform SparQL queries on the dataset. Figure 15 presents a SparQL query for selecting cities served by Company line Tam.

```
SELECT DISTINCT ? c i t y WHERE {
? s passim : serviceName ?o .
? s passim : ci tyThrough ? c i t y .
FILTER (? o = "TaM")
}
```

Figure 15. SparQL query on Passim RDF data

After making open data in Passim dataset available in RDF format, the next level is connecting Passim dataset to other open datasets in RDF triples available on the web. For linking open datasets, the resources in the dataset have to be linked to the equivalent resources in other datasets. For Passim datasets including fields like name of cities, departments and regions, it is possible to link to other datasets such as DBpedia.

DBpedia is a project that extracts structured information from Wikipedia and makes them available on the web [3]. As of September 2011, DBpedia knowledge base described more than 3.64 million entries, including 416,000 persons, 526,000 places, 106,000 music albums, 60,000 films, 169,000 organizations 5,400 diseases and so on. The DBpedia dataset features, labels and abstracts these 3.64 million entries in up to 97 different languages. As of September 2011, DBpedia dataset consisted of over 1 billion pieces of data presented in RDF triples [30]. Figure 16 illustrates an open dataset of DBpedia with structured piece of data for the city of Innsbruck located in Austria. [2]

Figure 16. DBpedia dataset for the city of Innsbruck in Austria

As DBpedia covers wide range of domains, data publishers increasingly started to set RDF links from their open-license datasets (such as Passim) that are already available on the web to DBpedia. On the other hand, RDF links pointing from DBpedia are published into other web data sources. Thus, it has resulted in the emergence of a web of data around DBpedia [3], as illustrated in Figure 17.

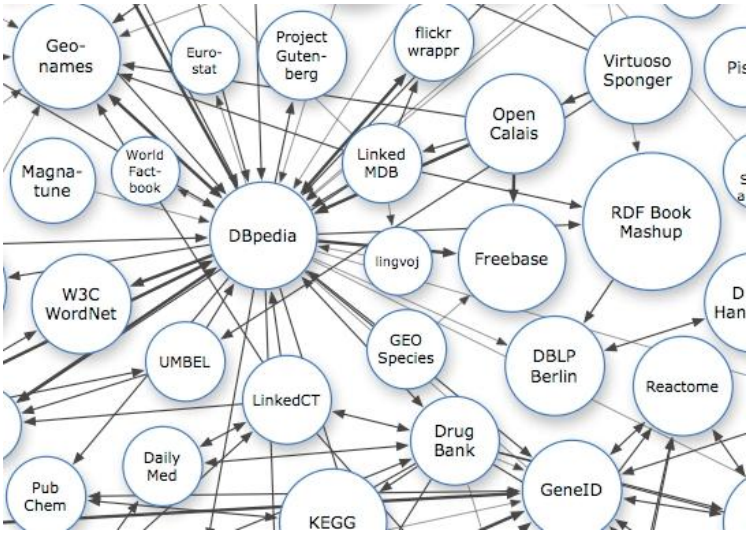

Figure 17. Linking DBpedia with other open datasets using RDF

The more there are open data on the web, the more it is important to link them together. In this way, it will be easier to find more information of the subject we are looking for on the web.

2.7.3 Querying and searching Linked Open Data

In previous sections, the importance of linking open datasets was discussed. In this section, it will be discussed how data in linked open datasets (such as DBpedia) can be searched and queried and how linked open data can be used in application development.

DBpedia is a large scale open dataset contains large amount of general-purpose knowledge. However, finding the right topic in DBpedia and its relationship with relevant topics in other linked datasets is difficult. Therefore, DBpedia has developed tools that help users to find their topics in relevant datasets and to find the relationships between entities existing in linked but separate datasets [3].

Relationship Finder[15] is a user interface that gives the user the possibility of exploring DBpedia knowledge base. Relation Finder allows users to find the connection between two different entities existing in DBpedia datasets. This tool contains a simple form to enter two entries. While user is typing the first entry, he will be offered by other relevant objects can be selected as the second entry. After the query is submitted, the user will be informed whether the connection exists between selected entries. If such a connection exists, user will be able to preview a connection between the objects, which is not necessarily the shortest. However, after that user will be provided with queries can compute the shortest connection exists between the two objects. [3]

Query Builder[16] is another tool allows users to express sophisticated queries on DBpedia datasets using a user friendly interface. Query Builder user interface initially provides a form to the user, containing three fields which should be filled by data about subject, predicate and object of a RDF triple. However, because of the wide coverage of DBpedia, users can hardly know existing relationship between RDF triples which are the base for querying the data in different datasets. Therefore, while user is typing identifier name in one of the fields, matching identifiers for other two fields will be offered for him. This method ensures that the entered identifier for the fields subject, predicate and object are really used in an existing RDF graph pattern, and that query will actually return results. [3]

Moreover, because DBpedia is under a free documentation license on the web, it can be used by client applications. In order to cover the requirements of different client applications, DBpedia is provided on the web through four access mechanism [3]:

- Linked data: Each Resource available in DBpedia is identified by a unique identifier (such as http://dbpedia.org.resource/Berlin). Resource identifiers in DBpedia are set to return RDF descriptions of the resource when they are accessed by Semantic Web search engines and

[15] http://relfinder.dbpedia.org
[16] http://querybuilder.dbpedia.org/

also to return simple HTML view of the same resource searched by a traditional web browser [3].

- SparQL endpoint: Client applications can send their queries over the SparQL protocol to the endpoint[17] which is available on the web. Figure 18 illustrates a simple SparQL query which is written in the SparQL endpoint provided by DBpedia. After running the query all the countries available in DBpedia will be available to the user in selected standard formats such as XML, JSON, CSV, and RDF [3].

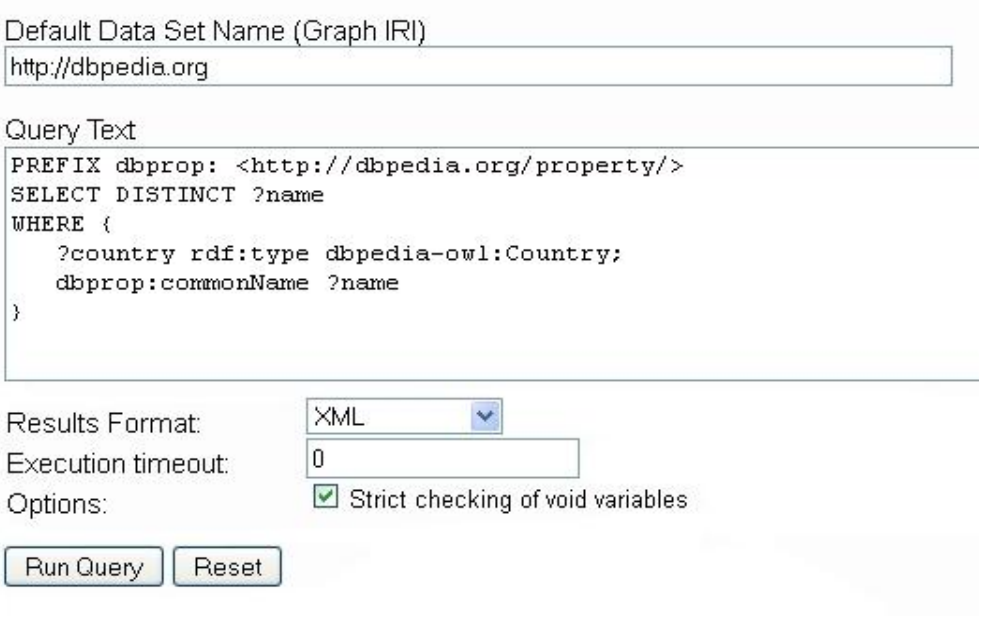

Figure 18. Select query on DBpedia SparQL endpoint

- RDF dumps: This service provides a download page on the web, which offers datasets extracted from Wikipedia editions in 30 languages. These datasets can be used in applications that rely on localized Wikipedia knowledge [3].
- Lookup index: This service is provided for open data publishers to find the most appropriate DBpedia recourse URIs to link their open dataset to [3].

[17] www.dbpedia.org/sparql

Client applications can choose the most efficient access mechanism to DBpedia based on the task they perform [3].

DBpedia Mobile is an application developed using DBpedia locations as navigation coordinates, which allows users to search, discover and publish open data related to their current location which is available in DBpedia. This application can be used in mobile devices such as iphone as well as standard web browsers. Based on the current GPS location of the mobile device, DBpedia Mobile presents an interactive map on the screen, including nearby locations available in DBpedia. [3]

Figure 19. DBpedia Mobile Running on an iphone 3G [3]

Locations will be labeled on the map in addition to a description box showing some information about the location. Clicking any of the location on the map, open data about the location may be retrieved from DBpedia or other linked open datasets using RDF links [3]. Figure 19 illustrates DBpedia Mobile Running on an iphone.

DBpedia Mobile is not limited to fixed available datasets currently existing in DBpedia. This application will be able to use all open datasets which will be linked to DBpedia in future or other data sources that will be reachable from DBpedia [3].

2.8 Summary

As it was discussed in Chapter 1 (Introduction), an open database is needed, which provides up-to-date version of data related to shops and stores in Lappeenranta through standard interfaces for public access. Considering open data cases discussed in Section 2.6, an open data system can be developed based on open data principles, which publishes shops and stores' data in standard formats for public access. Such an open data system can be named Open Data Lappeenranta (ODL). At the beginning, Open Data Lappeenranta can start from publishing standard data related to shops and stores in Lappeenranta, including their opening hours data. However, in the future, Open Data Lappeenranta can publish open data in different fields and topics ranging from finance, social and economic data to data related to public places in Lappeenranta City. Also, published data in Open Data Lappeenranta can be then linked with relevant datasets existing in other projects like DBpedia, adhering to Linked Open Data (LOD) principles.

Similar to what was discussed in Section 2.4 (The benefits of open data), Lappeenranta City, application developers and Lappeenranta citizens, in addition to Russian travelers will benefit from the open data published by Open Data Lappeenranta. City of Lappeenranta provides data (for example data related to shops and stores' opening hours) in standard formats. Providing the data in standard formats for public access will increase trust and community level in the city and will make an open city out of Lappeenranta with transparent government. Also, application developers will obtain free and easy access to shops and stores' data in standard formats for implementing innovative applications based on citizens and travelers' needs. Therefore, Lappeenranta citizens and Russian travelers to this city will enjoy new applications and services which have been developed based on published open data.

To show that it is possible to develop Open Data Lappeenranta, an open data system should be implemented which publishes specific data related to shops and stores in Lappeenranta in standard formats. The open data system is supposed to provide standard interfaces which make stores' data accessible for public. In this case, anybody will be free to access stores' data in standard formats using provided interfaces. In the next section the requirements for implementing such an open data system will be discussed.

2.8.1 Requirements

The open data system is supposed to publish stores' data in a way that everybody can access the data both in human-readable and machine-readable formats. Human-readable version of the data should be presented in an HTML webpage for public access, while machine-readable vision of data is supposed to be used in other services such as CrossBorderTravel.eu web portal. The published data in machine-readable format can be also used for developing innovative web and mobile applications. In fact, reusing machine-readable format of stores' data for developing applications will demonstrate the advantage of open data comparing to closed data.

To implement such an open data system, first, specific data of stores in Lappeenranta, (such as store name, phone number, coordinates and opening hours) should be collected into standard datasets including relevant fields based on what was discussed in Section 2.7.1 (Standard datasets for publishing data). Then, the data in datasets should be available for public access in different levels of data openness based on what was discussed in Section 2.3 (Different levels of openness). In our case, one-star data is supposed to be available in an HTML table, which is human readable; however, to generate machine-processable data format, data is supposed to move toward the third level of data openness which can be used in application development. In fact, the functionality of open data will be demonstrated based on the comparison between one-star data which is human-readable and three-star open data which is machine-readable.

Moreover, the open data system is supposed to make it possible for data owners (shops and stores in Lappeenranta) to update their own data in published datasets when it is necessary. In this case, two different types of interfaces should be provided by the system, manual interface and automatic interface. Manual interface is supposed to be implemented in a way that each shop will be able to access its own data in the published datasets and update the data manually. In contrast, automatic interface is supposed to be implemented in a way that data related to shop X will be updated automatically in published datasets when shop X updates the relevant data in its own database.

In Chapter 4 (Proof of concept), the implementation of such an open data system will be discussed.

3 Development process

Development process in this thesis consists of five parts: Literature review, Summary, Proof of concept, Evaluation and Discussion.

In the first part, Literature review, a background study was conducted for clarifying the current state of open data around the globe. Performing a qualitative research in this part helps the reader to "*gain a deeper understanding of the concept"* [31] of open data. As the result of the literature review, the first two research questions of the thesis were answered. (Why open data is the appropriate solution? What are the existing solutions for making an open data system?).

In Summary part, based on the literature review conducted on open data phenomenon, the idea for creating Open Data Lappeenranta (ODL) was suggested and the requirements for creating such an open data system were identified.

In Proof of concept (constructive part), already selected solutions for creating an open data system will be implemented. As the result of the constructive part, the third research question of the thesis will be answered. (How to implement an open data system in practice?).

Finally, in Evaluation and Discussion parts, the solution (which was selected for creating the open data system) will be evaluated, lessons learned from the constructive part will be discussed and a better idea for creating Open Data Lappeenranta will be proposed.

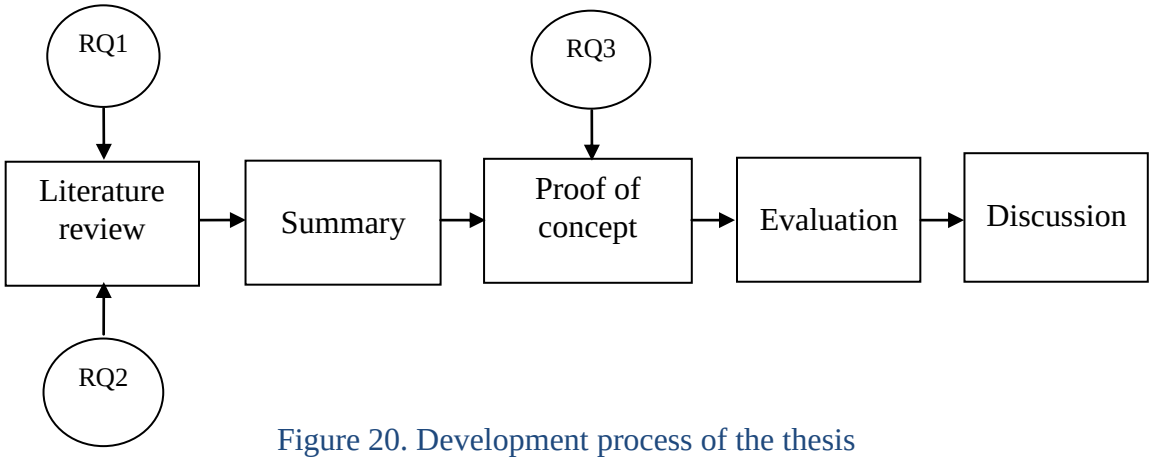

Figure 20. Development process of the thesis

The constructive research which is reported in this thesis will play an important role for encouraging local shops and stores in Lappeenranta to provide their opening hours data for the open data system implemented in this thesis. In fact, when they witness the benefits of open data in practice, they will find it useful to have their data in such an open data system which is implemented adhering to open data principles.

4 Proof of concept

In this chapter, the implementation of the open data system described in Section 2.8 (Summary) will be discussed. The open data system keeps data related to shops and stores in Lappeenranta. Also, this system makes it possible for data owners (shops and stores) to update their own data in the database using either manual or automatic interface. Figure 21 illustrates the open data system which publishes data related to shops and stores in Lappeenranta via standard interfaces. Moreover, the open data system provides updated version of stores' data for public access in two different levels of data openness. One-star data is accessible in an HTML table which is human-readable, while three-star open data which is machine-readable can be accessed by other applications and services such as CrossBorderTravel.eu. The open data system is supposed to publish open data in third level of data openness. However, moving toward four-star and five-star open data (Linked Open Data concepts) will be discussed in Chapter 6 (Discussion). Figure 21 illustrates the open data system with its standard interfaces, which publishes stores' data in standard formats.

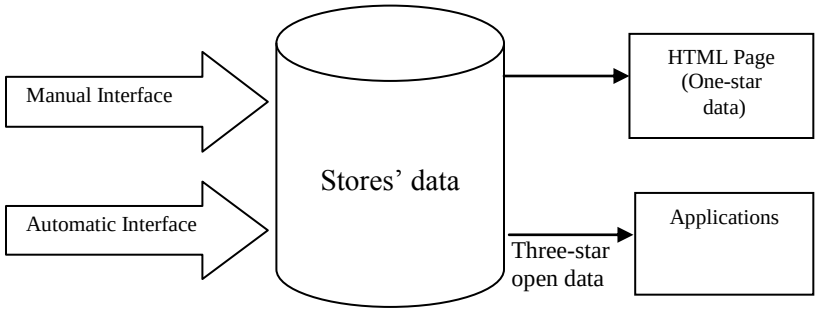

Figure 21. Primitive architecture of the open data system

Open data cases discussed in Section 2.6 (Open data cases developed with different solutions), will help us in this section to select appropriate platform, technologies and solution for implementing the open data system.

4.1 OGDI

4.1.1 OGDI architecture for open data system

To implement the open data system, Open Government Data Initiative (OGDI), was selected as the first design alternative because of what was described in Literature review Chapter about the efficiency and the quickness of this solution for publishing open data.

As it was discussed in Section 2.6.2 (OGDI), OGDI consists of three major components: data loader, data service and data browser. Before discussing the real implementation, it is figured the integration of OGDI components with the open data system which is going to publish stores' data for public access.

In the first step, stores' specific data (store name, opening hours and so on) that are going to be published as open data should be collected in CSV or KML files. KML files are preferred when stores' coordinate data (latitude, longitude) are available. In contrast, stores with no coordinate data can be stored in CSV files.

Using OGDI data loader user interface, CSV or KML files, containing stores' data, will be imported to the Windows Azure cloud platform and will be published on standard datasets with specific fields relevant to imported data. Depending on the imported data, the data loader software utility can either create a new dataset, or update data in a dataset which already exists in the Windows Azure cloud storage. Cloud storage in Windows Azure includes two different storages: configuration storage and data storage. Configuration storage is supposed to keep authentication information which enables data publisher to access the cloud. However, data storage is responsible for keeping the data which has been uploaded by the data publisher.

After publishing data in the cloud, OGDI data service retrieves list of datasets and authentication info from configuration storage, while public data will be retrieved to cloud service from data storage. Later on, OGDI data browser (interactive SDK) which is an ASP.NET web application (supposed to be available in an URL address) allows users to have visualized access to public data which is retrieved from the cloud into data service. As it was discussed in Section 2.6.2

(OGDI), users can query and browse the data in the cloud or have an interactive view of data in tables, charts and maps using data browser user interface.

Moreover, data browser makes it possible for users to receive published data in the cloud in different data formats from human-readable to machine-readable versions. For example HTML data can be received from data browser to create HTML pages to demonstrate one-star data view of available data in the cloud. On the other hand, machine-readable data can be received from data browser in different formats such as CSV, XML and JSON. In this way, three-star open data provided by data service and interactive SDK can be used for developing web and mobile applications.

Figure 22 illustrates OGDI solution for implementing the open data system. In the next section, the implementation of open data system with OGDI solution will be discussed in detail.

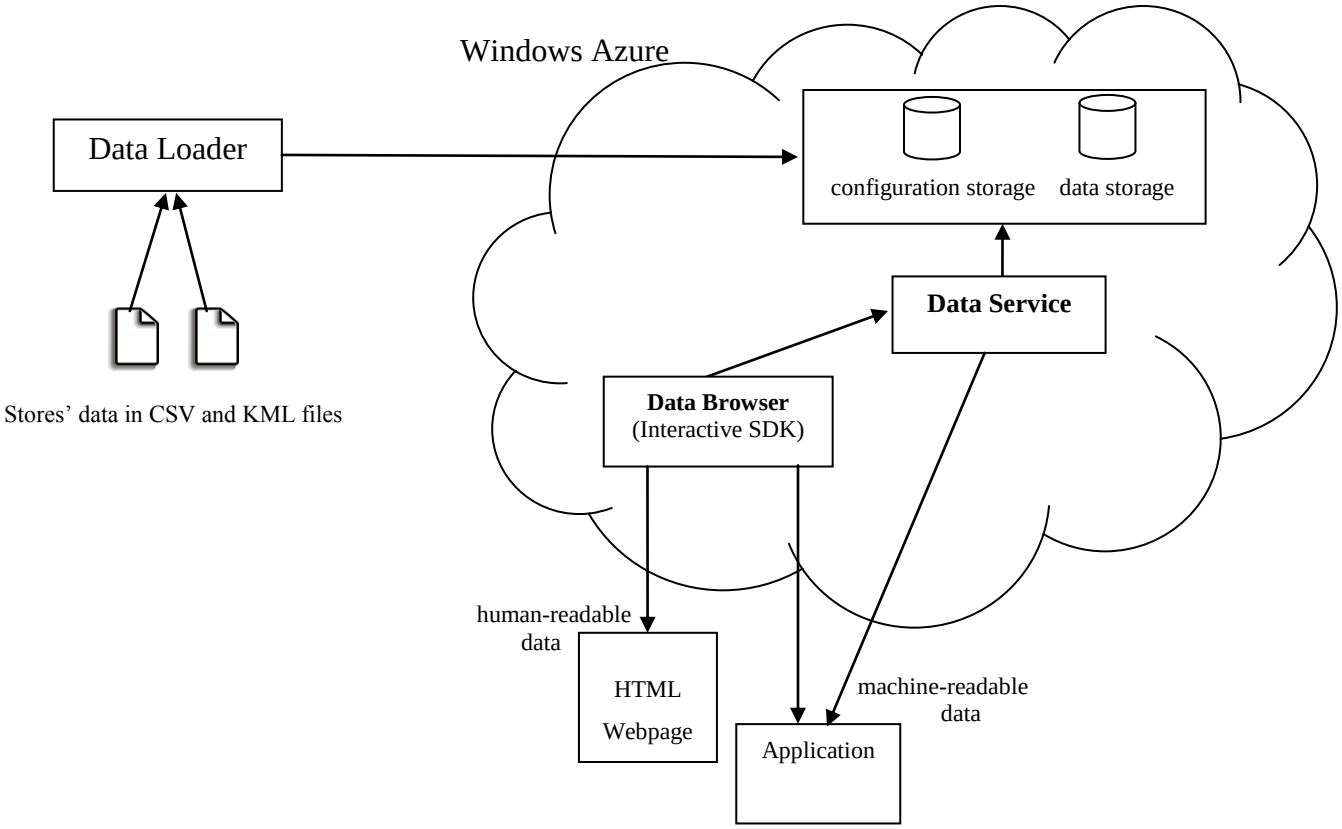

Figure 22. OGDI solution for implementing the open data system [42, 43]

4.1.2 Implementation of OGDI

As in OGDI solution, open data are published on Windows Azure cloud platform, so creating a Windows Azure account was the first step to do in implementation part. Microsoft offers different Windows Azure accounts with variety of services for developing web sites, virtual machines, mobile and cloud services. Customers are charged based on the bandwidth, SQL database and the storage size provided to the selected Windows Azure account. However, a three-month free trial version of Windows Azure cloud platform is also offered by Microsoft, which contains required services for developing the open data system with OGDI. Therefore as the start point in this project, a free of charge three-month trial account of Windows Azure was created.

After creating Microsoft Windows Azure account, a cloud service was created in the account including two different storages: configuration storage and data storage. As it was discussed in 4.1.1, configuration and data storages respectively keep authentication information about published datasets and actual data published in the cloud. Each of the storages was named with a unique identification (storesopeninghoursconfig and storesopeninghoursdata). In the rest of this section, it will be discussed how these unique IDs are used to configure OGDI data browser to the storage in the cloud [33]. Figure 23 presents created cloud service on the Windows Azure account, including configuration and data storages.

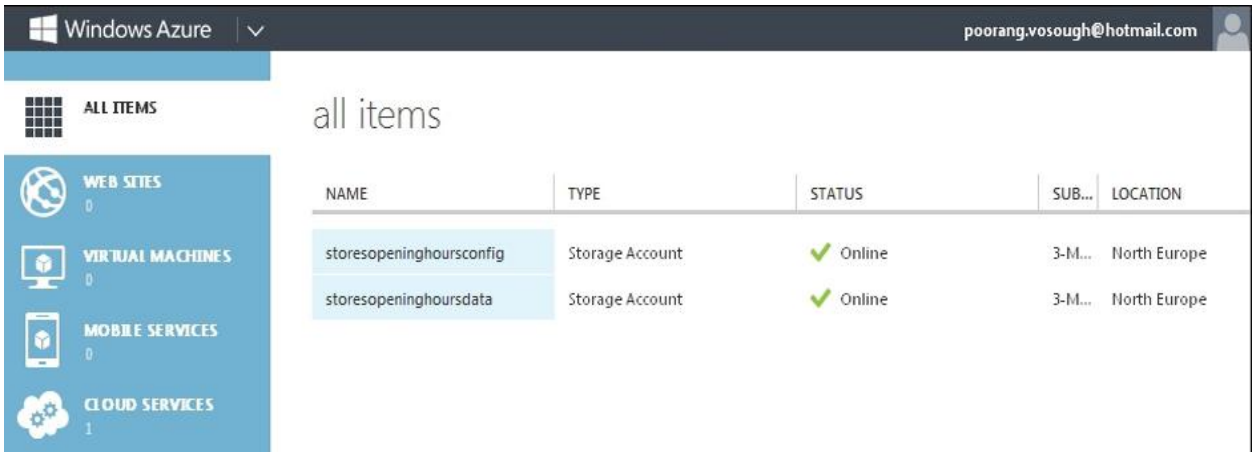

Figure 23. Cloud service on Windows Azure, including configuration and data storages

After creating the cloud service, open source OGDI DataLab version 5 (the latest version in February 2013) was downloaded from GitHub.com[18]. The project folder included sub folders such as data loader, data browser and data service (OGDI components), in addition to a Visual Studio solution file (ogdi.sln), the file for opening OGDI DataLab project in Microsoft Visual Studio.

Before running the project in Visual Studio, OGDI data browser should be configured to the created Windows Azure account and OGDI data service should be running on the cloud. Data browser includes two components, WebRole and WorkerRole, which keeps the unique name of configuration storage created in the cloud. Configuring data browser components with configuration storage name (storesopeninghoursconfig) makes a connection between data browser and the cloud platform which publishes the data. After finishing configuration part, OGDI data service component was run in Visual Studio solution explorer window to create a cloud service in Windows Azure. As it was discussed in 4.1.1, data service is supposed to retrieve published data in the cloud into data browser web application for public access. Figure

[18] https://github.com/openlab/datalab

24 presents created data service (storesopeninghours), which is running on Windows Azure cloud platform.

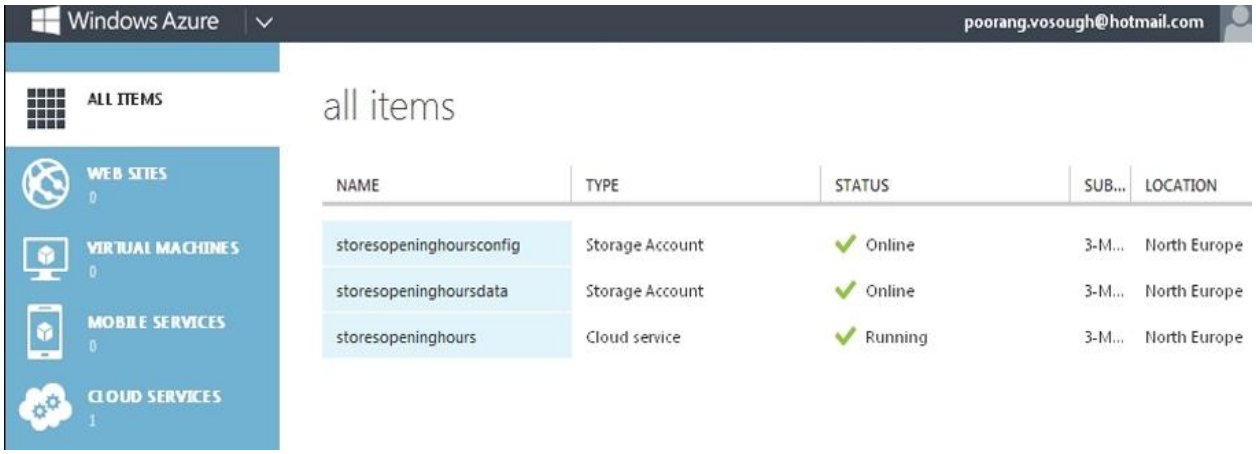

Figure 24. Data service is running on Windows Azure

So far, a Windows Azure could platform was created, OGDI data browser component was configured with the cloud and OGDI data service component run on the cloud. The next level is publishing stores' data (already collected in CSV and KML files) on Windows Azure using OGDI data loader component existing in OGDI DataLab project. However, after importing CSV and KML files containing stores' data, using DataLoaderGuiApp user interface, Visual Studio threw a serialization exception which remained unsolved in spite of considerable effort was done for fixing it. Considering project time limit and consulting with project supervisors, OGDI implementation remained incomplete.

4.2 Open data in JSON format

In this version of implementation, the open data system was developed using Apache web server (version 2.2.21), PHP engine (version 5.3.8) and MySQL database (version 5.5.16) on a Windows 7 platform (service pack 1).

To implement the open data system in this version, stores' data like (store name, store email, phone number, coordinates and opening hours) were collected in a MySQL database named open_database. An HTML webpage created with PHP, presents a human-readable view (HTML table) of stores' data existing in open_database. In fact, HTML table demonstrates one-star data view of stores' data in the database. Moreover, the HTML webpage allows stores in Lappeenranta to access their own data in the system using register and login features provided in the page. New stores can be added to open_database using a register form and data related to existing stores in open_database can be updated using a login form. In fact, register and login forms demonstrate a manual interface to the open data system, which can insert new stores and update existing stores in open_database.

Xmarket demonstrates a local store in Lappeenranta which has a private database (xmarket_database) in its own system. A specific table in xmarket_database is responsible for keeping the data related to Xmarket opening hours. Xmarket data is also kept in open_database located in the open data system. Using an automatic interface, Xmarket will be able to update its opening hours data in open_database. This automation was implemented using a JSON generator/parser. In fact, Xmarket publishes the machine-readable format (JSON) of its opening hours data on the web, which is used by the open data system for automatically updating the same data in open_database.

Similarly, the open data system is supposed to provide open_database data in machine-readable format for public access. The machine-readable format of open_database can be used for developing (pilot) web and mobile applications. In fact, the open data system provides existing data in open_database for public access as three-star open data. Using provided open data in applications will demonstrate the functionality of such an open data system in practice.

Figure 25 presents the open data system which is developed in this version of implementation, using Apache, MySQL, PHP technologies and JSON generator solution.

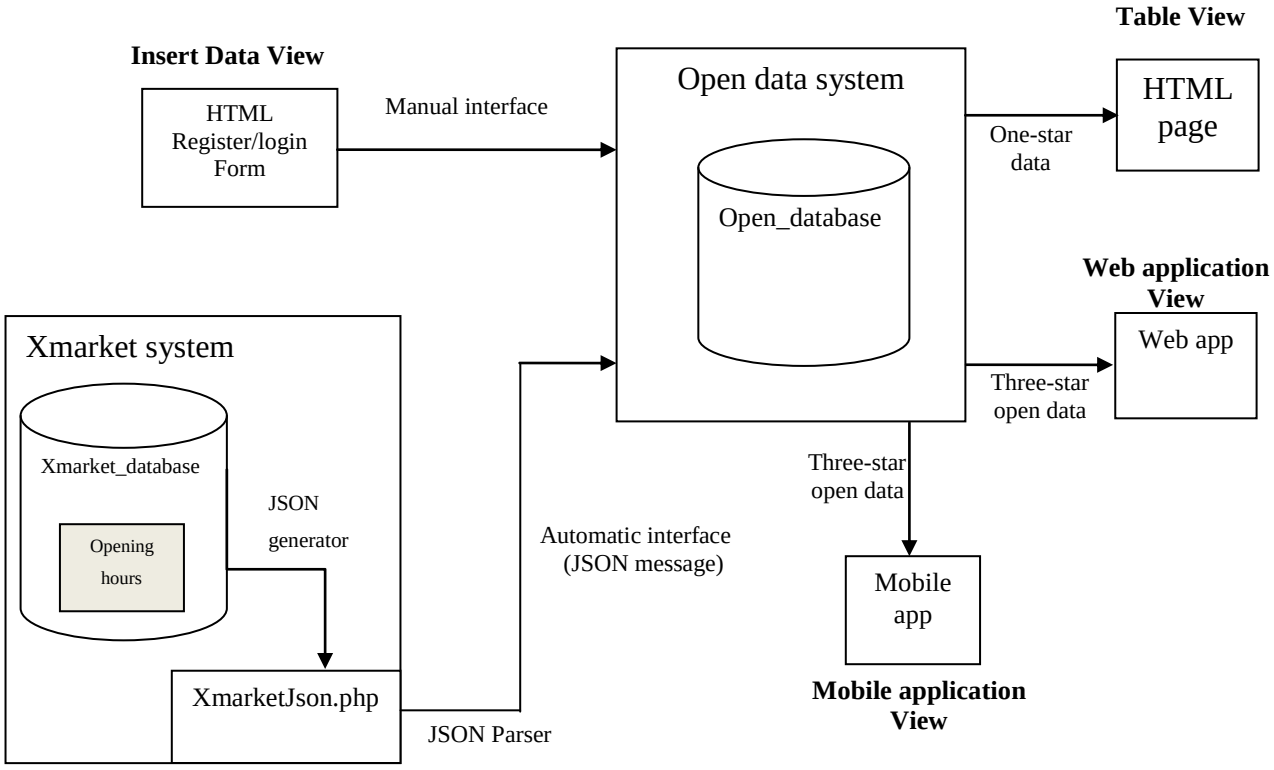

Figure 25. The open data system implemented using Apache, MySQL, PHP and JSON generator

4.2.1 Open data system

Open data system contains a database named open_database which includes two tables: stores_info and login. Stors_info table is designed with specific columns (store_id, store_name, store_email, phone_number, latitude, longitude, mon_open, mon_close, tue_open, tue_close, …, sun_open, sun_close) to keep data related to shops and stores in Lappeenranta. On the other hand, login table consists of columns (store_id, user_name, password) to keep register and login data of the stores which are going to access their own data in open_database using the manual interface. Store_id in login table is considered as a foreign key in stores_info table to relate the two tables together.

Open data system presents a table view of stores data existing in stores_info within a HTML webpage. The HTML table publishes a human readable format of stores' data as one-star data

which is appropriate for human readers. However, the data provided in the HTML table is not very (re)usable, because it is not machine-processable. Figure 26 illustrates one-star data published by open data system in an HTML webpage.

Store Name	Email	Phone_number	Latitude	Longitude	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Xmarket	xmarket@shops.fi	0465292334	61.054090	28.100599	08:00-16:00	08:00-16:00	08:00-16:00	08:00-16:00	08:00-16:00	10:00-16:00	12:00-16:00
Smarket	smarket@shops.fi	0107620500	61.051925	28.104635	07:00-21:00	07:00-21:00	07:00-21:00	07:00-21:00	07:00-21:00	07:00-18:00	12:00-18:00
Bilterna	bilterna@shops.fi	0207457020	61.040381	28.221017	09:00-19:00	09:00-19:00	09:00-19:00	09:00-19:00	09:00-19:00	09:00-17:00	12:00-18:00
Lidl	lidl@shops.fi	080005435	61.040462	28.219571	09:00-21:00	09:00-21:00	09:00-21:00	09:00-21:00	09:00-21:00	09:00-21:00	12:00-18:00
Kmarket	kmarket@shops.fi	054516888	61.051437	28.103927	07:00-21:00	07:00-21:00	07:00-21:00	07:00-21:00	07:00-21:00	07:00-18:00	12:00-18:00
Prisma	prisma@shops.fi	01023456	61.050291	28.178418	08:00-21:00	08:00-21:00	08:00-21:00	08:00-21:00	08:00-21:00	08:00-18:00	12:00-21:00
Siwa	siwa@shops.fi	0207002915	61.045526	28.106824	07:00-22:00	07:00-22:00	07:00-22:00	07:00-22:00	07:00-22:00	08:00-22:00	10:00-22:00

Figure 26. One-star data published in an HTML table

As Figure 26 presents, there are register and login options on the black bar located in top of the HTML webpage. Register and login options allow the shops and stores in Lappeenranta to access open data system via a manual interface. Using register option, stores can insert their own username and password into login table, via a user interface webpage. The user interface page also allows the stores to insert their relevant data like (store name, email, phone number, coordinates and opening hours) into stores_info table. On the other hand, login option allows the stores to enter their username and password in a user interface page to access their own data in stores_info table and update the data when it is necessary. After each registration and logging in to the open data system, new data will be updated in open_database and will be demonstrated in the HTML table consequently.

4.2.2 Automatic interface

As discussed in 4.2, Xmarket is an imaginary local shop in Lappeenranta. This shop owns its own database named xmarket_database including store_info table. Store_info consists of columns (such as store_name, store_email, phone_number, latitude, longitude, mon_open, mon_close, tue_open, tue_close, …, sun_open, sun_close) to keep relevant data about Xmarket. As Figure 26 presents, open_database in the open data system also keeps Xmarket's data.

Basically, the idea of creating such an imaginary shop like Xmarket is to demonstrate the possibility of an automatic interface from Xmarket system to the open data system, which is supposed to update Xmarket's data in open_database whenever Xmarket updates its own data in xmarket_database. To implement such an automatic interface, Xmarket's data existing in xmarket_database should be converted in a machine readable format (e.g. JSON) and be published on the web. In this case, open data system can access the machine-readable format of Xmarket's data and parse it into open_database. In fact, whenever the open data system is going to demonstrate stores' data from open_database in HTML webpage, it should first automatically update Xmarkets' data in open_database and then demonstrates the HTML webpage. To do this automation, the open data system first retrieves JSON format of Xmarket's data from the web and parses it into SQL data format, updates Xmarket' data in open_database, and finally demonstrates HTML webpage.

Data format is not important as long as it can be processed by machine. JSON format was selected in this implementation, because PHP provides useful functions for encoding and decoding JSON, which facilitates the coding process of generating and parsing data to and from JSON. Also, average time and space spent for encoding and decoding data to and from JSON format is more efficient comparing to similar formats such as XML [16]. PHP code for implementing the JSON generator is available in Appendix 1 to show that it is possible to publish Xmarket's data from its own database for public access in standard interface (JSON format) using less than thirty lines code. In this way, other shops and stores in Lappeenranta will find it safe and secure to update their own data in the open data system using the automatic interface implemented with the same logic.

4.2.3 Application demonstration

As discussed in 4.2.1, the open data system presents stores' data in an HTML table which is human-readable. However, one-star data in an HTML table is not easy to reuse. To demonstrate reusing data provided by the open data system in practice, the open data system is supposed to publish stores' data on the web in machine-readable format which can be used for developing web and mobile applications. In fact, open data system is going to provide three-star open data which can be the base for application development. Developing pilot applications using open data will demonstrate the functionality of such an open data system in practice.

To publish stores' data in open format, JSON (JavaScript Object Notation) was selected as the machine-readable format which can be used for developing applications. However, open data can also be published in other formats such as XML, which can be easily processed and reused in application development. To generate JSON data, specific stores' data (store name, store email, phone number, coordinates and opening hours), which are used in the applications, encoded to JSON format using PHP and published on the web in a local host URL address. Using the URL address, applications access stores' data in JSON format, process it and reuse it to create desired functionalities.

In the rest of this section, first it will be discussed a pilot web application implemented using PHP, Java script and HTML, in addition to applying services provided by Google maps within the code, for demonstrating the benefits of open data in practice. Later on, the mobile version of the application also will be discussed to demonstrate the functionality of using open data for developing mobile applications.

In this version of implementation, the web application was created in a local host URL address. The web application presents a table view of shops and stores in Lappeenranta, which currently are open, in addition to their email address, phone number and working hours in current day. The application also provides a map view which demonstrates currently open shops and stores on a map with red marks based on their coordinate (latitude and longitude) available in polished open data. Clicking on each mark, the application opens a pop-up window containing store name,

phone number and working hours in current day. Figure 27 presents the map view page of the application.

Figure 27. A pilot web application developed using JSON data published by the open data system

There are also Google map's embedded services in this application, such as providing satellite view of the map and possibility to zoom in and zoom out the map to discover more shops and stores available on the map. Clicking on "Table View" option located on top of the page takes the user back to the table view page.

In this version of implementation, URLs used for publishing JSON data and creating web pages, are embedded in a local host address starts with (http://127.0.0.1:8888); however, in actual implementation, 4.3 Integration with CrossBorderTravel.eu, real URL addresses will be used for publishing data and creating web pages.

Using stores' open data provided by the open data system, also an Android mobile application was developed, named OpenStoresLPR. The process of publishing the open data and reusing it in OpenStoresLPR is almost the same process which was described for developing the web application. The open data system encodes specific stores' data in JSON format using PHP and publishes the JSON data on the web in a URL address. Using the URL, OpenStoresLPR will access stores' data in JSON format, process them and reuse them to create application functionalities. The functionality of the Android application will be discussed in the rest of this section.

Opening the application, the user finds currently open shops and stores in a list view, in addition to their phone number, email address and working hours in current day. Clicking on the "Map View" button in bottom of the list, locates open shops and stores in Lappeenranta with red marks on a map. Moreover, the user's location is presented on the map with a blue mark. In fact, the map view helps the user to visually find open shops and stores in Lappeenranta city, which have the shortest distance to his or her location. Figure 28 shows both perspectives of the Android application, list view and map view.

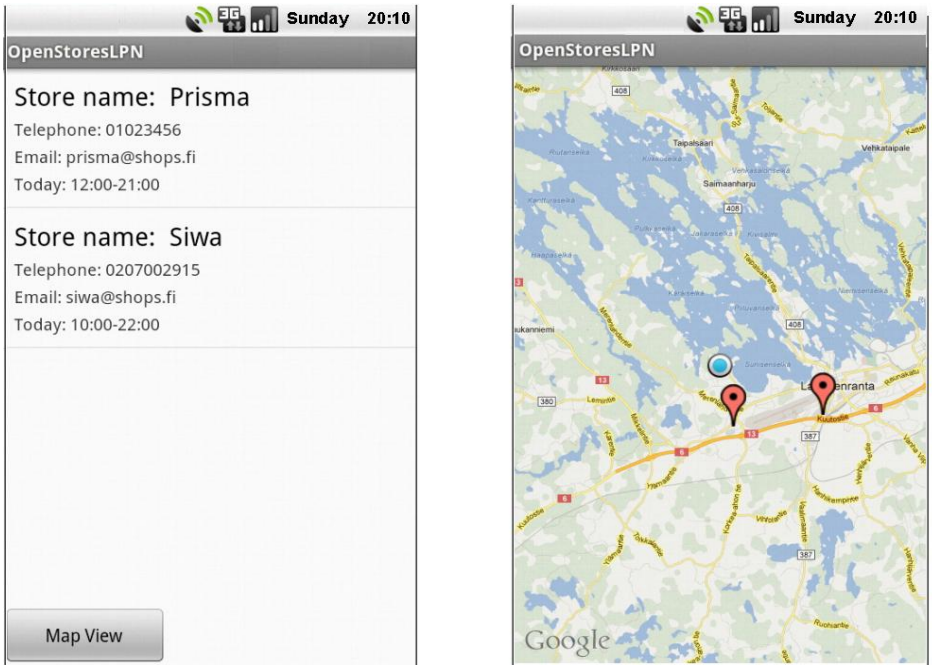

Figure 28. Android application developed using JSON data published by the open data system

Android and the web applications present open stores in Lappeenranta based on current data in the open data system. However, if the data in the open data system is updated using either manual or the automatic interface, data updating can be observed in both applications.

4.3 Integration with CrossBorderTravel.eu

The system implementation discussed in Section 4.2 (Open Data in JSON Format) showed that it is possible to develop such an open data system with manual and automatic interfaces to publish data in both human and machine readable formats. However in that implementation, local host URLs were used to run system web pages and interfaces, Xmarket web page and the application page. Similarly, MySQL databases in the system, like open_database and xmarket_database were also running on a local host server. In this section, the actual implementation of the open data

system will be discussed. Through this version of implementation, every components of the system including databases, web pages and interfaces run on CrossBorderTravel.eu server.

To start the actual implementation, system databases needed to be transferred from the local host server to the server on the web. Therefore, open_database and xmarket_database were transferred to CrossBorderTravel.eu server. Before transferring however, open_database was modified to be compatible with the CrossBorderTravel.eu database which keeps shops and store data. The table in CrossBorderTravel.eu database, keeping stores data includes the fields store id, store name, coordinates and opening hours, in addition to two more fields for keeping shops' category and their home page addresses. Therefore, two more columns (category and home_page) were added to stores_info table in open_database. Also, column types related to opening hours in stores_info table was modified in a way that support shops and stores in CrossBorderTravel.eu, which have null value in their opening hours columns. Nevertheless, xmarket_database keeping data related to Xmarket, transferred to CrossBorderTravel.eu server without any special modification.

After running open_database and xmarket_database on CrossBorderTravel.eu server, the web pages transacting data to or from these databases also transferred from local host URLs to server URL addresses provided by CrossBorderTravel.eu. First, HTML web page[19] showing stores' data in a human-readable format was run on the CrossBorderTravel.eu server to demonstrate table view of the data. Then, register and login web pages providing manual interface to the open data system were run on the server to allow shops and stores to insert and update data in the system. On the other hand, Xmarket[20] web page was run on the server to present the home page of an imaginary shop which updates its opening hours data in the open data system using the automatic interface.

Later on, web pages responsible for publishing three-star open data in JSON format were run on the server. For example, the web pages were supposed to produce data in machine-readable

[19] http://crossbordertravel.eu/od/opendatasystem/opendatasystem
[20] http://crossbordertravel.eu/od/xmarket/webpage/index

format for developing web[21] and mobile [22]applications run on CrossBorderTravel.eu server, having real URL addresses. Moreover, the web application [23] page was also transferred from local host address to CrossBorderTravel.eu server. Finally, the page which generates Xmarket data in JSON format was also available on the server. Using this page, the open data system automatically updates Xmarket data in open_database.

Now that all system components were implemented on CrossBorderTravel.eu server, the open data system and created web and mobile applications work with data related to real shops and stores in Lappeenranta to demonstrate the functionality of the system in real life. As it was discussed in Introduction Chapter of the thesis, CrossBorderTravel.eu web portal provides information about shops and stores in Lappeenranta (e.g. their locations and working time) to ease travelling for Russian travelers coming to Finland. At the present time (April 2013), CrossBorderTravel.eu keeps data about more than one hundred shops and stores in Lappeenranta and Saint Petersburg in its database. Moreover, CrossBorderTravel.eu publishes specific data of these shops, such as [shop id, shop name, category, home page, coordinates and opening hours] on the web in a machine-readable format (JSON)[24]. Therefore, similar to the automatic interface implemented for updating Xmarket data in the open data system, stores' data in CrossBorderTravel.eu available on the web in JSON format were parsed and inserted into open_database.

After automatic data inserting from CrossBorderTravel.eu database, the open data system publishes data related to more than one hundred shops and stores in JSON format. Using published open data, developed pilot applications present shops and stores are currently open in Lappeenranta and Saint Petersburg. Moreover, shops and stores can use both manual and automatic interfaces to update their data in the open data system. In this case,

[21] http://crossbordertravel.eu/od/opendatasystem/json_open_stores_webapp_table
[22] http://crossbordertravel.eu/od/opendatasystem/json_open_stores_android
[23] http://crossbordertravel.eu/od/webapp/tableview
[24] http://crossbordertravel.eu/json/shops/everything

CrossBorderTravel.eu will be able to update data related to shops and stores in Lappeenranta and Saint Petersburg using JSON data published by the open data system.

5 Evaluation

As discussed in 4.2, the open data and Xmarket systems including interfaces and databases were implemented using WAMP package consist of Apache web server, PHP engine and MySQL database on a Windows platform. MySQL is an appropriate alternative for keeping small scale of data (like names, coordinates and opening hours) of shops and stores in Lappeenranta. Moreover, to keep stores' data and to publish the data on the web in open format, WAMP is a convenient alternative as it is an open source solution.

To publish stores' data in open format, and to implement the automatic interface for the open data system, data serialization method was used. Data serialization is the process of converting data to a stream, and rebuilding the stream back into an object. XML and JSON are two most common modern data serialization formats. However, JSON is often touted as a lightweight and more efficient alternative for XML [16]. Thus, JSON format efficiency was the first reason that this machine-readable format was selected as the base of data serialization for publishing open data and creating the automatic interface. The second reason was that there are PHP functions which facilitate encoding and decoding data to and from JSON (Section 4.2.2).

Using data serialization, it was implemented the automatic interface which updates Xmarket data into the open data system in a secure way. The automatic interface was implemented by only thirty lines of PHP code which select relevant Xmarket data (like store name, category, home page, coordinates and opening hours) from xmarket_database and serialize SQL data to JSON and publish them on the web (See Appendix 1). Data serialization in JSON format does not involve risks like system hacking, SQL injection, data loss. In this way, it increases the possibility that local shops and stores in Lappeenranta will be convinced to use such a safe and reliable interface to automatically update their own data into the open data system. However, the manual interface allows the shops and stores, which cannot or do not want to use the automatic interface, to insert or update their data in the open data system. Comparing automatic and manual interfaces demonstrates the automated services which can be developed using machine-readable data formats like JSON rather than inserting the data manually via HTML pages.

Similarly, benefits of open data are demonstrated comparing the pages presenting stores' data in the open data system in human-readable[25] and machine-readable[26] formats. The stores' data presented in the HTML table is much easier to read rather than the same data in JSON format for human. However, the data in HTML table cannot be easily processed and reused in other applications and services, while the same data serialized to JSON format were used for developing pilot and mobile applications because of its machine-readable feature.

[25] http://crossbordertravel.eu/od/opendatasystem/opendatasystem
[26] http://crossbordertravel.eu/od/opendatasystem/machine-readable/clickJSON

6 Discussion

In Chapter 2 (Literature review), the benefits of open data were discussed and the state of open data phenomenon was tracked around the world. Also, open data cases (such as Open Data Albania and Open Data Niagara) were studied to identify different open data solutions for publishing data in standard formats. In Chapter 4 (Proof of concept), to show the benefits of open data in practice, the implementation of an open data system for publishing data related to shops and stores in Lappeenranta was discussed. In this chapter (Discussion), based on the concepts discussed in literature review and proof of concept chapters, Open Data Lappeenranta (ODL), is proposed. As it was mentioned in Section 2.8 (Summary), at the beginning, Open Data Lappeenranta can start from publishing standard data related to shops and stores in Lappeenranta. However, in the future, Open Data Lappeenranta can publish variety of open datasets in standard formats for public access, including data related to government agencies, hospitals, transportation, sports and so on.

To publish stores' opening hours as open data, each shop in Lappeenranta is supposed to have its own dataset in Open Data Lappeenranta with specific fields like (store id, store name, category, home page, latitude, longitude and opening hours). However in next steps, when Open Data Lappeenranta is extended to other fields and topics, depending on the field and topic the data is published about, Open Data Lappeenranta is supposed to provide standard datasets with specific placeholders to keep relevant data. For example, open datasets which keep data about hospitals in Lappeenranta, are supposed to include fields such as (hospital id, hospital name, region, address and coordinates data), similar to Niagara Region flu clinics discussed in 2.6.2. In contrast, open datasets keeping the bus transportation data in Lappeenranta are supposed to have fields like (bus id, bus line, stop id, stop name, stop coordinates, working time and so on), similar to Open Transport Data in France discussed in 2.7.1.

Open Data Lappeenranta is supposed to publish open data in different levels of openness. Similar to open data system implementation (discussed in Chapter 4), one-star and two-star data are published in a human-readable format (e.g. HTML and Excel tables), while three-star data which

are machine-readable, are published in a way that can be used in other applications and services. Moreover, Open Data Lappeenranta can make it possible for users to download data in different openness levels. For example, one-star and two-star data can be downloaded by users as Excel and PDF files, while three-star open data can be downloaded in formats such as XML, JSON, and CSV. Downloading data in standard formats gives the users the possibility to analyze data (e.g. financial open data) or use techniques like data mining to translate data in useful knowledge which can provide various types of support in decision making [22].

Open Data Lappeenranta can also publish open datasets in multiple languages (similar to DBpedia) considering the topic the data is about. For example, in open data related to stores' opening hours, there is no need for publishing data in different languages as publishing data are numeric. But, when open data is related to information beyond numeric data (e.g. information about public places in Lappeenranta City), providing data in multiple languages can be the useful for creating applications for users who want to receive services in a particular language (like Russians). In this case, applications like DBpedia Mobile, discussed in Section 2.7.2 (Linking open datasets), can be developed for mobile devices, which labels public places in Lappeenranta on the map in addition to a description box showing information about the location. If Open Data Lappeenranta publishes open datasets in multiple languages, users will be able to view description box in different languages such as English, Finnish and Russian.

After publishing open datasets on the web, the next step is leading Open Data Lappeenranta toward five-star open data, applying Linked Open Data (LOD) principles to the data published. For this purpose, open datasets in ODL should be converted to RDF triples (subject, predicate and object). Publishing open data in RDF triples makes it possible to link datasets in ODL with relevant datasets existing in other projects like DBpedia. In this case, standard data provided by ODL will be discovered by people using DBpedia. Similarly, standard data in DBpedia datasets will be discovered by people having access to ODL. In this way, if there is already a definition for each city or each shop in DBpedia, that definition can be reused rather than creating a new definition for the same object in ODL. This process prevents creating objects in open datasets, which describe the same reality. Also, providing open datasets in RDF triples enables client applications to query and browse open datasets in ODL linked with other datasets via query

interfaces such as SparQL endpoint. Moreover, users will be able to have a visualized access to SparQL queries from ODL open datasets linked with other datasets like DBpedia, through visualization tools.

Figure 28 illustrates the architecture of Open Data Lappeenranta. As the architecture presents, Lappeenranta City provides data to Open Data Lappeenranta. Provided data from Lappeenranta City can be in different fields and topics ranging from shops and stores opening hours to data related to health centers and public places in Lappeenranta City. Regardless of the structure of the data in its original data source, ODL database keeps provided data in structured formats. Open Data Lappeenranta then, publishes data in different levels of openness from one-star data to five-star linked open data.

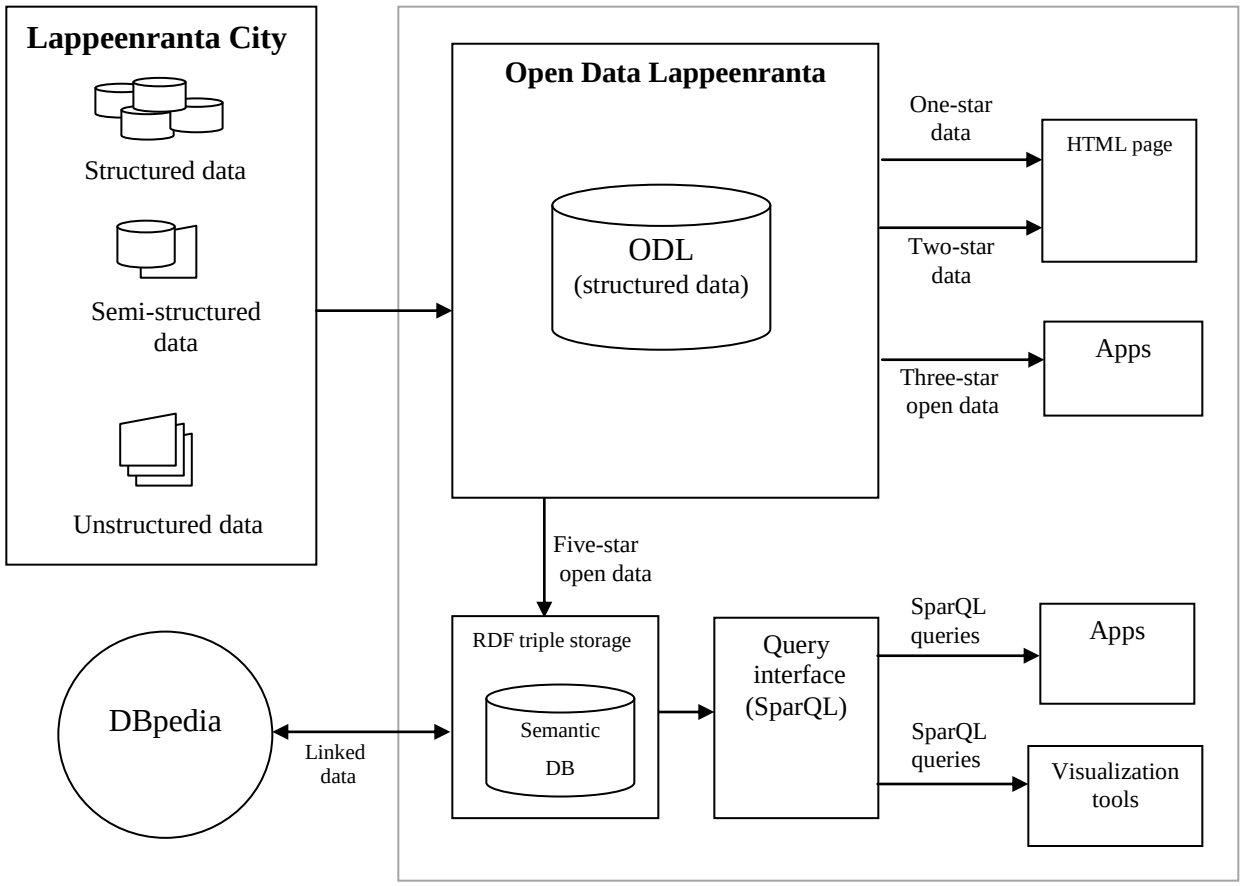

Figure 28. Open Data Lappeenranta (ODL) architecture

Open data published by ODL impacts different aspects of Lappeenranta City. ODL can improve businesses in Lappeenranta. Considering published open data related to shops and stores in Lappeenranta, Russian or Finnish customers can easily find the location and working time of shops and stores and their categories (similar to the open data system implementation). In this case, shops and stores in Lappeenranta will have more customers and more sells. So, open data published by ODL can be of value for Lappeenranta City. When businesses grow in Lappeenranta, more taxes are paid to Lappeenranta City and there will be more job opportunities for citizens. Also, application developers will obtain free and easy access to standard data published by ODL in various fields and topic vary from stores' opening hours to data related to public places in Lappeenranta for implementing innovative applications based on citizens and travelers' needs. In this case, Lappeenranta citizens and travelers will be provided with new applications and services created by developers.

7 Conclusion

In this thesis, different open data cases were discussed, which were developed in countries such as US, Brazil, Albania and France. In each case, depending on the purpose of the project, data was published in various data formats (such as XML, JSON, KML and so on) using different solutions for publishing data (like W3C standards and OGDI). Regardless of the data formats and publishing solutions, in all open data cases, the goal is publishing data for public access in standard formats which can be processed by machines or software agents. Published data in standard formats can be used for developing innovative services and applications to ease peoples' lives.

The open data system implemented in this project publishes data related to shops and stores in Lappeenranta in both human-readable (HTML table) and machine-readable format (JSON). The comparison between the two data formats demonstrates that the data published in machine-readable format can be used as the base of creating new web and mobile applications while human-readable data in an HTML table is difficult to reuse although it is easy-to-read for humans. Similar to developed pilot applications, CrossBorderTravel.eu web portal can also use the published JSON data to automatically update shops and stores' opening hours in its database.

In this project, JSON was selected as the machine-readable format for publishing stores' data. The reason for selecting JSON was the advantages of this format comparing to other machine-readable formats for publishing data (like XML). JSON is a text-based format and can be written and parsed on any platforms including Android platform, which was used as the platform for developing OpenStoresLPR application. Also, JSON time and size efficiency in data serialization and deserialization makes this data format an appropriate alternative for developing mobile applications [16].

REFERENCES

[1] Hoxha, J. & Brahaj, A. 2011. Open Government Data on the Web: A Semantic Approach. International Conference on Emerging Intelligent Data and Web Technologies (EIDWT), 7-9 September, 2011, Karlsruhe, Germany. pages 107 – 113.

[2] Machado, A. L. & De Oliveira, J. M. P. 2011. DIGO: An Open Data Architecture for e-Government. 15th IEEE International Enterprise Distributed Object Computing Conference (EDOCW), 29 August – 2 September 2011, Sao Paulo, Brazil. pages 448 – 456.

[3] Bizer, Ch., Lehmann, J., Kobilarov, G., Auer, S., Becker, Ch., Cyganiak, R. & Hellmann, S. 2009. DBpedia - A crystallization point for the Web of Data. Jurnal of WebSemantics, vol. 7, issue. 3, pages 154–165.

[4] Bizer, C. 2009. The Emerging Web of Linked Data. IEEE Journal of intelligent systems, vol. 24, Issue. 5, pages 87-92.

[5] Oliver, M., Palacin, M., Valls, V. & Domingo, A. 2012. Sensor Information Fueling Open Data. IEEE 36th Annual Computer Software and Applications Conference Workshops (COMPSACW), 16-20 July 2012, Barcelona, Spain. pages 116-121.

[6] Hausenblas, M. & Karnstedt, M. 2010. Understanding Linked Open Data as a Web-Scale Database. Second international conference on Advances in Databases Knowledge and Data Applications (DBKDA), 11-16 April 2010, Galway, Ireland. page 56 – 61.

[7] Lee, G. & Kwak, Y. H. 2011. Open Government Implementation Model: a Stage Model for Achieving Increased Public Engagement. Proceedings of the 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging Times (dg.o '11). June 2011, New York , USA. pages 254-261.

[8] Hoxha, J. 2011. Open.data.al: increasing the utilization of government data in Albania, Proceedings of the 7th International Conference on Semantic Systems (I-Semantics '11), September 2011, New York , USA. pages 237-240.

[9] Böhm, Ch., Naumann, F., Freitag, M., George, S., Höfler, N., Köppelmann, M., Lehmann, C., Mascher, A. & Schmidt, T. 2010. Linking Open Government Data: What Journalists Wish They Had Known. Proceedings of the 6th International Conference on Semantic Systems (I-SEMANTICS '10), September 2010, New York, USA.

[10] Plu, J. & Scharffe, F. 2012. Publishing and linking transport data on the Web. Proceedings of the First International Workshop on Open Data (WOD '12). May 2012, New York, USA. pages 62-69.

[11] Hendler, J., Holm, J., Musialek, C. & Thomas, G. 2012. US Government Linked Open Data: Semantic.data.gov. IEEE Journal of Intelligent Systems, vol. 27, no. 3, pages 25-31

[12] Breitman, K., Salas, P., Casanova, M.A., Saraiva, D., Viterbo, J., Magalhães, R.P., Franzosi, E. & Chaves, M. 2012. Ope government data in Brazil. IEEE Journal of Intelligent Systems, vol 27, no.3, pages 45-49

[13] Alkhateeb, F., Baget, G. F. & Euzenat, F. 2009. Extending SPARQL with regular expression patterns (for querying RDF). Journal of Web Semantics: Science, Services and Agents on the World Wide Web, vol. 7, Issue. 2, pages 57-73.

[14] Bizer, C., Heath, T. & Berners-Lee, T. 2009. Linked Data - The Story So Far. International Journal on Semantic Web and Information Systems (IJSWIS). vol. 5 issue. 3.

[15] Cyganiak, R., Maali, F. & Peristeras, V. 2010. Self-service linked government data with dcat and gridworks, Proceedings of the 6th International Conference on Semantic Systems (I-Semantics '10), September 2010, New York , USA. article no. 37.

[16] Sumaray, A. & Makki, K. 2012. A comparison of data serialization formats for optimal efficiency on a mobile platform, Proceedings of the 6th International Conference on Ubiquitous Information (ICUIMC '12), February 2012, New York , USA. article no. 48.

[17] Urlik Andersen, Ch. & Bro Pold, S. 2012. Occupation of the Open City. Proceedings of the 4th Media Architecture Biennale Conference (MAB ’12), November 2012, New York, USA. pages 1-4.

[18] Cerri, D. & Fuggetta, A. 2007. Open standards, open formats, and open source. Journal of Systems and Software, vol. 80, Issue. 11, pages 1930-1937.

[19] Lapi, E., Tcholtchev, N., Bassbouss, L. & Marienfeld, F. 2012. Identification and Utilization of Components for a Linked Open Data Platform. IEEE 36th Annual Computer Software and Applications Conference Workshops (COMPSACW), July 2012, Izmir. pages 112-115

[20] Friberger, M.G. & Togelius, J. 2012. Generating interesting Monopoly boards from open data. IEEE Conference on Computational Intelligence and Games (CIG), September 2012, Granada. pages 288-295.

[21] Guang-Jie Ren, Glissmann, S. 2012. Identifying Information Assets for Open Data: The Role of Business Architecture and Information Quality.14th IEEE International Conference on Commerce and Enterprise Computing (CEC). September 2012, Hangzhou. pages 94-100.

[22] Milić, P., Veljković, N. & Stoimenov, L. 2012. Framework for open data mining in e-government. Proceedings of the Fifth Balkan Conference in Informatics (BCI '12). September 2012, New York City. pages 255-258.

[23] Ma, L., Su, Z., Pan, Y., Zhang, L. & Liu, T. 2004. RStar: an RDF storage and query system for enterprise resource management. Proceedings of the thirteenth ACM international conference on Information and knowledge (CIKM '04). November 2004, New York City. pages 484-491.

[24] Hausenblas, M. 2011. Utilising linked open data in applications. Proceedings of the International Conference on Web Intelligence, Mining and Semantics (WIMS '11). May 2011, New York City. article no. 7.

[25] Berners-Lee, T. 2009. Linked Data [online document]. [Accessed 26 February 2013]. Available at http://www.w3.org/DesignIssues/LinkedData.html

[26] Zimmermann, A. 2011. Leveraging the Linked Data Principles for Electronic Communications. Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology (WI-IAT '11). August 2011, Washington DC, vol. 3, pages 385-388.

[27] Wikipedia. 2013. Linking open-data community project [online document]. [Accessed 10 March 2013]. Available at http://en.wikipedia.org/wiki/Linked_data#Linking_open-data_community_project

[28] Open Government Canada. 2013. Open Government Consultation [online document]. [Accessed 26 February 2013]. Available at : http://www.open.gc.ca/index-eng.asp

[29] Tychon, G. G. 2013. Open Data Principles [online document]. [Accessed 25 February 2013]. Available at http://datatogo.xspatial.com/?q=content/open-data-principles

[30] Wikipedia. 2013. DBpedia [online document]. [Accessed 7 April 2013]. Available at https://en.wikipedia.org/wiki/DBpedia

[31] Järvinen, P. 2004. On Research Methods. Tampere: Opinpajan Kirja.

[32] Myers, M. D. 2008. Qualitative Research in Business & Management. London: Sage.

[33] The MyOpenCity.ca Team. 2012. Installing OGDI DataLab version 5 on Azure [online document]. [Accessed 15 February 2013]. Available at http://myopencity.ca/wp-content/uploads/2012/09/Installing-OGDI-DataLab-version-5-on-Azure-v.1.pdf

[34] CodePlex-Project Hosting for Open Source Software. 2012. OGDI [online document]. [Accessed 25 February 2013]. Available at http://ogdi.codeplex.com

[35] Open Cities Project. 2013. Open Cities [website]. [Accessed 5 May 2013]. Available at http://www.opencities.net/

[36] IBM. 2011. IBM'sSmarter Cities Challenge–Helsinki Report. [online document]. [Accessed 15 February 2013]. Available at http://smartercitieschallenge.org/executive_reports/SCC_ExecutiveSummary_Helsinki.

[37] CKAN. 2013. The open source data portal software. [web site]. [Accessed 5 May 2013]. Available at http://ckan.org

[38] Open Government Data Initiative. 2012. For Developers [online document]. [Accessed 1 March 2013]. Available at http://openregina.cloudapp.net/Developers/Index

[39] Microsoft. 2013. Server and Cloud Platform [online document]. [Accessed 1 March 2013]. Available at http://www.microsoft.com/en-us/server-cloud/windows-azure.aspx

[40] Open Government Data Initiative (OGDI). 2013. Welcome to Open Government Data Initiative (OGDI) [online document]. [Accessed 1 March 2013]. Available at http://datadotgc.cloudapp.net

[41] Chignard, S. 2013. A Brief History of Open Data. [online document]. [Accessed 6 May 2013]. Available at http://www.paristechreview.com/2013/03/29/brief-history-open-data

[42] GitHub. 2013. OGDI DataLab v5 Projects Topology [online document]. [Accessed 2 March 2013]. Available at https://github.com/openlab/DataLab/wiki/OGDI-DataLab-v5-Projects-Topology

[43] Curry, K. 2010. Open Government Data Initiative in 7 Slides [online document]. [Accessed 1 March 2013]. Available at http://www.slideshare.net/kmcurry/open-government-data-initiative-in-7-slides.

[44] RDF working group. 2004. Resource Description Framework (RDF) [online document]. [Accessed 5 May 2013]. Available at http://www.w3.org/RDF/

[45] The City of Waterloo. 2012. Open Data Waterloo: open government data to benefit the community [online document]. [Accessed 26 February 2013]. Available at http://www.waterloo.ca/en/opendata/learnaboutopendata.asp#What%20is%20Open%20Data?

[46] Open government data. 2013. Welcome to Open Government Data [online document]. [Accessed 28 February 2013]. Available at http://opengovernmentdata.org

[47] Statistics Finland. 2011. Tourism from abroad to Finland increased in 2010 [onlinedocument]. [Accessed 20 May 2013]. Available at http://www.stat.fi/til/rajat/2010/rajat_2010_2011-06-15_tie_001_en.html

[48] Roberts, T. 2012. The problem with open data [online document]. [Accessed 27 February 2013]. http://www.computerweekly.com/opinion/The-problem-with-Open-Data

Appendix 1

This is JSON generator PHP code, which converts specific Xmarket data (store_name, category, home_page, latitude, longitude and opening hours) to JSON format.

```
<?php

//database configuration
$config['mysql_host'] = "**************";
$config['mysql_user'] = "**************";
$config['mysql_pass'] = "**************";
$config['db_name']    = "xmarket_database";
$config['table_name'] = "xmarket_info";

//connect to host
$connect=mysql_connect($config['mysql_host'],$config['mysql_user'],$config['mysql_pass']);
//select database
@mysql_select_db($config['db_name']) or die( "Unable to select database");

$sql = "SELECT store_name,category,home_page, latitude, longitude,
                        mon_open, mon_close,tue_open,tue_close,
                        wed_open,wed_close,thu_open,thu_close,
                        fri_open,fri_close,sat_open,sat_close,sun_open,sun_close
                        FROM ".$config['table_name'];


//encode Xsupermarket data to Json
 $result = mysql_query($sql);
 $rows = array();
   while($r = mysql_fetch_assoc($result)) {
     $rows[''][] = $r;
   }
// prining the Json data in HTTP
 print json_encode($rows);


 //close connection
 mysql_close($connect);

?>
```