Linked Data Application Development Methodology

PhD Thesis

by

Milos Jovanovik

"Ss. Cyril and Methodius" University in Skopje
FACULTY OF COMPUTER
SCIENCE AND ENGINEERING

Linked Data Application Development Methodology

Executive Summary
of the PhD Thesis

Milos Jovanovik

Supervisor:
Prof. Dimitar Trajanov, PhD

Skopje, November 2016

Abstract

The vast amount of data available over the distributed infrastructure of the Web has initiated the development of techniques for their representation, storage and usage. One of these techniques is the Linked Data paradigm, which aims to provide unified practices for publishing and contextually interlinking data on the Web, by using the World Wide Web Consortium (W3C) standards and the Semantic Web technologies. This approach enables the transformation of the Web from a web of documents, to a web of data. With it, the Web transforms into a distributed network of data which can be used by software agents and machines. The interlinked nature of the distributed datasets enables the creation of advanced use-case scenarios for the end users and their applications, scenarios previously unavailable over isolated data silos. This creates opportunities for generating new business values in the industry.

The adoption of the Linked Data principles by data publishers from the research community and the industry has led to the creation of the Linked Open Data (LOD) Cloud, a vast collection of interlinked data published on and accessible via the existing infrastructure of the Web. The experience in creating these Linked Data datasets has led to the development of a few methodologies for transforming and publishing Linked Data. However, even though these methodologies cover the process of modeling, transforming / generating and publishing Linked Data, they do not consider reuse of the steps from the life-cycle. This results in separate and independent efforts to generate Linked Data within a given domain, which always go through the entire set of life-cycle steps.

In this PhD thesis, based on our experience with generating Linked Data in various domains and based on the existing Linked Data methodologies, we define a new Linked Data methodology with a focus on reuse. It consists of five steps which encompass the tasks of studying the domain, modeling the data, transforming the data, publishing it and exploiting it. In each of the steps, the methodology provides guidance to data publishers on defining reusable components in the form of tools, schemas and services, for the given domain. With this, future Linked Data publishers in the domain would be able to reuse these components to go through the life-cycle steps in a more efficient and productive manner. With the reuse of schemas from the domain, the resulting Linked Data dataset will be compatible and aligned with other datasets generated by reusing the same components, which additionally leverages the value of the datasets.

This approach aims to encourage data publishers to generate high-quality, aligned Linked Data datasets from various domains, leading to further growth of the number of datasets on the LOD Cloud, their quality and the exploitation scenarios. With the emergence of data-driven scientific fields, such as Data Science, creating and publishing high-quality Linked Data datasets on the Web is becoming even more important, as it provides an open dataspace built on existing Web standards. Such a dataspace enables data scientists to make data analytics over the cleaned, structured and aligned data in it, in order to produce new knowledge and introduce new value in a given domain. As the Linked Data principles are also applicable within closed environments over proprietary data, the same methods and approaches are applicable in the enterprise domain as well.

Keywords: Linked Data, Data Science, Methodology, Reuse, Methods, Tools, Open Data, Semantic Web.

Background and Motivation

One especially active research field in the last decade has been the field of data management: data representation, storage and access. The vast amount of data available on the Web has initiated the development of techniques for managing data distributed across its existing infrastructure. One of these techniques is the Linked Data paradigm, which aims to enable uniform practices for publishing interlinked data on the Web by using the technologies of the Semantic Web [48][47][53]. This enables the transformation of the classic Web from a web of documents into a web of data, which is in line with the original idea of the Semantic Web [44]. With the transformation into a distributed network for data access, the Web can be used by software agents and machines.

The interlinked nature of these distributed datasets represents the basis for the development of advanced use-case scenarios for the end users and their applications, scenarios previously unavailable over isolated datasets. Further, these scenarios represent a solid foundation for the development of new business solutions from both the ICT industry and independent developers. They can develop innovative applications and services for the end users, creating new business value in the industry and in the society [45][68].

The adoption of the Linked Data Principles [43] by a large number of companies, research centers and institutions from around the world, has led to the creation of a global data space with interconnected data from different domains, such as people, companies, books, publications, movies, music, TV and radio programs, gene information, proteins, drugs, clinical trials, online communications and social media, statistical and scientific data, etc. [53]. This data network is called the Linked Open Data (LOD) Cloud - a vast network of datasets published and interconnected according to the Linked Data Principles. The data from this network are available via the existing infrastructure of the Web (Fig. 1).

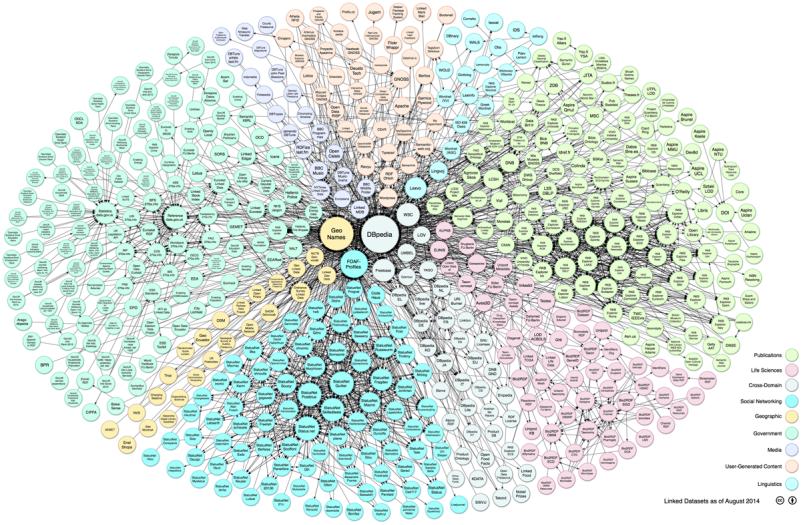

Figure 1: Linking Open Data cloud diagram 2014, by Max Schmachtenberg, Christian Bizer, Anja Jentzsch and Richard Cyganiak. http://lod-cloud.net/.

The data from the LOD Cloud can be searched and retrieved by using the technologies of the Semantic Web, i.e. by using the SPARQL query language and the concept of SPARQL query federation [78]. This enables access to data from distributed data sources on the Web in a way similar to a local database access. The LOD Cloud enables new application development options for a specific domain, or a combination of domains. Unlike the Web 2.0 mesh applications which function using a predetermined set of data sources, applications using Linked Data can operate over an (almost) unlimited, global data space. With each new dataset added to the LOD Cloud, the applications using the LOD data can automatically make use of this newly available data.

The Linked Data Principles provide general directions for making a given dataset available on the Web in Linked Data format. But, there are several different tools, methods and techniques for generating and publishing such datasets, and their application depends on the type of the source data, its nature, and a list of other factors. These methods and techniques are united in several methodologies for the Linked Data life-cycle, which offer different approaches for handling Linked Data in specific domains and for specific purposes. Some of these methodologies are aimed towards Government Linked Data, such as those from [55] and [90]. The methodology described in [41] is specific for the set of tools developed as part of the LOD2 project [19]. One general methodology is [52]. In [79] and [91], the authors present methodologies aimed towards Linked Data in the TV and library domains. In [80] and [65] the authors represent a methodology for providing a higher data quality in Linked Data datasets.

In this PhD thesis, based on our experience with generating Linked Data in various domains and based on the existing Linked Data methodologies, we define a new Linked Data methodology with a focus on reuse. It consists of five steps which encompass the tasks of studying the domain, modeling the data, transforming the data, publishing it and exploiting it. In each of the steps, the methodology provides guidance to data publishers on defining reusable components in the form of tools, schemas and services, for the given domain. With this, future Linked Data publishers in the domain would be able to reuse these components to go through the life-cycle steps in a more efficient and productive manner. With the reuse of schemas from the domain, the resulting Linked Data dataset will be compatible and aligned with other datasets generated by reusing the same components, which additionally leverages the value of the datasets.

This approach aims to encourage data publishers to generate high-quality, aligned Linked Data datasets from various domains, leading to further growth of the number of datasets on the LOD Cloud, their quality and the exploitation scenarios. With the emergence of data-driven scientific fields, such as Data Science, creating and publishing high-quality Linked Data datasets on the Web is becoming even more important, as it provides an open dataspace built on existing Web standards. Such a dataspace enables data scientists to make data analytics over the cleaned, structured and aligned data in it, in order to produce new knowledge and introduce new value in a given domain. As the Linked Data principles are also applicable within closed environments over proprietary data, the same methods and approaches are applicable in the enterprise domain as well.

Problem Description and Objectives

The existing Linked Data methodologies are generally focusing on the necessary steps for locating, generating / transforming and publishing Linked Data, but come short in covering the reuse part of the dataset life-cycle, i.e. the part involving use of the created schemas, tools, services, etc., by future data publishers from the same domain. None of the existing methodologies offers guidelines, mechanisms or methods which would motivate and enable the data publishers to share the mechanisms developed during the generation and publishing of the Linked Data dataset from a given domain. Because of this, the knowledge of the process for creating the datasets remains separated from the Linked Data community and usually resides in scientific papers or reports. With this, the Linked Data life-cycle needs to be implemented from scratch for a given domain, each time new data publishers decide to generate Linked Data datasets.

In domains in which multiple data publishers might be interested in generating Linked Data datasets covering the same or a similar set of entities, a methodology providing guidance in creating and applying reusable life-cycle steps as components would enable initial data publishers to share their knowledge and expertise in the domain, whilst providing benefit for all future data publishers from the same domain. They would be able to reuse not just the knowledge, but the actual tools, schemas and services developed by the initial data publishers. The reuse of schemas inevitably results in compatible and aligned datasets, which can easily be interconnected. The benefit of such a methodology is the lowering of the barrier for data publishers to generate and publish Linked Data datasets both in scenarios in which the data publishers are not proficient in Linked Data or they are not familiar with the domain in question. This leads to the generation and publishing of more Linked Data from the domain in question, linked to previously developed datasets and to the LOD Cloud.

In this PhD thesis, based on existing Linked Data methodologies and our extensive experience in applying the Linked Data principles in various domain, we propose and develop a new Linked Data methodology, which focuses on the principle of reuse of the life-cycle steps, for a given domain. It includes steps aimed towards modeling and aligning the data, transforming the data into 5-star Linked Data, publishing the created datasets on the Web and defining use-cases or developing applications on top of the dataset. The guidelines from the methodology are aimed towards assisting data owners and publishers from a given domain in publishing their data in the same aligned, Linked Data format. Their data, once transformed into Linked Drug Data and interlinked with other data already published using the same reusable components, could be used in new user-centric and analytical application and services.

As a validation of the proposed methodological guidelines, we apply them in the drug and healthcare domain. We apply the methodology within an automated system which gathers drug data from the official national drug registries of twenty-three different countries, executes data cleaning, aligns and transforms the data into 5-star Linked Data and publishes them on the Web in a common, aligned and consolidated Linked Drug Data dataset. Based on the guidelines, we develop reusable components from the life-cycle: a common schema, a data template, a transformer, a SPARQL-based tool for extending and interlinking the dataset and a web-based tool for transforming, interlinking and publishing the data. We then demonstrate a set of user-centric and

analytical use-case scenarios over the generated dataset, which are otherwise unavailable or very time-consuming in a scenario where a user works with the data available on the Web in HTML webpages.

The approach of our methodology aims to encourage data publishers to generate high-quality, aligned Linked Data datasets from various domains, leading to further growth of the number of datasets on the LOD Cloud, their quality and the exploitation scenarios. That would be in line with continuing the process of replacing the current web of documents with the envisaged web of data, which represents an open dataspace accessible via the existing infrastructure of the Web. Such an open and interlinked dataspace can represent a solid ground for data-driven science. With the emergence of data-driven scientific fields, such as Data Science, creating and publishing high-quality Linked Data datasets on the Web is becoming even more important. It enables data scientists to make data analytics over the cleaned, structured and aligned data in order to produce new knowledge and introduce new value in a given domain. As the Linked Data principles are also applicable within closed environments over proprietary data, the same methods and approaches are applicable in the enterprise domain as well.

Content of the Thesis

The thesis consist of an introduction chapter, followed by a chapter for Linked Data, a chapter for an overview of existing Linked Data methodologies, a chapter for our experience with applying the Linked Data principles in various domains, a chapter for our modular Linked Data methodology focused on reuse, and a conclusion chapter.

Chapter 1

The first chapter, 'Introduction', provides the motivation, problem description and the objectives of our research and the PhD thesis. Part of its content were elaborated in Chapter 'Background and Motivation' and Chapter 'Problem Description and Objectives'.

Chapter 2

The second chapter, 'Linked Data', presents the four principles of Linked Data, the Linked Open Data (LOD) Cloud, the 5-Star Open Data metric system, as well as the connection between Linked Data and the Semantic Web and its technologies. Here we also elaborate on the existing techniques and best practices for generating and publishing Linked Data, both on the Web and within private repositories, such as those in enterprise systems.

Chapter 3

The third chapter, 'Linked Data Methodologies', focuses on an in-depth analysis of the existing Linked Data methodologies aimed at formalizing the life-cycle of Linked Data in the government domain or in general. The W3C Government Linked Data Working Group has created official guidelines for publishing and accessing open (government) data using the Linked Data principles [3], and with it they suggest three existing methodologies which can be used with the Linked (Government) Data life-cycle: the methodology of Hyland et al., the methodology of Hausenblas et al. and the methodology of Villazón-Terrazas et al. In addition to these three methodologies, we include the updated Linked Data life-cycle developed by the LOD2 Project, aimed at extracting, creating, enriching, linking and maintaining Linked Data.

Hyland et al. [55] define a methodology for Linked Government Data, which consists of six steps. Their methodology is based on the specifications and best practices by the W3C, and consists of the following steps: (1) Identify, (2) Model, (3) Name, (4) Describe, (5) Convert, (6) Publish, and (7) Maintain. The methodology contains most steps which are part of the generally accepted Linked Data life-cycle, but is missing guidelines on how to use the generated Linked Data. The authors believe that the usage of the generated dataset should be left to the users and other interested parties, and according to them, is not a task for the Linked Data publisher.

Hausenblas et al. [52] state that the existing data management approaches - which assume control over the data, the schema and the data generation - cannot be used in the environment of the Web, due to its open and decentralized nature. Their methodology consist of the following steps: (1) Data awareness, (2) Modeling, (3) Publishing, (4) Discovery, (5) Integration, and (6) Use-cases. It also covers most of the general Linked Data life-cycle steps, but does not provide detailed guidelines for the process of publishing the generated Linked Data dataset on the Web.

Similarly to Hyland et al., based on their experience in linked government data production, Villazón-Terrazas et al. [90] define a set of methodological guidelines for generating, publishing and exploiting Linked Government Data. Their life-cycle consists of the following steps: (1) Specify, (2) Model, (3) Generate, (4) Publish, and (5) Exploit. This is the only existing methodology in the Linked Data domain which covers all of the life-cycle steps, but unfortunately is focused on government data.

The Linked Data life-cycle supported by the LOD2 integrated environment [41] consists of (1) Extraction, (2) Storage, (3) Authoring, (4) Interlinking, (5) Classification, (6) Quality, (7) Evolution/Repair, and (8) Search/Browsing/Exploration. Even though this is the only methodology which provides software tools for the denoted steps, and the number of steps here is larger than in the other methodologies, it still misses some key elements of the Linked Data life-cycle, such as the data modeling, the definition of the URI format for the entities and the ways of publishing the generated dataset. The provided tools are also general, and cannot be applied in a specific domain without further work and domain knowledge.

Chapter 4

The fourth chapter, 'Transformation and Usage of Linked Data in Various Domains', consists of our work with Linked Data in the domains of crime data, public transport and air pollution, the financial domain, the multimedia domain and the healthcare and drug domain. In it we describe our work on eleven different Linked Data research and application projects. As part of these projects, we have designed, developed and published seven ontologies, transformed, consolidated, linked and published twelve Linked Data datasets, developed numerous sets of user-centric and analytic use-cases over the datasets and developed and published six web and mobile applications which solely use these datasets as a data layer.

Crime Data. In the crime data domain, we developed the first crime map for the Republic of Macedonia, based on data published by the Macedonian Ministry of Interior Affairs. By using web crawling, data gathering, data cleaning, natural language processing and geo-location, we generate an Open Data dataset for all published crimes in Macedonia [86]. We categorize the crimes in several different categories and use them in a web application[1], which aims to present the public with a general overview of crime patterns on the territory of the Republic of Macedonia.

Public Transport and Air Pollution. In the public transport and air pollution domain, we worked on four different research projects: we worked with Macedonian public transport data, with Swedish public transport data, with CO_2 emissions data from vehicles in the European Union, and with air pollution data from Skopje, Macedonia.

In the first research project, we generated and published 4-star Open Data from the public transport company JSP Skopje [73]. After obtaining the data, we first transformed it into the standardized General Transit Feed Specification (GTFS) format (3-star data), and then, using the Transit Ontology, the W3C Geospatial Vocabulary, and our own GTFS-ext Ontology[2], we transformed the data into RDF format (4-star data). Over the dataset, we generated a set of use-case scenarios which can be executed as SPARQL queries over the public SPARQL endpoint we provide.

In the second research project, we designed and developed an automated system which transforms and publishes the public transport from Sweden to 5-star Linked Data [77]. In order to realize the annotation, we defined a new ontology, the Transport Administration (TAO) Ontology[3]. In order to transform the dataset into a Linked Data dataset, the automated system links the city entities from our dataset to city entities from DBpedia. Then we use the generated and published Linked Data dataset in a web application[4] which exploits the 5-star nature of the dataset and its links to entities from DBpedia and the LOD Cloud.

[1] `http://crimemap.finki.ukim.mk/`
[2] `http://linkeddata.finki.ukim.mk/lod/ontology/gtfs-ext#`
[3] `http://linkeddata.finki.ukim.mk/lod/ontology/tao#`
[4] `http://sta.linkeddata.finki.ukim.mk/`

In the third research project, we worked with data about CO_2 emissions from vehicles in the European Union [75]. We generated and published the data as Linked Data, using our Vehicle Emissions Ontology (VEO)[5], developed as part of the research. The Linked Data dataset was linked to automobile entities from DBpedia, and these links were further exploited in several use-cases we defined over the published dataset.

The fourth research project in this domain was about air quality data from the region in and around Macedonia's capital, Skopje [72]. Using air quality measurement stations, we created a system for gathering the raw data, extending it using interpolation, annotating it as RDF, interlinking it to location entities from DBpedia and publishing it as Linked Data on the Web. We defined a set of use-cases over the generated and published dataset, and demonstrated how the data can be exploited by applications and services from third-parties.

Financial Data. In the financial domain, we worked on a research project in which we gathered, transformed and consolidated financial data from the Macedonian Stock Exchange (MSE), websites of large Macedonian companies and the World Bank [76]. We transformed the source data into 4-star and 5-star Linked Data using the Registered Organization Vocabulary, the Asset Description Metadata Schema, and our own Corporate Financial Reports and Loans Ontology (CFRL)[6], which we used for interlinking the different source data, i.e. to interlink the company entities with their loans and financial reports. We then developed a set of use-case scenarios which demonstrate the advantages of having a consolidated financial dataset as a Linked Data dataset, publicly available over a SPARQL endpoint.

Multimedia. In the multimedia domain, we worked on a research project for generating, consolidating and publishing Linked Music Data from global music charts [61]. We designed an automated system to crawl and gather playlist and chart data from various global music stations, align them, annotate them using our own Playlist Ontology[7] and interlink them with various entities from the LOD Cloud. The Linked Music Data dataset is then used in a web application which provides general user-centric scenarios, but also provides analytics over the time-dynamics and global distribution of music taste, based on genres, artists, etc.

Healthcare. In the healthcare and drug domain, we worked on four different research projects: we worked with drug data from the Macedonian Health Insurance Fund, with data about medical institutions registered in Macedonia, then drug data from the Macedonian Drug Bureau, and we made a global analysis of the negative effects between food and drugs on the level of cuisines and drug categories.

In the first research project, we generated Linked Drug Data from the data published by the Macedonian Health Insurance Fund, regarding registered drug products in Macedonia [60]. We gathered the 2-star data and created a consolidated and interlinked 5-star Linked Data dataset, using our own HIFM Ontology[8]. We interlinked the drug products from the dataset based on their target and function, and we also linked them to their corresponding generic drugs from the DrugBank dataset, in the LOD Cloud. We then presented a set of use-cases which exploit the linked nature of the generated and published dataset, especially the links between the drugs from the dataset and their links to generic drugs from the LOD Cloud.

In the second research project, we extended the previous dataset with drug products from the Fund with data about the medical institutions in Macedonia, their category, address information, work hours and schedule, and with privately obtained drug availability lists for a small group of pharmacies [59]. For the annotation process we reused the HIFM Ontology, and extended it with a few missing properties. We then developed a set of use-case scenarios which exploit the links between the three separate datasets: the drug product dataset, the drug availability lists and the medical institutions dataset. We were able to demonstrate scenarios in which, for instance, a user is able to locate the nearest pharmacy which is currently open and which has the drug of interest, using the public Linked Data dataset we provided.

In the third research project, we improved on the first project by designing a completely

[5] `http://linkeddata.finki.ukim.mk/lod/ontology/veo#`
[6] `http://linkeddata.finki.ukim.mk/lod/ontology/cfrl#`
[7] `http://linkeddata.finki.ukim.mk/lod/ontology/po#`
[8] `http://linkeddata.finki.ukim.mk/lod/ontology/hifm#`

automated and sustainable system for gathering, cleaning, transforming and publishing Linked Drug Data for drug products from Macedonia [58]. This time, we used the newly published drug registry by the Macedonian Drug Bureau, which provided extensive data about all registered drugs. For annotation purposes we had to design a new ontology, the DBM Ontology[9]. Similarly as in the first project in this domain, we interlinked the drug products based on their function, and we linked them to the corresponding generic drugs from the DrugBank dataset. We then developed a mobile application, "Mobile Pharmacist", which solely uses the Linked Drug Data dataset as a data-layer, and based on the links the dataset has to entities from the LOD Cloud, it provides the users with in-depth information about the drugs registered in Macedonia.

In the fourth research project, we made a global analysis on the negative interactions between national cuisines and drug categories [57]. We analyzed known food-drug interactions from the perspective of national cuisines and their recipes and ingredients, and from the perspective of drug categories. Additionally, we analyzed the effect of ingredients in the negative cuisine-drug interactions. The analysis was based on two Linked Data datasets we generated from different sources: the drug dataset[10] and the recipes dataset[11]. The results showed two significant patterns: the drugs from ATC categories B, C, N and V have negative interactions with recipes from South Europe, Asia, Latin America and Africa, while drugs from the ATC categories A, D, G, J, L and S have negative interactions with recipes from North America and Europe. These patterns are mostly due to the use of garlic and ginger in the recipes from the first pattern, and the use of milk in the second. The impact of milk and garlic varies across the world, mainly due to cultural, historical and biological reasons for their presence or lack thereof in a given cuisine. The maps showing the varying impact of cuisines and ingredients on drug categories, are publicly available[12].

Chapter 5

The fifth chapter, 'Linked Data Methodology Focused on Reuse', contains the main results from the PhD thesis. It contains the details regarding our new proposed and developed methodology for Linked Data, which was developed based on the existing methodologies presented in Chapter 3, and our extensive experience in the domain of Linked Data, presented in Chapter 4. A full description of the methodology is presented in the next chapter, 'Main Results'.

Chapter 6

The sixth chapter, 'Conclusion', concludes the PhD thesis with an overview of the problem, the motivation, the main objective and the results from the research. A shorter version of the conclusion is presented in this executive summary, in the last chapter, 'Conclusion'.

[9] http://linkeddata.finki.ukim.mk/lod/ontology/dbm#
[10] http://datahub.io/dataset/drug-dataset
[11] http://datahub.io/dataset/recipe-dataset
[12] http://viz.linkeddata.finki.ukim.mk/

Main Results

Linked Data Methodology Focused on Reuse

Based on our experience with applying the Linked Data principles in the domains of public transport and air pollution [73, 77, 75, 72], the financial domain [76], the entertainment domain [61] and the healthcare domain [60, 59, 58, 57], we developed a methodology for Linked Data, focused on reusable components as support for the methodology steps. These guidelines build on the existing Linked Data methodologies and contain actions which cover the general Linked Data lifecycle. Their purpose is to guide data publishers through the process of generating high quality, 5-star Linked Data in order to interlink, align and consolidate the data from a given domain. The relationship between the existing methodologies and our guidelines is outlined in Table 1.

Table 1: Aligning our methodological guidelines with existing Linked Data methodologies.

	Our Methodology	Hyland et al.	Hausenblas et al.	Villazón-Terrazas et al.	LOD2
Reuse by Step Modularity	**I. Domain and Data Knowledge**	1. Identify	1. Data Awareness	1. Specify	1. Extraction 2. Storage
	II. Data Modeling and Alignment	2. Model 3. Name	2. Modeling	2. Model	
	III. Transformation into 5-star Linked Data	4. Describe 5. Convert	3. Publishing 4. Discovery 5. Integration	3. Generate	3. Authoring 4. Interlinking 5. Classification 6. Quality 7. Evolution/ Repair
	IV. Publishing the dataset on the Web	6. Publish		4. Publish	
	V. Use-cases, Apps and Services		6. Use cases	5. Exploit	8. Search/ Browsing/ Exploration

Our methodological guidelines consist of the following steps (Fig. 2):

I. Domain and Data Knowledge

II. Data Modeling and Alignment

III. Transformation into 5-star Linked Data

IV. Publishing the Linked Data dataset on the Web

V. Use-cases, Applications and Services

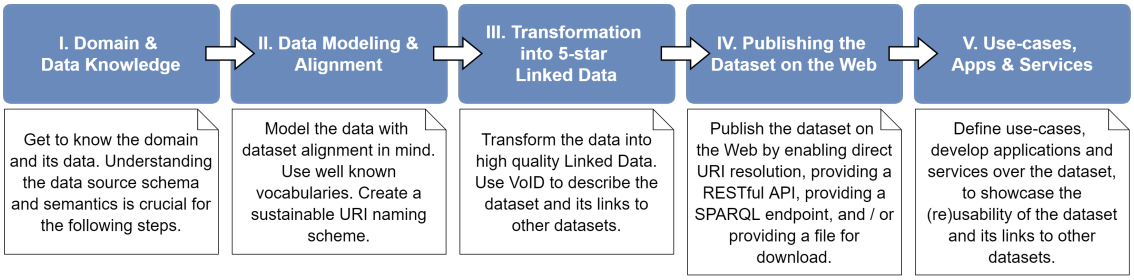

Figure 2: The methodology for consolidating drug data using the Linked Data approach.

Step I: Domain and Data Knowledge

The first step corresponds to the first steps from the existing methodologies: it is important for the data publisher to know the domain and the data in it very well. This understanding of the data source schema and semantics is crucial for the following steps which will involve data modeling, schema alignment and data transformation.

If this is the first time the data publisher comes across Linked Data, our advice is to first get familiar with the 5-star data system from Tim Berners-Lee [42], the four principles of Linked Data [43], and the LOD Cloud [15]. After that, it is important for the data publisher to get familiar with the domain in question and with the meaning of the dataset selected for transformation. For this, a consult with a domain expert is usually necessary and therefore advised. Another approach is to explore the existing Linked Data datasets which are similar to or from the same domain as the one of interest. For this, the Datahub portal [5] and the LOD Cloud cache instance [16] could be used. These activities will help the data publisher to get a better insight into the types of data which currently exist as Linked Data, their schema, their similarities and differences and their existing and potential links. It will also help him / her determine the ontologies and vocabularies already used in the domain, which can be important for the next step.

Step II: Data Modeling and Alignment

In the next step, the data publisher should focus on data modeling and alignment with other existing or future datasets. The data publisher has to choose the correct schema for the dataset, in order to annotate it correctly, i.e. use the data fields which are necessary for the final use-cases, annotate the fields unambiguously and with the correct semantics and make the correct schema choices which will allow seamless alignment with other datasets. Additionally, the data publisher has to define the URI naming scheme for the data entities, and optionally for the ontology or vocabulary classes and properties.

Data Schema. The data schema is defined with the choice of vocabularies or ontologies to be used. The principles of ontology engineering and usage have been developed for this purpose exactly: to maximize the chances of reuse, and therefore allow better alignment between datasets [40]. This means the data publisher should always try to reuse an existing vocabulary or ontology, giving advantage to those which are most widely used. A few tools for ontology and vocabulary discovery exist, and the data publisher should use them in this stage. The two most notable are Linked Open Vocabularies (LOV) [18] and DERI Vocabularies [6], which also provide usage statistics which can be used to assess the impact of a given vocabulary or ontology in a specific domain.

However, datasets tend to have specific fields, which are not covered by existing ontologies. In this cases, the existing ontology or vocabulary should be extended, or a new one should be defined. However, each time a new ontology is developed, it is important to define the mappings between the new classes and properties and the classes and properties from other ontologies, in order to enable ontology matching and RDF-based reasoning, for schema alignment.

Another important approach in this step is the use of upper-level ontologies and vocabularies; they can provide a schema for many different and specific domains, due to their generality. Having two or more datasets annotated with the same upper-level ontology or vocabulary allows interlinking and inference between them, i.e. it improves the alignment which is crucial for data consolidation.

URI Formats. From the URI naming scheme perspective, it is important to determine the types of entities which exist in the dataset. This will help in defining the entity URIs for the Linked Data dataset. According to the Linked Data principles, each entity in the dataset - along with the classes and properties in the ontology - needs to have a unique identifier in the form of an HTTP URI. In order to provide better performance when using the dataset in the future, our experience suggests using separate URL paths for different entity types, e.g. `http://example.com/drug/`, `http://example.com/interaction/`, `http://example.com/disease/`, etc. An additional recommendation is to use slash-based URIs, instead of hash-based ones. This may result in using an additional HTTP request by the machine accessing the URI, but it provides better performance when accessing large datasets [36].

Step III: Transformation into 5-star Linked Data

During the third step, the source dataset should be transformed into a 5-star Linked Data dataset. After the transformation, additional metadata for the generated dataset needs to be added.

Data Transformation. The process of transformation can be executed in many different ways, and with various software tools, e.g. OpenRefine [22], LODRefine [20], D2R Server [4], Virtuoso [35], Silk Framework [27], etc. In order to make the correct choices about the tools to be used for the transformation process, it is important to distinguish the characteristics of the dataset first. The nature of the dataset will determine if (a) the transformation is a one-time task, a task which will have to be executed on a given time interval (e.g. once a month), or a continually running task; (b) old versions of the transformed dataset are necessary for versioning and as backup, if during future transformations only the changes in the data are needed for transformation, i.e. 'delta' updates are performed, or if older data are no longer necessary for the particular use-case; (c) manual or automated data cleansing is needed before the first transformation and / or subsequent transformations; (d) the source dataset is always available at the same location and is accessible via the same interfaces. These specifics of the dataset in question can then help the data publisher determine if the transformation task can be fully or partially automated, and identify the parts of the transformation workflow which require human attention and input.

Metadata. Adding metadata about the newly created Linked Data dataset is significant from the data reuse perspective - using vocabularies such as VoID [38] help ubiquitously determine the characteristics of the dataset and the links the dataset has to other Linked Data datasets, through software agents. VoID metadata contains information about the name, description and category of the dataset, versioning information and update frequency, contact information, the license under which the dataset is made available, the links to the SPARQL endpoints and URI lookup endpoints, used vocabularies and their properties and classes. It also explicitly defines the links between the dataset and other Linked Data sets, defined in the dataset itself. The use of the VoID vocabulary is explicitly stated in the corresponding steps in the methodologies of Hyland et al., Hausenblas et al. and Villazón-Terrazas et al.

Step IV: Publishing the Linked Data dataset on the Web

In the forth step, the generated 5-star Linked Data dataset, along with its VoID metadata, should be published on the Web. This should be done following the W3C recommendations for publishing Linked Data on the Web [3], which suggest enabling direct URI resolution, providing a RESTful API, providing a SPARQL endpoint, and / or providing the dataset as a file for download.

There is a large palette of tools and software platforms which allow seamless Linked Data publishing. Among them are D2R Server [4] and Virtuoso [35], which allow Linked Data publishing of datasets which are originally in an RDF file (Turtle, N3, RDF/XML, JSON-LD, etc.), a CSV

file, or in a relational data base. These platforms then allow access to the Linked Data dataset via HTML pages, via RDF file downloads and via a SPARQL endpoint which can be used as a RESTful API as well.

Another important part of this step is the announcement of the newly created Linked Data dataset to the public. For this, information about the dataset along with its VoID metadata should be published on popular data portals, such as Datahub.io [5]. The data publisher is also advised to join the LOD Cloud [17]. Both these actions will enable higher visibility of the dataset. The announcement for the newly published Linked Data dataset should also be done via existing communication channels of the data publisher and his / her organization. In order to facilitate further use and reuse of the dataset, it is important to provide a form or a contact email address for interested parties to be able to report data or access issues, and provide feedback. On the organization side, it is important that these reports and requests are attended to in a timely fashion; otherwise the usability of the dataset is significantly lowered.

Step V: Use-cases, Applications and Services

The last step refers to defining use-case scenarios and / or developing specific applications and services which will use the data from the newly created Linked Data dataset, to showcase the (re)usability of the dataset and its links to other Linked Data datasets. This will present the potential of the contextually linked datasets to future interested parties.

The use-cases can be text-based scenarios, specific SPARQL queries, or prototype applications, describing the ways in which the data from the new dataset can be browsed, retrieved and used. Here, a specific focus should be given on how the links to other Linked Data datasets can be exploited to reach other data, not present in the original data source, to extend its context. With this, the data publisher will show to interested parties that the original dataset has more value when combined with datasets from the same or similar context, instead of being used in an isolated scenario. Besides such use-case, the same effects of the Linked Data dataset can be showcased with the development of applications and services. They bring more visibility to the general (re)usability of the Linked Data dataset, but generally require more time and effort.

The created use-cases, applications and / or services, should be shared and announced to the public, along with the dataset itself and its VoID metadata. The use of the same channels from the previous step is advised.

Modularity

In order to facilitate future Linked Data creation in the domain, by both the same and other data publishers, we recommend development of reusable components during the steps of the methodology. Data publishers should make their Linked Data life-cycle modular, i.e. constructed of loosely-coupled components which can be reused in the same domain. Here, by loosely-coupled we mean components which can be used separately when necessary, but which also form a seamless workflow for generating a high-quality, 5-star Linked Data dataset. The reuse of such components, like in other software development cases, reduces development time and increases productivity [67, 70].

Step I, which focuses on studying the domain in question, cannot be encapsulated within a software tool or component. However, the knowledge gathered in this step helps shape the tasks in the following steps, which means that it will translate into their tools and components. If that is the case, then future data publishers from the same domain can go through this step more quickly, or completely skip it.

The tasks in Step II, where the data publisher should develop the data schema and consider alignment of the dataset with existing and future datasets from the domain, can be developed as a separate, reusable component. The component can take the form of a data schema, defined in RDFS or OWL, which represents the vocabulary or ontology to be used for annotating the source dataset, in order to transform it in a 4-star RDF dataset. The vocabulary / ontology should consist of existing or newly created classes and properties, which provide good alignment with

existing datasets from the domain, but also consider future alignment with data which will be published by others in the same domain.

The transformation of the source dataset into a Linked Data dataset in Step III is a process which can be developed as reusable. All software tools which can be used in this step require a structured dataset as input, use a mapping from the source data schema to an RDF schema, and output RDF. This means that if a data publisher defines a reusable RDF schema in Step II, he / she can define (a) a template for the source dataset in a given format, (b) a mapping of the source dataset based on the input template and the data schema from Step II, (c) a process, software component, tool or hosted service which will use the source dataset and the mapping and generate the desired Linked Data dataset in RDF format. For instance, the template can be an XML, CSV, TSV, or a file in another format, the mapping can be defined in the language supported by the transformation tool, such as R2RML [49], OpenRefine RDF Extension-based mapping [20], etc. The actual transformation process can be encapsulated in an automated script which gets the source dataset which conforms to the template, sends it to the transformation tool and gets the outputted RDF. This can be achieved in various ways, depending on the tools being used. In case of a D2R Server, the data should be loaded into a relational database, and the server should be started with the appropriate mapping file. In the case of Virtuoso, the data can be loaded in a relational database or as a CSV, and then the transformation based on the mapping file should be triggered. In case an OpenRefine / LODRefine instance is used, the process needs to be manual, as the data needs to be loaded and the transformation action list needs to be applied manually. However, an alternative BatchRefine instance can be used [7], which provides an HTTP REST-based interface over OpenRefine / LODRefine features. This means an HTTP call to a BatchRefine instance can trigger a transformation of the source dataset (sent in the request) based on the OpenRefine RDF Extension-based mapping (referred to in the request).

In Step IV, the data publisher needs to publish the generated Linked Data dataset on the Web according to the W3C recommendations. If a publicly available Virtuoso or D2R Server instance is used in Step III, the dataset will already be publicly available. The SPARQL endpoints of both platforms allow access to the data from the dataset for querying and as a RESTful API. For URI resolution, it is important that the URI naming scheme from Step III corresponds to the domain used for generating and publishing the data. In the case of D2R Server, the entity URIs are dereferenceable by default. In Virtuoso, however, certain URL rewrite rules need to be applied: the script to do this can also be developed as a parameterized and reusable component. In order to aid future data publishers in the domain, a data publisher can consider developing a service for data publishing: the service would get the generated Linked Data dataset, store it in the platform and expose it on the Web via dereferenceable URIs and a SPARQL endpoint.

The use-cases, applications and services from Step V are dependent on the data publisher and his / her idea of exploiting the linked nature of the generated dataset. However, depending on the domain, it is possible to have an application or a service which uses data from a given RDF repository in which the newly generated Linked Data dataset from the domain can be added as well. If a data publisher reuses the components made available from previous data publishers in the domain, the resulting dataset will consist of entities from the same type, annotated with the same vocabulary or ontology, and therefore completely aligned with the existing dataset(s). In such a case, a service which allows new data publishers to add their Linked Data datasets from the domain, developed using the same reusable components, into an existing RDF repository, would mean that applications and services built on top of it will use the new data as well.

Formalization

We can define the practical part of our Linked Data life-cycle for a given domain (d) as a tuple of ordered, loosely-coupled components:

$$T_d = (C_{2d}, C_{3d}, C_{4d}, C_{5d}) \tag{1}$$

where C_{2d}, C_{3d}, C_{4d} and C_{5d} are the reusable components from Step II, Step III, Step IV and Step V, in domain d, respectively. The components can be composite, i.e. they can consist of

smaller components and tools which serve a specific purpose in the step in question, for instance $C_{3d} = (C_{3_a d}, C_{3_b d})$. As we already stated, Step I does not have a practical component; it's focused on studying the domain for the purpose of implementing the domain specifics into the following steps.

When a data publisher is working with creating a Linked Data dataset in domain d, and develops a set of reusable components: $C_{2d}, C_{3d}, C_{4d}, C_{5d}$, which can be combined into a tuple $T_d = (C_{2d}, C_{3d}, C_{4d}, C_{5d})$, future data publishers from the same domain d should be able to reuse the entire tuple, i.e. the entire set of components in the given order, in order to transform their source datasets into Linked Data datasets. In this case, the datasets generated using the same tuple (T_d) would be completely aligned, as they use the same schema (C_{2d}), the same source data template and transformation process (C_{3d}). This would allow seamless consolidation of the datasets, without the need for additional ontology mappings and data alignment, which is still a great challenge in the data management domain [81] and therefore an important gain.

In case a data publisher has a source dataset which is slightly different from the one for which a reusable tuple $T_d = (C_{d2}, C_{3d}, C_{4d}, C_{5d})$ exists, he / she can modify the data schema C_{2d} and get C'_{2d}. The modified data schema would require a change in the transformation process, as well, meaning that the data publisher will define and use a new reusable component, C'_{3d}. Depending on the implementation of C_{4d} and C_{5d}, they may be modified or used directly as provided. In the case they remain the same, the new tuple for the data publisher would take the following form:

$$T'_d = (C'_{2d}, C'_{3d}, C_{4d}, C_{5d}) \tag{2}$$

where C'_{2d} and C'_{3d} are the new, modified components for Step II and Step III. These components can then also be published, and made available for future data publishers from the domain d, along with the original C_{2d} and C_{3d} components. In this case, the generated Linked Data dataset would not be completely aligned with the datasets generated using the T_d tuple, but the datasets will be significantly closer schema-wise than if a data schema is developed from scratch.

In general, a data publisher from domain d does not need to define reusable components for all of the steps, as is the case in equation 1. There are several other valid combinations, such as $T_d = (C_{2d})$, where only the data schema is made available for reuse; $T_d = (C_{2d}, C_{3d})$, where the data schema, the source data template and the transformer process are made reusable; $T_d = (C_{2d}, C_{3d}, C_{4d})$, where a dataset publishing service is also made available; $T_d = (C_{4d})$, where only such service is made available, without the schema and the transformer; $T_d = (C_{4d}, C_{5d})$, where a service for adding the dataset in the data layer of existing applications and services is made available; $T_d = (C_{5d})$ where such a service is the sole reusable component defined. In all cases, the reusable components from the tuple will enable more efficient execution of the tuple from future data publishers in the same domain.

Potential Drawbacks

As the authors of [69] point out, this type of pre-planned reuse approach has several potential drawbacks: (a) development of reusable components can be more expensive than otherwise [50], (b) deciding which component to make reusable is not an easy task [87], and (c) reusable components are often developed with certain assumptions in place, which limit the scope of future reuse [46]. However, there are several differences between making Linked Data life-cycle components reusable and doing so for general software components.

The first drawback, the additional cost, does not apply for the tools and components from Step II and Step III. The data schema and the transformation process need to be developed even when reusability is not the focus. Making the data schema (C_{2d}), the structured data template and the mapper files (C_{3d}) available to future interested data publishers from the domain does not introduce additional cost to the process. Providing a live BatchRefine, Virtuoso or D2R Server instance for the transformation process, for example, can add a certain cost, but as these tools are open source, publishing the schema and the mappers would provide sufficient reusability of the process for future data publishers. The components from Step IV and Step V would require

additional resources from the initial data publisher in the domain, as setting up a service for publishing third-party datasets is not part of the default life-cycle. However, the incentive for a data publisher to provide such services would be the future availability of the data from new datasets from the domain - generated using the same reusable components - within applications and services built on top of it. Enabling future data publishers to seamlessly provide and publish Linked Data datasets from the domain in a hosted platform, would provide additional data in the RDF repositories being used by the applications and services running on top of it. If such applications and services belong to the original data publisher, he / she will have a larger dataset as a data layer. If the applications and services are third-party, various economic models can be employed to make such a service a viable option.

The second drawback, the difficulty of deciding which components to make reusable, is mitigated by our definitions on the possible structure of the T_d tuple. The data publisher from the domain can choose which form of a reusable tuple he / she can provide to future data publishers.

The third drawback, the limitations of reuse due to assumptions applied during implementation, is avoided due to the scope of the reusable components and tuple: the domain. In equation 1, the components and the tuple refer to the domain d in which they can be used. Therefore, a component intended for reuse, such as C_{3d}, has the domain d as a scope, and cannot be incorrectly assumed to be applicable in another domain. The same logic applies on the level of sub-components, $C_{3d} = (C_{3_a d}, C_{3_b d})$, and the tuple level, $T_d = (C_{2d}, C_{3d})$, where the sub-components and the tuple have scope of the domain d.

Validation

In order to validate the methodology and the proposed guidelines, we chose the domain of drug products. The source datasets are national drug registries which contain the drug products registered to be sold in the country. To facilitate the generation of Linked Drug Data datasets for all possible source datasets, we developed a reusable tuple for the drug product domain:

$$T_{drugs} = (C_{2drugs}, C_{3drugs}, C_{4drugs}) \tag{3}$$

where $C_{3drugs} = (C_{3_a drugs}, C_{3_b drugs}, C_{3_c drugs})$ is a composite component for Step III. With this, the tuple gets the form:

$$T_{drugs} = (C_{2drugs}, (C_{3_a drugs}, C_{3_b drugs}, C_{3_c drugs}), C_{4drugs}) \tag{4}$$

Here we will make an overview of the reusable components for the drug product domain, and then we will present a validation and a proof-of-concept project, in which we apply the proposed methodological guidelines and reusable components. The system gathers drug data from the official national drug registries of twenty-three different countries, executes data cleaning, aligns and transforms the data into 5-star Linked Data and publishes them on the Web in a common, aligned and consolidated Linked Drug Data dataset.

Reusable Components for the Drug Data Domain

As part of the methodological guidelines, with the intent to provide help to the data publishers working in the drug product data domain, we designed and developed a set of tools as reusable components. The set consists of the RDF schema (C_{2drugs}), the CSV template ($C_{3_a drugs}$), the OpenRefine transformation script ($C_{3_b drugs}$), the SPARQL-based tool for extending and interlinking the dataset ($C_{3_c drugs}$) and the web-based tool for automated transformation, interlinking and publishing (C_{4drugs}) of the generated Linked Drug Data dataset.

RDF Schema (C_{2drugs})

In order to model the domain of drug products on a global scale, according to Step II of the methodology, we needed to create one common and reusable schema for all national drug data

repositories, and then use it for annotating the drug data. With it, the goal was to provide alignment of drug data from different sources, with different format and different levels of data granularity, in order to enable simpler data exploitation.

First, we analyzed the national drug data repositories of 31 countries[13] and the analysis helped us define a common set of properties which exist and which we want to use in our Linked Drug Data dataset. This set consisted of 24 properties, including the brand name of the drug, the generic name, the ATC code, the EAN code (barcode), the list of active substances, the drug strength, dosage form, cost, manufacturer, the country it was registered in, the details about its license, etc. Not all national drug data registries provide all of the data and properties we selected for our schema, but we did not want to decide against using those properties - they are useful where available.

Following the best practices in ontology and vocabulary use [40], we started by considering reuse of classes and properties from existing vocabularies. We used the common set of properties we defined in the previous step as a starting point and found that the Schema.org vocabulary [24] was fully applicable for our set. The Schema.org vocabulary, as part of its Health and Lifesciences Extension [9], contains a definition of the class `schema:Drug` and contains a large set of properties applicable to it [30]. As we can see on Fig. 3, the RDF schema uses the DrugBank ontology and the RDFS vocabulary, as well, for interoperability purposes.

Schema.org is a joint initiative of Google, Bing, Yahoo and Yandex, as a common vocabulary intended for structured markup on web pages [12, 10, 11]. It is used by these search engines to introduce rich snippets about people, events, products, movies, restaurants, books, tv shows, etc. It is also used in Google's Knowledge Graph, in emails confirming reservations and receipts (from restaurants, hotels, airlines, etc.) both from Gmail and Microsoft's Cortana, it is used for rich pins on Pinterest, as well as from Apple's Siri [51]. Its use on the Web has been increasing in the past few years, more rapidly than the more rigorous, general-purpose vocabularies and ontologies before it [71]. Its success is mainly attributed to its simplicity: it uses a generally flat hierarchy of classes, so that the boundaries of implementation for data publishers and webmasters is kept low.

The growing use of the Schema.org vocabulary, as well as its domain generality, has put the vocabulary in a position in which it is being used for aligning existing ontologies and datasets. This is happening in the healthcare domain, as well [83]. With the release of Schema.org version 3.0 [25], the medical and healthcare related terms [21] have been moved to the Health and Livesciences Extension [9], to enable and ensure future collaborative development of the terms by the Healthcare Schema Vocabulary community group at W3C [37, 89]. This plan for a long-term support by the community from the domain instills sufficient certainty for us to choose the Schema.org vocabulary, instead of the domain specific ontologies [54], to provide a common schema for drug products on a global scale.

In order to provide alignment between the generated datasets and the LODD and DrugBank datasets, we use several properties from the DrugBank ontology to describe the drug products. Additionally, each drug product is an instance of a specific class from the ATC Classification Ontology [29], in order to classify the drug according to the ATC classification system, based on its ATC code(s). We also chose `rdfs:seeAlso` as it is the most widely used relation for interlinking similar entities in the LOD Cloud [82].

Just as any other RDF schema, vocabulary and ontology, the RDF schema selected for our Linked Drug Data datasets can evolve over time; it can be extended and modified in the future by us or third-parties, as the field of drug data evolves.

CSV Template ($C_{3_a drugs}$)

In order to enable data publishers to annotate their drug data with the RDF schema from Fig. 3, as defined in Step III, we need a formal template for the data which is being prepared for

[13] Austria, Azerbaijan, Belgium, Canada, Costa Rica, Croatia, Cyprus, Czech Republic, Egypt, Estonia, EU's European Medicines Agency, France, Hungary, Ireland, Italy, Macedonia, Malta, Netherlands, New Zealand, Nigeria, Norway, Romania, Russia, Serbia, Slovakia, Slovenia, South African Republic, Spain, Uganda, Ukraine and USA.

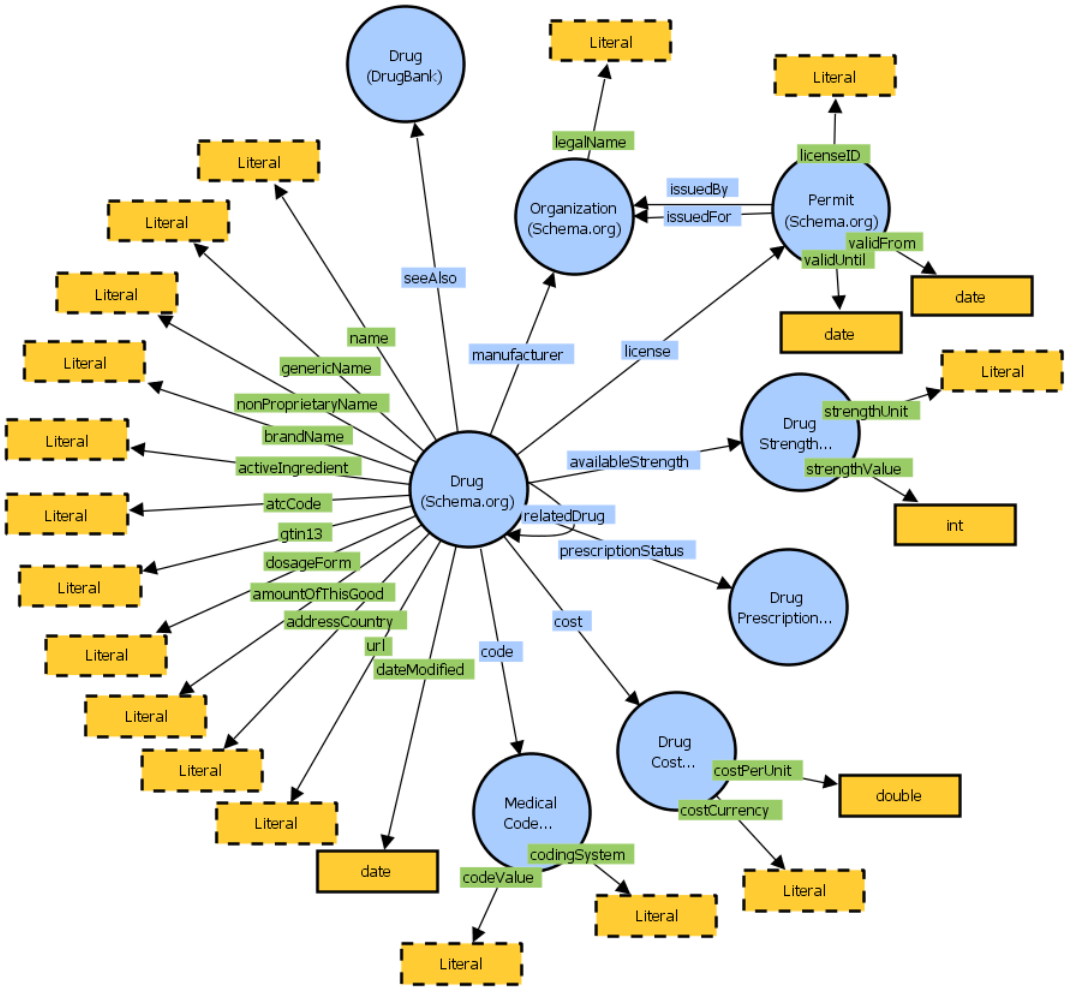

Figure 3: The RDF vocabulary designed for the drug data domain, comprised of Schema.org, DrugBank and RDFS classes and properties.

transformation, and a formal transformation process. For the former, we define a CSV template, available publicly and as open-source [32]. The CSV template contains 39 columns which represent the different data fields needed from the source data for the transformation process. They inlude the URI of the drug, its brand name, generic name(s), manufacturer(s), ATC code, active substance(s), strength, cost, etc. They are modeled to fit with the RDF schema, which encompasses all data necessary for high-quality modeling of the domain.

The data type of the different columns is usually a simple text value, except where we note otherwise. Some important notes regarding the field data types include: the strength value is divided into an integer-value column denoting the strength, while the unit is part of a text-value column denoting the strength unit; similarly, the cost of the drug is separated into a float value and a currency value, where the currency code needs to comply with the ISO standard for denoting currencies [14]; the several date columns need to be formatted as "YYYY-MM-DD"; the prescription status should be enumerated as either "OTC" or "PrescriptionOnly"; the country where the drug is registered in needs to be denoted using a country code according to an ISO standard [13]; if there are multiple generic names, manufacturers or active substances, they should be denoted one-per-column in the available `genericNameN`, `manufacturerN` and `activeSubstanceN` columns, respectively, etc. The details about the other column data types are available on the project website [32].

The CSV template uses a vertical line character (`|`) as a delimiter, since the regular CSV separators such as a coma (`,`) and a semicolon (`;`) are very often present in the cell values when working with drug data, and can therefore be misinterpreted. It is important to note that the order of the columns in the CSV template is not relevant, if used with our OpenRefine transformation script.

As with the RDF schema, the CSV template is open and publicly available, and therefore can be extended or modified in the future by both us and third-parties, as the drug data field evolves and more Linked Drug Data dataset are being created.

OpenRefine Transformation Script ($C_{3_b drugs}$)

Step III of the methodology contains the task of transforming the source data into the RDF schema selected in Step II. Since we defined an RDF schema which can be applied in the drug data domain for drug products which are registered in different countries, we also provide a tool as a reusable component which can help automate the transformation process, while ensuring compliance of the generated data with the defined RDF schema and therefore providing aligned, high-quality 5-star Linked Data for the drug domain. The intent of this tool is to lower the bounds of transforming data into RDF and Linked Data for data publishers which are not deeply involved and experienced in the Semantic Web and Linked Data practices, as well as aid the experienced Linked Data publishers which may not be familiar with the domain of drugs.

We provide this Linked Data generation tool in the form of an OpenRefine transformation script. OpenRefine [22] is an open-source software for working with structured data, usually CSV, TSV, XML, etc. It provides users with functionalities for working with large datasets: the users can record their action over a small set of example rows, and then apply them over the entire source data. Here, the actions can include data transformations, merging, data cleaning tasks, manipulation of the columns, etc. It also has an RDF extension which allows reconciliation of cell values against RDF data from SPARQL endpoints. This allows linking of cell values with entities from a SPARQL endpoint, for unambiguous identification of entities. It also allows mapping of the source data into RDF, by defining an 'RDF skeleton'. The output of this action is an RDF file generated from the source data, according to the definitions in the 'RDF skeleton'.

OpenRefine's ability to save the user actions and then export them in JSON format, allows reuse of the sets of actions for different datasets. This gives us the ability to define the data transformation which can be reused over different source drug datasets, which have the same columns. As we have a CSV template, we can use this as part as our set of tools. The defined list of data transformation actions we created is what we have as our OpenRefine transformation script [32].

Our OpenRefine transformation script is designed for data complying with the CSV template, and its output is a Linked Drug Data dataset which uses our RDF schema. The transformation script contains three actions:

A. reconcile the columns `genericName1`, `genericName2`, ..., `genericName5` against DBpedia,

B. reconcile the column `atcCode` against the DrugBank dataset, and

C. create an RDF schema skeleton

Action A. uses the RDF extension feature of OpenRefine which uses the cell value from a selected column to find potential entities from a given SPARQL endpoint which can be matched to the entity denoted by the row. In our case, we use the five `genericName` columns - which hold the generic name of the active substance of the drug entity - and we try to match each of them to a generic drug entity from the DBpedia SPARQL endpoint, using its `rdfs:label` value. If the reconciliation service finds a matching candidate entity, an instance of `dbo:Drug`, we use it in step C. to create an RDF triple which links the drug entity from our CSV dataset with the matched generic drugs from DBpedia, via an `rdfs:seeAlso` relation, for instance:

RDF Triples Example 1

```
@prefix dbp: <http://dbpedia.org/resource/>
@prefix mkd: <https://lekovi.zdravstvo.gov.mk/drugsregister/detailview/>

mkd:55446 rdfs:seeAlso dbp:Clopidogrel .
```

Action B. does a similar reconciliation, but on the `atcCode` column from the CSV dataset and against the DrugBank endpoint. It tries to find matches between the value of the `atcCode` column on our side and the `drugbank:atcCode` value of `drugbank:drugs` instances from the endpoint. Unlike the situation in A., here we can have more than one matching candidate from DrugBank. The reason is that there can be multiple `drugbank:drugs` instances which have the same ATC code, i.e. share the same therapeutic, pharmacological and chemical properties. Similar as in A., we use all matching candidates from the reconciliation in step C. to create RDF triples which link the drug entity from our CSV dataset to the matched generic drug entities from DrugBank, such as:

RDF Triples Example 2

```
@prefix mkd: <https://lekovi.zdravstvo.gov.mk/drugsregister/detailview/>
@prefix dbd: <http://wifo5-04...uni-mannheim.de/drugbank/resource/drugs/>

mkd:841690570 rdfs:seeAlso dbd:DB00201 ;
              rdfs:seeAlso dbd:DB00316 .
```

Action C. creates the RDF schema skeleton, which contains the rules for mapping the consolidated CSV file into RDF. In the RDF schema skeleton (Fig. 3), we define mappings between the CSV columns and certain RDF triple patterns. Some of the mappings are straight-forward, such as the mappings of the brand name, the generic name, the dosage form, the country, the url, the description, etc. For them, we define the URI of the drug as a subject, we denote a specific property for the triple, and then we define the value of the column as a literal or an object of the triple. For instance, the brand name of a drug is mapped into RDF triples with the following format:

Mapping Example 1

```
<drug-URI> schema:name <value-of-brandName-column> ;
           drugbank:brandName <value-of-brandName-column> .
```

However, other mappings are more complex. Mappings of values such as the ATC code, the cost, the strength, the manufacturer, the license details, etc., need new entities to be created, entities of different types. For instance, in order to add the information about the ATC code to the drug entity, we need to create a new blank node of type `schema:MedicalCode`, which has two additional triples: one with the `schema:codeValue` property and one with the `schema:codingSystem` property. This ATC code mapping can be represented with:

Mapping Example 2

```
<drug-URI> schema:code <blank-node-ID> .
<blank-node-ID> rdf:type schema:MedicalCode ;
                schema:codeValue <value-of-atcCode-column> ;
                schema:codingSystem ‘‘ATC’’ .
```

The license mappings were the most complex, which we can see from Fig. 3:

Mapping Example 3

```
<drug-URI> schema:license <blank-node-1-ID> .
<blank-node-1-ID> rdf:type schema:Permit ;
      schema:licenseID <value-of-licenseNumber-column> ;
      schema:validFrom <value-of-licenseValidFrom-column>^^xsd:date ;
      schema:validUntil <value-of-licenseValidUntil-column>^^xsd:date ;
      schema:issuedBy <blank-node-2-ID> ;
      schema:issuedFor <blank-node-3-ID> .
<blank-node-2-ID> rdf:type schema:Organization ;
      schema:legalName <value-of-licenseIssuedBy-column> .
<blank-node-3-ID> rdf:type schema:Organization .
      schema:legalName <value-of-licenseIssuedFor-column> .
```

Aside from using OpenRefine's user interface for defining the RDF skeleton, we used its GREL language for mapping the reconciliation results from actions A. and B. into `rdfs:seeAlso` triples.

As a result of the transformation script, a Linked Data dataset with links to the LOD Cloud is created. Similarly as the other tools, the transformation script is available as an open-source JSON file, which can be extended and modified in the future.

SPARQL-Based Tool for Extending and Interlinking the Dataset ($C_{3_c drugs}$)

Once the drug dataset is transformed into a Linked Data dataset with the other tools, an additional action is required in Step III to create the internal links between drugs which share the same functionality, i.e. share the same therapeutic, pharmacological and chemical properties, in order to create a better basis for use-cases. We need to create links between drugs from the dataset which have the same function, i.e. are aimed to treat the same condition. To create these links, we use drug's ATC codes. According to the World Health Organization coding scheme [1], if two drugs have the same ATC code, they share the same function. For this purpose, we define a reusable SPARQL query [32] which detects all pairs of drugs from the dataset which have the same ATC code, and using the `schema:relatedDrug` property creates a pair of triples for them, for instance:

RDF Triples Example 3

```
@prefix rus: <http://www.vidal.ru/drugs/>
@prefix mkd: <https://lekovi.zdravstvo.gov.mk/drugsregister/detailview/>

rus:trombopol__22439 schema:relatedDrug mkd:51201 .
mkd:51201 schema:relatedDrug rus:trombopol__22439 .
```

These two triples create a two-way link between the drugs in the dataset, denoting their functional similarity. The SPARQL query results with storing the newly created RDF triples in the same RDF graph where the dataset is already stored. These interlinkings can be utilizes for providing the users with alternative drugs they may require for treating their condition, either in the same of in a different country.

Since not all source registries contain the ATC code information, and in order to increase the number of interlinked drug products from the dataset and support better data analytics, we define an additional reusable SPARQL query [32] which assignes ATC codes to all drug products from the dataset which miss this information. The SPARQL query detects drugs without an ATC code, finds the generic drug from DBpedia which the drug product is linked to with the `rdfs:seeAlso` relation, gets the ATC code of the DBpedia generic drug and assigns it to the drug product in question. Since the SPARQL query for interlinking drugs from the dataset depends on the ATC code, this SPARQL query for extending the dataset with missing ATC code values should be executed first.

Both SPARQL queries are parametrized and should be edited before execution. They can be executed over the Linked Data storage used for storing the Linked Data dataset generated with the other tools.

Web-Based Tool for Automated Transformation, Interlinking and Publishing (C_{4drugs})

The generated Linked Drug Data dataset needs to be published on the Web according to the Linked Data principles and best practices, as advised in Step IV. In order to aid the data publishers, this step can be automatically executed by using a web-based tool we provide. The data publishers can upload the generated Linked Data dataset(s) on the LinkedDrugs project website [33], and after a human-based quality assessment, the dataset will be automatically published. For this we use a publicly available Virtuoso instance [34], from which the new dataset is available on the Web as Linked Data, via its SPARQL endpoint [28]. The RDF graph identifier is returned to the data publisher after the successful upload process.

Besides publishing finished Linked Drug Data datasets, the web-based tool and its automated process can also execute the previous steps of the methodology for the data publisher: (a) they can generate an interlinked Linked Data dataset from an input CSV file, and (b) they can extend the dataset with missing ATC codes and interlink drugs with `schema:relatedDrug` relations from an input RDF file. For the former, the uploaded CSV file needs to be generated following our CSV template, and based on it, the predefined RDF schema and the OpenRefine transformation script, our web-based tool and its server-side process will generate the Linked Data dataset. Using the SPARQL-based tool from above, it will then extend the dataset with missing ATC codes and generate links between the drugs from the dataset, based on their ATC codes. For the latter, the web-based tool directly creates the missing `drugbank:atcCode` relations and adds `schema:relatedDrug` relations between similar drugs from the uploaded Linked Drug Data dataset in RDF. With this, we provide the convenience to move most of the data processing from the methodological guidelines away from the data publishers, and simplify their workflow.

When a data publisher uses our web-based tool at [33] to publish a Linked Drug Data dataset, our system also adds it to the global Linked Drug Data dataset - the LinkedDrugs dataset - by storing it in another RDF graph and generating `schema:relatedDrug` triples for linking the drugs from the new dataset with the drugs from the existing datasets in LinkedDrugs, and vice-versa. The LinkedDrugs dataset then contains data for drug products provided by different publishers, including our team, and is available via a permanent, dereferenceable URI, which supports HTTP content negotiation [23].

Applying the Methodology in the Drug Data Domain

After developing the reusable components for the drug product domain, we applied the tuple $T_{drugs} = (C_{2drugs}, C_{3drugs}, C_{4drugs}) = (C_{2drugs}, (C_{3_a drugs}, C_{3_b drugs}, C_{3_c drugs}), C_{4drugs})$, i.e. the outlined steps, within a concrete project. The project consisted of designing and developing an automated system for transforming and generating 5-star Linked Drug Data from twenty-three different countries: Austria, Azerbaijan, Costa Rica, Cyprus, Egypt, Estonia, Ireland, Macedonia, Malta, Netherlands, New Zealand, Nigeria, Norway, Romania, Russia, Serbia, Slovakia, Slovenia, South African Republic, Spain, Uganda, Ukraine and USA. The countries were chosen to represent the global diversity and to show that a holistic solution for generating Linked Drug Data on a global scale is possible.

This automated system and its workflow represent a concrete example of applying the methodological guidelines and reusable components developed as part of the thesis, and thus serve as their validation scenario.

Generating the LinkedDrugs Dataset

The national drug registries of many countries around the world are available online. As we already outlined, we analyzed the national drug registry websites of 32 countries in order to define

a common set of properties, i.e. a schema skeleton, for the target Linked Drug Data dataset. These steps of domain analysis and RDF schema definition correspond to the activities denoted in Step I and Step II of the methodological guidelines, which are already done as part of the C_{2drugs} component, and we directly use them for our specific application.

In order to design, test and validate our automated system for gathering 2-star drug data from the national drug registries and generating 5-star Linked Data from the drug domain on a global scale, we selected a subset of twenty-three countries. We aimed for a diverse subset, which will encompass different global regions.

The drug registries of these countries are available online. Their websites are listed in the project page on GitHub [32]. The drug data from most of the national registries is available in a structured format in HTML pages, intended for human consumption via searching and browsing on the website itself. The data from a smaller group of countries is available via structured files in Microsoft Excel or PDF formats, available for direct download.

In this Section, we describe the automated system which gathers the data, performs data cleaning, aligns the data with the predefined schema skeleton ($C_{3_a drugs}$), uses the transformation script ($C_{3_b drugs}$) and the SPARQL queries ($C_{3_c drugs}$) to transform the data to RDF, extend it with missing ATC codes and add links to drugs from the same dataset and drugs from DrugBank and DBpedia, thus turning the dataset into a Linked Data dataset. The workflow of these actions is depicted in Fig. 4. These steps represent the activities defined in Step III of the methodological guidelines.

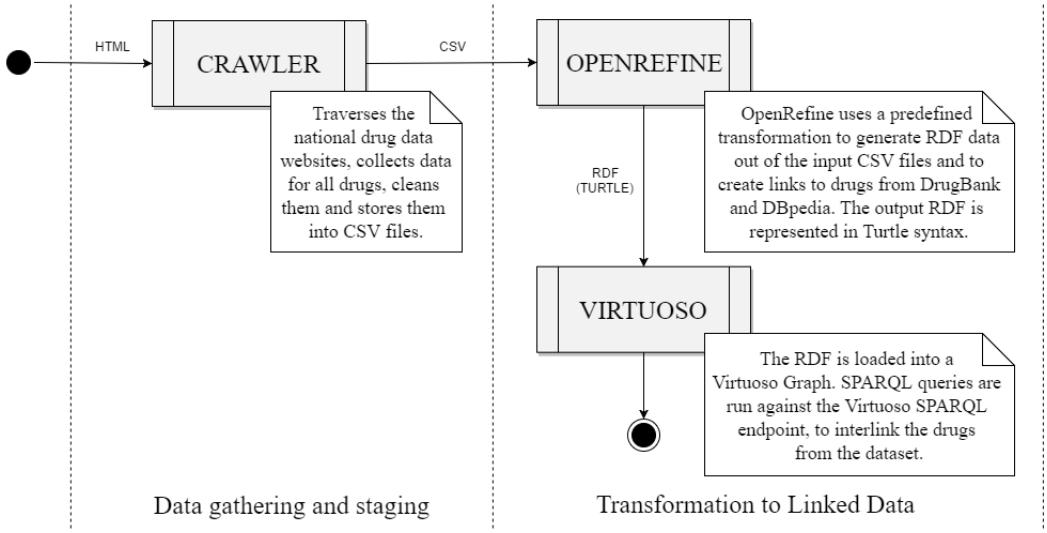

Figure 4: Workflow: Transforming 2-star data from different national drug data registries to 5-star Linked Drug Data.

Data Gathering and Staging. In order to create a sustainable system for Linked Drug Data, we had to design a way to collect the necessary data from the national drug registries, on a scheduled basis. Therefore, we developed a set of specialized web crawler applications which crawl the designated drug registry websites, collect the necessary data, clean it and store it in a predefined CSV format (Fig. 4). We need to use a set of such crawlers as the target websites differ in structure and available data. The output CSV files from the crawlers use the predefined CSV structure described above.

Like most of the data available on the Web, the drug data available on the national drug registry websites is not evenly structured, nor completely clean. We therefore needed to extend our crawlers with functionalities which perform data cleaning tasks and work to detect data from all variations of the source webpages.

In order to define the unique URIs of the drugs from the dataset, we used the URLs from

their specific web pages, e.g. `https://lekovi.zdravstvo.gov.mk/drugsregister/detailview/53457`. According to the Linked Data principles, the entity URIs need to be web locations where users and software agents can get more information about the entity, and this approach satisfies that requirement.

As many of the drug entities contain information about more than one generic name, manufacturer, active substance and strength, the crawlers are tasked with splitting them into the corresponding columns in the CSV files. Additionally, information about the cost of the drug and its strength are split into separate fields for value and currency / unit, to match the CSV template. The crawlers also take care about the specific formats needed for some of the columns, such as the dates, the country codes, the currency codes, and the prescription status.

The drug data from the several of the countries were an exception, as they are available for download as Microsoft Excel or PDF files. For these datasets, the crawlers have to restructure the columns from the source data and generate a CSV file with the correct column names according to the CSV template. For these drugs, we generate custom URIs as identifiers, which have the format `http://linkeddata.finki.ukim.mk/lod/data/loddw/drugs/{countryCode}#{drugID}`. Here, `drugID` is an ID generated by the crawler, `countryCode` is a three-letter country code (according to [13]) of the country where the drug was registered and the other parts of the URI identify the project and the datatype on our Linked Data website: `/lod/data/loddw` is the project and `/drugs` is the categorization of the data.

The result of this stage in the workflow, in our case with the twenty-three selected countries, is a set of twenty-three separate CSV files which follow the CSV template. The only difference is that some of the CSV files can have no values in some columns, due to the data being unavailable online. When we get all twenty-three CSV files, the first part of the workflow is done and we can continue with the second part.

Transformation to Linked Data. The CSV files can be combined into one CSV file, or remain separate. The only difference will be in the performance of the next step which can be done as a one longer process, or as twenty-three separate and shorter processes. To keep the processing time per transformation shorter, we use twenty-three separate CSV files, each representing the the drug dataset of a separate country.

We send the twenty-three CSV files to a BatchRefine instance [2], which represents a wrapper over an OpenRefine instance with the RDF extension, that can be used as a REST-based service. The HTTP POST calls to the BatchRefine REST-based service are done with a BASH script and contain (a) the CSV file which needs to be transformed and (b) the OpenRefine transformation script defined as a supporting tool for the guidelines. The result of the call is a transformed RDF output, which contains part of the generated Linked Drug Data dataset.

The output of our BatchRefine transformations are twenty-three RDF files in Turtle format. These RDF files are a Linked Data dataset: they contain 5-star data about the drugs from the twenty-three countries, along with links to generic drugs from the LOD Cloud - more specifically, links to generic drugs from both DBpedia and DrugBank. As we will see later in the text, we can use these links to fetch data about the drugs from our dataset which we don't have on our end and which do not exist on the source national drug registry websites, but can be found in other datasets on the Web and can potentially prove to be of interest for the end-users.

After the transformation with BatchRefine is done, we load the RDF files into a Virtuoso instance [34] using a BASH script. All RDF files are loaded into the same RDF graph. The latest run of the workflow (Fig. 4) resulted in over 248,000 distinct drugs in this step, represented with a total of over 7,450,000 RDF triples and with over 278,000 outgoing links to drugs from the LOD Cloud.

After the RDF data has been stored into an RDF graph in Virtuoso, we use the SPARQL queries for extending the dataset with missing ATC codes and interlinking related drugs, as described previously. We execute the SPARQL queries over our dataset stored in Virtuoso, using a BASH script. In the latest run of the workflow (Fig. 4), 38,000 new ATC codes were added for drug products which did not have an ATC code in the source registry. Then, 91,780,000 'schema:relatedDrug' triples were added between similar drug products, i.e. 45,890,000 pairs of drugs from our Linked Drug Data dataset were identified to have the same function, but exist

under different brand names, are from different countries, or are produced by different manufacturers, or maybe have a different packaging size, strength, cost, etc. As we will see further in the text, we can utilize these interlinkings for providing the users with alternative drugs they may aquire for treating their condition, either in the same of in a different country.

The workflow shown in Fig. 4 is activated on a scheduled period, currently set at one month. In order to handle the data changes during updates, we backup the RDF graph holding our dataset and replace it with the newly created RDF graph. With this we employ versioning and maintain the default graph identifier to always denote the latest version of the LinkedDrugs dataset. The latest run of the workflow resulted in over 99,000,000 RDF triples total in the LinkedDrugs dataset.

According to the recommendations in Step IV of our guidelines, the dataset needs to be published on the Web, where it will be publicly available. Therefore, we published our Linked Drug Data dataset according to the best practices for publishing Linked Data [3], via a permanent, dereferenceable URI, which supports HTTP content negotiation: `http://linkeddata.finki.ukim.mk/lod/data/drugs#` [23]. The dataset is hosted at a live Virtuoso instance [34], in a named RDF graph `<http://linkeddata.finki.ukim.mk/lod/data/drugs#>` which holds the latest version of the dataset, publicly available via a SPARQL endpoint [28] which serves as a REST-based service.

Additionally, data dumps of the dataset are available on Datahub.io [31].

With this, the processing of the twenty-three source drug dataset with the reusable tuple T_{drugs}, is done.

Usage Scenarios over the LinkedDrugs Dataset

With the automated system and its workflow we start with twenty-three different, distributed and browsable datasets, available on the Web and intended for human consumption via HTML webpages, and using the methodological guidelines and tools we manage to create a consolidated dataset of interlinked and schema-aligned drug products from different countries, additionally linked to generic drugs from the LOD Cloud. In order to demonstrate the advantages of having the drug data in a 5-star data quality format, and therefore implement the recommendations from Step V of the methodological guidelines, we will show a few use-case scenarios via SPARQL queries. The most basic scenario would be to select all data about a single drug of interest, which is very simple and straight-forward, and therefore omitted here. Since the Virtuoso SPARQL endpoint at [28] can be used as a REST-based service, these SPARQL queries could be used from any type of application to always access and exploit the latest data available.

Interlinked Drug Data. The interlinking created between the drugs from the dataset, with the `schema:relatedDrug` triples, can be utilized in a use-case which allows the end-users to discover drugs from the same country or another country which have the same therapeutic, pharmacological and chemical properties as the drug of his / her interest. This is a useful feature when the drug of interest is not available or when the user is traveling abroad. Getting information about drugs with the same properties and their respective prices can be useful for determining the drug from a specific category which is affordable in the specific case. This can be used by pharmacists, doctors and even patients for gathering information and determining the appropriate treatment. An example SPARQL query which can be used to identify the drugs from all countries which have the same therapeutic, pharmacological and chemical properties as the drug of interest, is shown below.

Query 1

```
prefix schema: <http://schema.org/>
prefix drugbank: <http://www4.wiwiss.fu-berlin.de/drugbank/resource/drugbank/>

SELECT ?drug ?name ?gtin ?strengthValue ?strengthUnit ?costPerUnit
?costCurrency ?manufacturerName ?prescriptionStatus ?country
FROM <http://linkeddata.finki.ukim.mk/lod/data/drugs#>
WHERE {
```

```
8     <http://www.legemiddelverket.no//Legemiddelsoek/Sider/Legemiddelvisning.
9                             aspx?pakningId=c5a02a30-d471-4ba0-8197-38955f384dd8>
10        schema:relatedDrug ?drug .
11    OPTIONAL { ?drug schema:name ?name }
12    OPTIONAL { ?drug schema:gtin13 ?gtin }
13    OPTIONAL { ?drug schema:prescriptionStatus ?prescriptionStatus }
14    OPTIONAL { ?drug schema:addressCountry ?country }
15    OPTIONAL {
16      ?drug schema:cost ?costEntity .
17      ?costEntity schema:costPerUnit ?costPerUnit ;
18                  schema:costCurrency ?costCurrency .
19    }
20    OPTIONAL {
21      ?drug schema:availableStrength ?strengthEntity .
22      ?strengthEntity schema:strengthValue ?strengthValue ;
23                      schema:strengthUnit ?strengthUnit .
24    }
25    OPTIONAL {
26      ?drug schema:manufacturer ?manufacturerEntity .
27      ?manufacturerEntity schema:legalName ?manufacturerName .
28    }
29  }
30  ORDER BY ?name
```

In the query, only a handful of data of interest of the related drugs are selected, but depending on the specific use-case, they can be expanded. Query 1 can also be modified to include the specific drug of interest, by modifying the drug URI in line 8. In our example in Query 1, we use the Norwegian drug "Airomir" as an example, and get results for over 300 distinct and related drugs from many different countries in the dataset. The query and its full results can be viewed online on Seminant [26], at `http://seminant.com/queries/5803e77573656d19eb6c5d00`. Partial results are shown in Table 2.

Table 2: Partial results from Query 1.

Drug Product	Manufacturer	Country
Activent Sr	Medical Union Pharmaceuticals - Egypt	EG
Aerolin 100mcg/dose Inhaler		EG
Aeroline 400 Inhaler		EG
Aerotropa	Pharco B International-egpyt	EG
Agolin	Agog Pharma Ltd	UG
Airomir	iNova Pharmaceuticals (New Zealand)	NZ
Airomir Autohaler	Teva Sweden AB	NO
Airomir Autohaler	iNova Pharmaceuticals (New Zealand)	NZ
Airomir Autohaler 100 microgramos	Teva Pharma S.L.U.	ES

Linked LOD Drug Data. The main advantage of having links between data from different and distributed datasets is the ability to query them from a single point, over the existing infrastructure of the Web, using W3C standards such as HTTP, SPARQL and RDF. As we have `rdfs:seeAlso` links from drugs in our dataset to corresponding generic drugs from the DrugBank and DBpedia datasets, we can utilize them to get additional information about the drugs from our dataset whenever we are browsing them. Such additional information will come from the

DrugBank and DBpedia datasets, and can include additional drug description, the interactions the drug has with other drugs or with certain foods, the drug mechanism of action, the drug pharmacology, absorption, biotransformation and toxicity, the list of alternative brand names and a list of webpages for the drug on other locations on the Web. This data is not available on the original national drug registry websites, which are the source for our dataset; it is data retrieved directly from the distributed DrugBank and DBpedia dataset, using SPARQL federation [78].

An example of a federated SPARQL query which selects information about a drug of interest from the DrugBank and DBpedia datasets is shown below.

Query 2

```
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix schema: <http://schema.org/>
prefix drugbank: <http://wifo5-04.informatik.uni-mannheim.de/drugbank/resource/drugbank/>
prefix dbo: <http://dbpedia.org/ontology/>
prefix dbp: <http://dbpedia.org/property/>

SELECT ?loddrug ?genericName
(group_concat(distinct ?brandName; separator = ", ") AS ?brandNames)
?comment ?description ?biotransformation ?affectedOrganism ?absorption
?chemicalFormula ?toxicity
(group_concat(distinct ?foodInteraction; separator = " ") AS ?foodInteractions)
(group_concat(distinct concat(?interactingDrugLabel, ': ', ?interactionStatus);
                                        separator = ". ") AS ?drugInteractions)
(group_concat(distinct ?url; separator = ", ") AS ?urls)
WHERE {
  GRAPH <http://linkeddata.finki.ukim.mk/lod/data/drugs#> {
      <http://www.aemps.gob.es/cima/especialidad.do?metodo=verPresentaciones&codigo=79539>
              rdfs:seeAlso ?loddrug .
  }
  SERVICE <http://wifo5-04.informatik.uni-mannheim.de/drugbank/sparql> {
      OPTIONAL { ?loddrug drugbank:description ?desc . }
      OPTIONAL { ?loddrug drugbank:genericName ?gname . }
      OPTIONAL { ?loddrug drugbank:brandName ?bname . }
      OPTIONAL { ?loddrug drugbank:biotransformation ?biotransformation . }
      OPTIONAL { ?loddrug drugbank:affectedOrganism ?affectedOrganism . }
      OPTIONAL { ?loddrug drugbank:absorption ?absorption . }
      OPTIONAL { ?loddrug drugbank:chemicalFormula ?chemicalFormula . }
      OPTIONAL { ?loddrug drugbank:foodInteraction ?foodInteraction . }
      OPTIONAL { ?loddrug foaf:page ?page . }
      OPTIONAL { ?loddrug drugbank:toxicity ?toxicity . }
      OPTIONAL {
          ?drugInteractionEntity drugbank:interactionDrug1 ?loddrug ;
                                 drugbank:interactionDrug2 ?interactingDrug ;
                                 drugbank:text ?interactionStatus .
          ?interactingDrug rdfs:label ?interactingDrugLabel .
      }
  }
  SERVICE <http://dbpedia.org/sparql> {
      OPTIONAL {
          ?loddrug dbo:abstract ?abstract .
          FILTER (langMatches(lang(?abstract), "en"))
      }
      OPTIONAL {
          ?loddrug rdfs:label ?label .
          FILTER (langMatches(lang(?label), "en"))
      }
      OPTIONAL {
          ?loddrug rdfs:comment ?comment .
```

```
49            FILTER (langMatches(lang(?comment), "en"))
50        }
51        OPTIONAL { ?loddrug dbp:tradename ?tradename . }
52        OPTIONAL { ?loddrug dbo:wikiPageExternalLink ?externalLink . }
53    }
54    BIND(IF(bound(?abstract), ?abstract, ?desc) as ?description)
55    BIND(IF(bound(?bname), ?bname, ?tradename) as ?brandName)
56    BIND(IF(bound(?gname), ?gname, ?label) as ?genericName)
57    BIND(IF(bound(?page), ?page, ?externalLink) as ?url)
58  }
```

The query selects some very important data about the drug of interest and its active ingredient from DrugBank and DBpedia. The most significant are the chemical, biological and pharmacological properties of the drug, along with its interactions with food and with other drugs. This data is not always available on the national drug data registries, but is of high importance for the end-users, especially the pharmacists and doctors who may require them when determining treatment for acute conditions of a patient who is already on a treatment of a chronic medical condition.

Table 3: Partial results from Query 2.

Description (from DBpedia)
Duloxetine (Cymbalta, and generics) is a serotonin-norepinephrine reuptake inhibitor (SNRI) created by Eli Lilly. It is mostly prescribed for major depressive disorder, generalized anxiety disorder, fibromyalgia and neuropathic pain. Duloxetine failed to receive US approval for stress urinary incontinence amid concerns over liver toxicity and suicidal events; however, it was approved for this indication in the UK, where it is recommended as an add-on medication in stress urinary incontinence instead of surgery.
Food Interactions
Food does not affect maximum levels reached, but delays it (from 6 to 10 hours) and total product exposure appears to be reduced by only 10 percent. People taking this product who drink large amounts of alcohol are exposed to a higher risk of liver toxicity. Take without regard to meals.
Drug Interactions
Amitriptyline: Possible increase in the levels of this agent when used with duloxetine. Ciprofloxacin: Ciprofloxacin increases the effect/toxicity of duloxetine. Desipramine: Possible increase in the levels of this agent when used with duloxetine. Flecainide: Possible increase in the levels of this agent when used with duloxetine. Fluvoxamine: Fluvoxamine increases the effect and toxicity of duloxetine. Imipramine: Possible increase in the levels of this agent when used with duloxetine. Isocarboxazid: Possible severe adverse reaction with this combination. Nortriptyline: Possible increase in the levels of this agent when used with duloxetine. Phenelzine: Possible severe adverse reaction with this combination. Propafenone: Possible increase in the levels of this agent when used with duloxetine. Rasagiline: Possible severe adverse reaction with this combination. Thioridazine: Increased risk of cardiotoxicity and arrhythmias. Tranylcypromine: Possible severe adverse reaction with this combination

An example run of Query 2, for the drug product "Duloxetina" from Spain, results in details for the generic drug "Duloxetine" from both DrugBank and DBpedia: `http://seminant.com/queries/5803e9b973656d19eba65e00`. Among other details, it also shows the 3 specific food -

drug interactions the drug is involved in, along with the 13 drug - drug interactions it has. Partial results from the query are shown in Table 3.

Analytics. Aside from the use-case scenarios for end-users, our Linked Drug Data dataset can be used for analytical queries as well. These analytical queries allow interested parties to gain insight into the drug markets of different countries, allowing them to analyze the available consolidated data using a single entry point for querying and using a single query language. The analytics could be built-in in specific analytic applications, or can be executed with separate and standalone SPARQL queries.

To get a better understanding of the analytical possibilities over consolidated drug data from multiple countries, we will look at an example query which identifies the most common drug categories per country. This would allow the user, e.g. pharmaceutical company, to gain a better knowledge on the national drug markets and make an informed decision about placing their drug in the country of interest. A general SPARQL query for this analytical scenario is given below:

Query 3

```
prefix schema: <http://schema.org/>
prefix drugbank: <http://www4.wiwiss.fu-berlin.de/drugbank/resource/drugbank/>

SELECT count (distinct ?drug) as ?count ?atc ?country
FROM <http://linkeddata.finki.ukim.mk/lod/data/drugs#>
WHERE {
      ?drug a schema:Drug ;
          schema:addressCountry ?country ;
          drugbank:atcCode ?atcCode .
      FILTER (strlen(?atcCode) > 3)
      BIND(SUBSTR(xsd:string(?atcCode), 1, 3) AS ?atc)
}
GROUP BY ?country ?atc
ORDER BY DESC (?count)
```

A sample run of Query 3 shows that Romania, Spain, Netherlands, Ireland and Slovakia have most drugs in the category "Agents acting on the renin-angiotensin system" (ATC C09), Russia and South African Republic have most drugs in the category "Antibacterials for systemic use" (ATC J01), while USA has most drugs in the "Psycholeptics" (ATC N05) category. These partial results are shown in Table 4. The full results from the query are available at `http://seminant.com/queries/5803ebc473656d19ebac5e00`.

Table 4: Partial results from Query 3.

Drugs	ATC Prefix	Country
5362	C09	RO
2152	C09	ES
1536	J01	RU
1488	C09	NL
1270	N05	US
976	J01	ZA
758	C09	IE
709	C09	SK
707	N02	NZ

Another analytical scenario would be to assess the average drug price per drug category, per country. It could be used by medical authorities in a country to determine the cost situation per category in other countries and use the information for local regulations. It could also be used by a pharmaceutical company to determine the price range for a new drug, before it goes to market. An example SPARQL query which can be used for such an analysis is given below:

Query 4

```
prefix schema: <http://schema.org/>
prefix drugbank: <http://www4.wiwiss.fu-berlin.de/drugbank/resource/drugbank/>

SELECT (?totalCost / ?drugCount as ?averageCost) ?costCurrency ?atc
       ?drugCount ?country
WHERE {
  SELECT count (distinct ?drug) as ?drugCount
         sum (xsd:float(?cost)) as ?totalCost
         ?costCurrency ?atc ?country
  FROM <http://linkeddata.finki.ukim.mk/lod/data/drugs#>
  WHERE {
       ?drug a schema:Drug ;
           schema:addressCountry ?country ;
           drugbank:atcCode ?atcCode ;
           schema:cost ?costEntity .
       ?costEntity schema:costPerUnit ?costPerUnit ;
               schema:costCurrency ?costCurrency .
       FILTER (strlen(?atcCode) > 3)
       BIND(SUBSTR(xsd:string(?atcCode), 1, 3) AS ?atc)
       FILTER (?costPerUnit != "0"^^xsd:double)
       BIND(REPLACE(?costPerUnit, ",", ".") AS ?cost)
  }
}
GROUP BY ?country ?atc ?costCurrency
ORDER BY DESC (?averageCost)
```

Table 5: Partial results from Query 4.

Avg. Price	Currency	ATC Prefix	Country
93480.60	NOK	M09	NO
47221.40	NOK	R07	NO
39557.30	NOK	A16	NO
32021.40	MKD	A16	MK
28837.20	MKD	H01	MK
27478.20	MKD	B02	MK
22500.00	AUD	R07	AU
17822.00	SVN	R07	SI
13360.10	EUR	V10	CY
10679.50	ZAR	LO4	ZA
10127.10	ZAR	B06	ZA
9880.81	ZAR	A16	ZA

A sample run of Query 4 identifies that the ATC drug categories with the highest average price in Norway are "Other drugs for disorders of the musculo-skeletal system" (ATC M09), "Other respiratory system products" (ATC R07) and "Other alimentary tract and metabolism products" (ATC A16). In Macedonia they are "Other alimentary tract and metabolism products" (ATC A16), "Pituitary and hypothalamic hormones and analogues" (ATC H01) and "Antihemorrhagics" (ATC B02), while in Australia and Slovenia they are "Other respiratory system products" (ATC R07) and in South African Republic they are "Immunosuppressants" (ATC L04), "Other hematological agents" (ATC B06) and "Other alimentary tract and metabolism products" (ATC A16). These partial results are shown in Table 1, while the full results which include other countries as well, are available at `http://seminant.com/queries/5803ed6e73656d19eb537e00`.

In cases when an inter-country comparison of the pricing is necessary, an application could use a currency converter to transform the values to the same currency of choice, and make the comparison.

The "Global Open Drug Data (GODD)" Web Application

In the time of writing this PhD thesis, the generated dataset is being used by the "Global Open Drug Data (GODD)" web application [8] (Fig. 5), developed by a group of students from the Faculty of Computer Science and Engineering in Skopje. This application solely uses our Linked Drug Data dataset as a data layer.

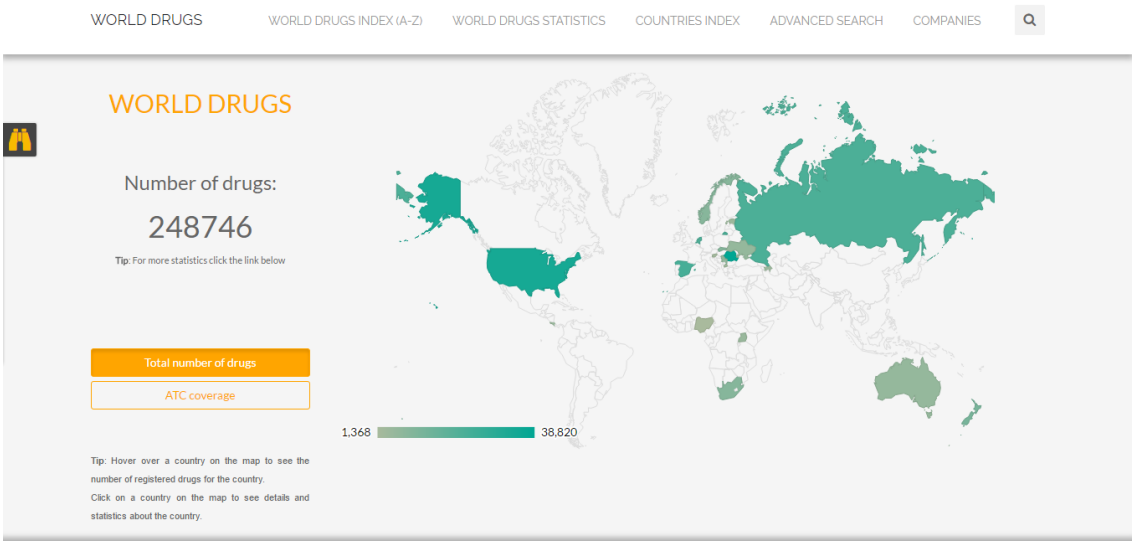

Figure 5: The landing page of the "Global Open Drug Data (GODD)" web application.

Based on the data from the dataset, the application provides a set of services for its users: an overview of the drug details, taken both from the source drug registries and from LOD Cloud datasets, a comparative analysis of single drug products and groups of drugs from different countries; an analysis of the global coverage of drugs and ATC categories, as well as a coverage on a country level; a comparative analysis of the absolute and average prices of drug products and ATC categories in different countries, etc. (Fig. 6).

This application demonstrates exactly those advantages which motivated us to generate one such Linked Drug Dataset, i.e. they demonstrate a plethora of user-centric and analytical scenarios which can be of great importance to patients, pharmacists, doctors, pharmaceutical companies and healthcare authorities, on a global scale.

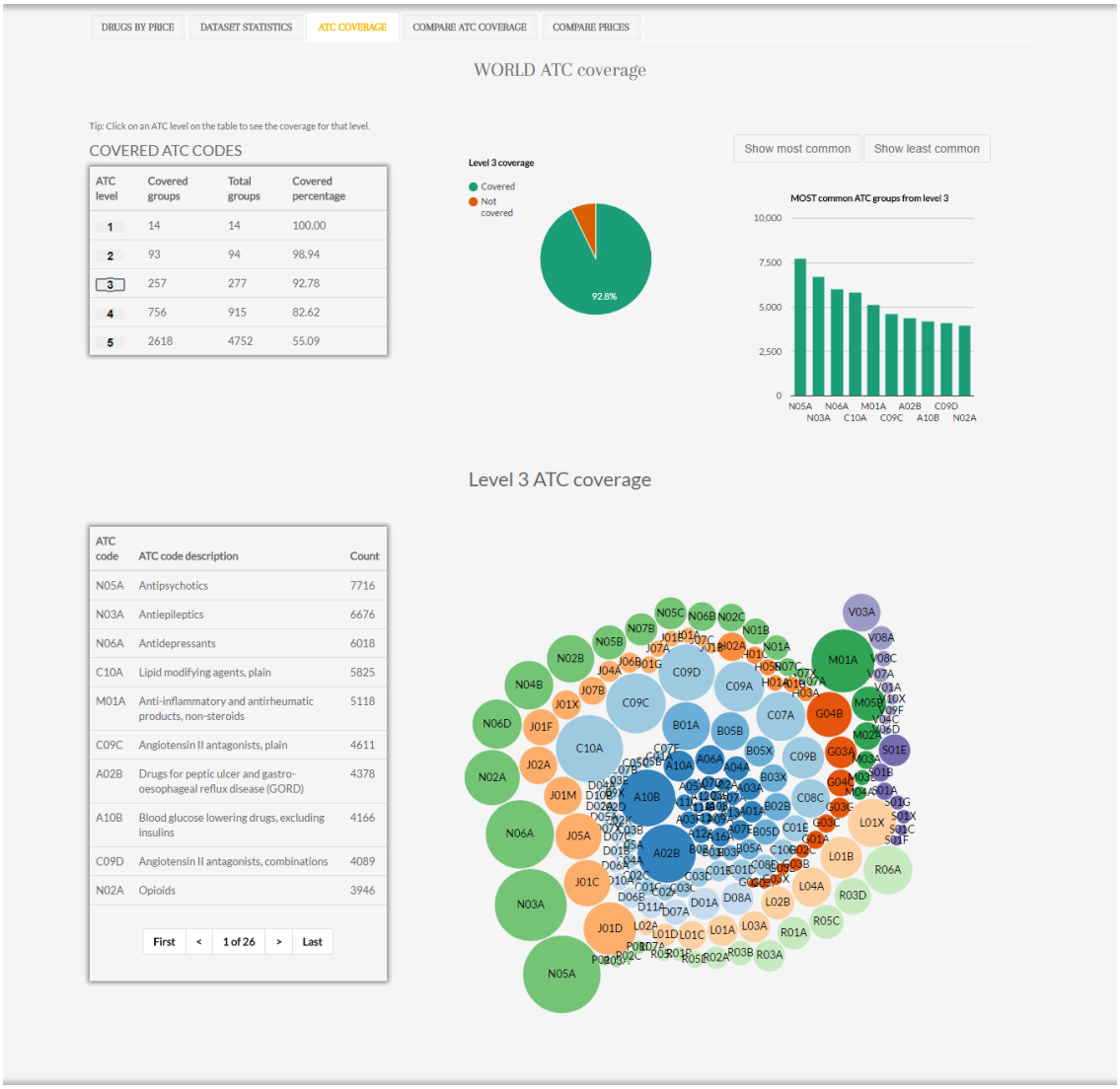

Figure 6: The global drug coverage at level 3 of the ATC classification.

Conclusion

The main objective of this PhD thesis was to develop a general Linked Data methodology which encompasses the entire life-cycle of a Linked Data dataset from any given domain: identification, modeling, annotation, transformation, publication and effective use, with a specific focus on process reuse. In order to achieve this, we needed to: (a) implement the Linked Data life-cycle in various domains, and (b) analyze the existing Linked Data methodologies.

The former provided us with practical analysis of the methods and techniques which can be used in each of the life-cycle steps for a Linked Data dataset in a given domain. We drew experience on the different ways to obtain the source data, different modeling approaches, the best ontology (re)use practices, the varying methods for data transformation which can range from manual to fully automated, the different ways of publishing the dataset on the Web, as well as the extensive ways the datasets can be used in real-world applications and services. Many of the steps and decisions are highly dependent on the type or domain of the source data, while there are also approaches which are influenced by the purpose of the generated Linked Data dataset. We tried to identify all of these specifics and best practices, via the numerous research projects in the last few years, presented in Chapter 4: 'Transformation and Usage of Linked Data in Various Domains' of the main PhD thesis manuscript and in brief in chapter 'Content of the Thesis' in this executive summary.

The latter enabled us to identify the specifics, advantages and drawbacks of the existing Linked Data methodologies. We managed to detect a general set of steps in each life-cycle, independent of the domain. Further, for each of the steps we managed to draw a conclusion on the specific and general practices which could become part of a new general Linked Data methodology. The details of our analysis are presented in Chapter 3: 'Linked Data Methodologies' of the main PhD thesis manuscript and are also briefly described in the chapter 'Content of the Thesis' in this executive summary.

Based on these two approaches, we propose and develop a new Linked Data methodology, focused on the principle of process reuse. More specifically, our methodology focuses on the reuse of the life-cycle steps, for a given domain. It includes steps which are aimed towards getting familiar with the domain in question, modeling and aligning the data, transforming the data into 5-star Linked Data, publishing the created datasets on the Web and defining use-cases or developing applications and services on top of the dataset. The guidelines from the methodology are aimed towards assisting data owners and publishers from a given domain in publishing their data in the same aligned, Linked Data format. Their data, once transformed into Linked Data and interlinked with other data already published using the same reusable components, could be used in new user-centric and analytical application and services.

As a validation of the proposed methodological guidelines, we apply them in the drug and healthcare domain. We apply the methodology within an automated system which gathers drug data from the official national drug registries of twenty-three different countries, executes data cleaning, aligns and transforms the data into 5-star Linked Data and publishes them on the Web in a common, aligned and consolidated Linked Drug Data dataset. Based on the guidelines, we develop reusable components from the life-cycle: a common schema, a data template, a transformer, a SPARQL-based tool for extending and interlinking the dataset and a web-based tool for transforming, interlinking and publishing the data. We then demonstrate a set of user-centric and analytical use-case scenarios over the generated dataset, which are otherwise unavailable or very

time-consuming in a scenario where a user works with the data available on the Web in HTML webpages.

With this, we show that a methodology which provides guidelines for developing reusable components for the Linked Data life-cycle allows data publishers to share their expertise in a given domain, whilst lowering the boundary for future Linked Data publishers to develop and publish new datasets on the Web in the same domain. The methodology allows data owners and data publishers which are not highly proficient in the domain of Linked Data to gain access to tools, services and other components which can be reused in their domain of interest, in order to generate and publish an aligned Linked Data dataset. On the other hand, the methodology provides proficient Linked Data publishers with means of fast and easy access to domains which are not their specialty, via the same reusable life-cycle components.

We believe that this combined benefit for different stakeholders in the Linked Data domain will eventually lead to a larger number of high-quality, aligned Linked Data datasets published as part of the LOD Cloud, and thus lead to better applications, services and analytics which rely on the structured data available on the Web. With the recent emergence of the data-driven scientific field of Data Science, creating and publishing high-quality structured data is becoming even more important. It enables data scientists to make data analytics over this cleaned, structured and aligned data in order to produce new knowledge and introduce new value within a given domain. The high level of quality of the published Linked Data datasets would thus lead to yielding better analytical results.

Bibliography

[1] ATC Codes: Structure and Principles. http://www.whocc.no/atc/structure_and_principles. Accessed: 2016-01-22.

[2] BatchRefine. https://github.com/fusepoolP3/p3-batchrefine. Accessed: 2016-03-23.

[3] Best Practices for Publishing Linked Data. http://www.w3.org/TR/ld-bp/. Accessed: 2016-01-22.

[4] D2R Server: Accessing databases with SPARQL and as Linked Data. http://d2rq.org/d2r-server. Accessed: 2016-01-22.

[5] Datahub Portal. http://datahub.io/. Accessed: 2016-01-22.

[6] DERI Vocabularies. http://vocab.deri.ie/. Accessed: 2016-01-22.

[7] Fusepool P3 BatchRefine Transformer. https://fusepoolp3.github.io/batch-refine/. Accessed: 2016-03-23.

[8] Global Open Drug Data (GODD). http://godd.finki.ukim.mk/. Accessed: 2016-07-20.

[9] Health and Lifesciences Extension of the Schema.org Vocabulary. http://health-lifesci.schema.org/. Accessed 10 October 2016.

[10] Introducing Schema.org: A Collaboration on Structured Data. http://www.ysearchblog.com/2011/06/02/introducing-schema-org-a-collaboration-on-structured-data/. Accessed: 2016-01-22.

[11] Introducing Schema.org: Bing, Google and Yahoo Unite to Build the Web of Objects. https://blogs.bing.com/search/2011/06/02/introducing-schema-org-bing-google-and-yahoo-unite-to-build-the-web-of-objects. Accessed: 2016-01-22.

[12] Introducing Schema.org: Search Engines Come Together for a Richer Web. https://googleblog.blogspot.mk/2011/06/introducing-schemaorg-search-engines.html. Accessed: 2016-01-22.

[13] ISO 3166-1 alpha-3 Standard. https://en.wikipedia.org/wiki/ISO_3166-1_alpha-3. Accessed: 2016-03-23.

[14] ISO 4217 Standard. https://en.wikipedia.org/wiki/ISO_4217. Accessed: 2016-03-23.

[15] Linked Open Data (LOD) Cloud. http://lod-cloud.net/. Accessed: 2016-01-22.

[16] Linked Open Data (LOD) Cloud cache instance. http://lod.openlinksw.com/. Accessed: 2016-01-22.

[17] Linked Open Data (LOD) Cloud: How To Join. http://lod-cloud.net/#how-to-join. Accessed: 2016-01-22.

[18] Linked Open Vocabularies (LOV). http://lov.okfn.org/. Accessed: 2016-01-22.

[19] LOD2 Project. http://lod2.eu/. Accessed: 2016-03-23.

[20] LODRefine. https://github.com/sparkica/LODRefine/. Accessed: 2016-01-22.

[21] Medical and Healthcare Related Terms of the Schema.org Vocabulary. http://schema.org/docs/meddocs.html. Accessed 10 October 2016.

[22] OpenRefine. http://openrefine.org/. Accessed: 2016-01-22.

[23] Permanent URI of the LinkedDrugs Dataset. http://linkeddata.finki.ukim.mk/lod/data/drugs#. Accessed: 2016-04-12.

[24] Schema.org Vocabulary. http://schema.org/. Accessed: 2016-01-22.

[25] Schema.org Vocabulary Releases. http://schema.org/docs/releases.html. Accessed 10 October 2016.

[26] Seminant: SPARQL Execution and Sharing. http://seminant.com/. Accessed: 2016-03-23.

[27] Silk Framework. http://silkframework.org/. Accessed: 2016-01-22.

[28] SPARQL Endpoint at the Faculty of Computer Science and Engineering in Skopje. http://linkeddata.finki.ukim.mk/sparql. Accessed: 2016-01-22.

[29] The ATC Classification Ontology. http://bioportal.bioontology.org/ontologies/ATC. Accessed 10 October 2016.

[30] The 'Drug' class, from the Schema.org Vocabulary. http://schema.org/Drug. Accessed: 2016-01-22.

[31] The LinkedDrugs Dataset on Datahub. https://datahub.io/dataset/linked-drugs. Accessed: 2016-04-17.

[32] The LinkedDrugs Project on GitHub. https://github.com/etnc/linked-drugs. Accessed: 2016-04-12.

[33] The LinkedDrugs Project Website. http://drugs.linkeddata.finki.ukim.mk/. Accessed: 2016-04-12.

[34] Virtuoso Instance at the Faculty of Computer Science and Engineering in Skopje. http://linkeddata.finki.ukim.mk. Accessed: 2016-01-22.

[35] Virtuoso Universal Server. http://virtuoso.openlinksw.com/. Accessed: 2016-01-22.

[36] W3C: Hash vs. Slash. https://www.w3.org/wiki/HashVsSlash. Accessed: 2016-01-22.

[37] W3C Healthcare Schema Vocabulary Community Group. http://www.w3.org/community/schemed/. Accessed 10 October 2016.

[38] Keith Alexander and Michael Hausenblas. Describing Linked Datasets - on the Design and Usage of VoID, the Vocabulary of Interlinked Datasets. In *Linked Data on the Web Workshop (LDOW 09), in conjunction with 18th International World Wide Web Conference (WWW 09)*. Citeseer, 2009.

[39] Aleksandar Andreevski, Riste Stojanov, Milos Jovanovik, and Dimitar Trajanov. Semantic Web Integration with SPARQL Autocomplete. In *Proceedings of the 12th Conference for Informatics and Information Technology*, pages 131–134. Faculty of Computer Science and Engineering, Skopje, 2015.

[40] Grigoris Antoniou and Frank Van Harmelen. *A Semantic Web Primer*. MIT Press, 2004.

[41] Sören Auer, Lorenz Bühmann, Christian Dirschl, Orri Erling, Michael Hausenblas, Robert Isele, Jens Lehmann, Michael Martin, Pablo N Mendes, Bert Van Nuffelen, et al. Managing the Life-Cycle of Linked Data with the LOD2 Stack. In *The Semantic Web–ISWC 2012*, pages 1–16. Springer, 2012.

[42] Tim Berners-Lee. 5-star Open Data. `http://5stardata.info/`.

[43] Tim Berners-Lee. Linked Data. `https://www.w3.org/DesignIssues/LinkedData.html`, June 2009. Accessed: 2016-01-22.

[44] Tim Berners-Lee, James Hendler, Ora Lassila, et al. The Semantic Web. *Scientific American*, 284(5):28–37, 2001.

[45] Tim Berners-Lee and Nigel Shadbolt. There's Gold to Be Mined from All Our Data. *The Times, London*, 2011.

[46] Ted J Biggerstaff. The Library Scaling Problem and the Limits of Concrete Component Reuse. In *Software Reuse: Advances in Software Reusability, 1994. Proceedings., Third International Conference on*, pages 102–109. IEEE, 1994.

[47] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked Data - the Story so Far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22, 2009.

[48] Christian Bizer, Tom Heath, Kingsley Idehen, and Tim Berners-Lee. Linked Data on the Web. In *Proceedings of the 17th International Conference on World Wide Web*, pages 1265–1266. ACM, 2008.

[49] Souripriya Das, Seema Sundara, and Richard Cyganiak. R2RML: RDB to RDF Mapping Language. `https://www.w3.org/TR/r2rml/`. Accessed: 2016-03-23.

[50] John E Gaffney Jr and RD Cruickshank. A General Economics Model of Software Reuse. In *Proceedings of the 14th International Conference on Software Engineering*, pages 327–337. ACM, 1992.

[51] RV Guha, Dan Brickley, and Steve Macbeth. Schema.org: Evolution of Structured Data on the Web. *Communications of the ACM*, 59(2):44–51, 2016.

[52] Michael Hausenblas. Linked Data Life Cycles. `http://www.slideshare.net/mediasemanticweb/linked-data-life-cycles`.

[53] Tom Heath and Christian Bizer. Linked Data: Evolving the Web into a Global Data Space. *Synthesis Lectures on the Semantic Web: Theory and Technology*, 1(1):1–136, 2011.

[54] Robert Hoehndorf, Dietrich Rebholz-Schuhmann, Melissa Haendel, and Robert Stevens. Thematic Series on Biomedical Ontologies in JBMS: Challenges and New Directions. *Journal of Biomedical Semantics*, 5:15, 2014.

[55] Bernadette Hyland and David Wood. The Joy of Data - a Cookbook for Publishing Linked Government Data on the Web. In *Linking Government Data*, pages 3–26. Springer, 2011.

[56] Martina Janevska, Milos Jovanovik, and Dimitar Trajanov. HTML5 based Facet Browser for SPARQL Endpoints. In *Proceedings of the 11th Conference for Informatics and Information Technology*, pages 149–154. Faculty of Computer Science and Engineering, Skopje, 2014.

[57] Milos Jovanovik, Aleksandra Bogojeska, Dimitar Trajanov, and Ljupco Kocarev. Inferring Cuisine-Drug Interactions Using the Linked Data Approach. *Scientific Reports*, 5, 2015.

[58] Milos Jovanovik, Marjan Georgiev, and Dimitar Trajanov. Towards Consolidating Brand-Name Drug Data Using Linked Data: The Case Study of the Macedonian Drug Bureau. Submitted for publication.

[59] Milos Jovanovik, Bojan Najdenov, Gjorgji Strezoski, and Dimitar Trajanov. Linked Open Data for Medical Institutions and Drug Availability Lists in Macedonia. In *New Trends in Database and Information Systems II*, Advances in Intelligent Systems and Computing, pages 245–256. Springer International Publishing, 2015.

[60] Milos Jovanovik, Bojan Najdenov, and Dimitar Trajanov. Linked Open Drug Data from the Health Insurance Fund of Macedonia. In *Proceedings of the 10th International Conference for Informatics and Information Technology*, pages 56–61. Faculty of Computer Science and Engineering, Skopje, 2013.

[61] Milos Jovanovik, Matej Petrov, Bojan Najdenov, and Dimitar Trajanov. Linked Music Data from Global Music Charts. In *Proceedings of the 10th International Conference on Semantic Systems (SEMANTiCS 2014)*, pages 108–115. ACM, 2014.

[62] Milos Jovanovik, Petar Ristoski, and Dimitar Trajanov. A System for Suggestion and Execution of Semantically Annotated Actions Based on Service Composition. In *ICT Innovations 2013*, Advances in Intelligent Systems and Computing, pages 97–109. Springer International Publishing, 2014.

[63] Aleksandar Kareski, Milos Jovanovik, and Dimitar Trajanov. Desktop Gateway: Semantic Desktop Integration with Cloud Services. In *Proceedings of the Sixth Balkan Conference in Informatics*, pages 162–168, 2013.

[64] Eduard Klein, Stephan Haller, Adrian Gschwend, and Milos Jovanovik. Sustainable Linked Open Data Creation: An Experience Report. In *Electronic Government and Electronic Participation*, volume 23 of *Innovation and the Public Sector*, pages 99–110. IOS Press, 2016.

[65] Dimitris Kontokostas, Patrick Westphal, Sören Auer, Sebastian Hellmann, Jens Lehmann, Roland Cornelissen, and Amrapali Zaveri. Test-Driven Evaluation of Linked Data Quality. In *Proceedings of the 23rd international conference on World Wide Web*, pages 747–758. International World Wide Web Conferences Steering Committee, 2014.

[66] Martin Kostovski, Milos Jovanovik, and Dimitar Trajanov. Open Data Portal based on Semantic Web Technologies. In *Proceeding from the 7th South East European Doctoral Student Conference*, 2012.

[67] Charles W Krueger. Software Reuse. *ACM Computing Surveys (CSUR)*, 24(2):131–183, 1992.

[68] Vivek Kundra. *Digital Fuel of the 21st Century: Innovation through Open Data and the Network Effect*. Joan Shorenstein Center on the Press, Politics and Public Policy, 2012.

[69] Josip Maras, Maja Štula, and Ivica Crnković. Towards Specifying Pragmatic Software Reuse. In *Proceedings of the 2015 European Conference on Software Architecture Workshops*, page 54. ACM, 2015.

[70] M Douglas McIlroy, JM Buxton, Peter Naur, and Brian Randell. Mass-Produced Software Components. In *Proceedings of the 1st International Conference on Software Engineering, Garmisch Pattenkirchen, Germany*, pages 88–98. sn, 1968.

[71] Robert Meusel, Christian Bizer, and Heiko Paulheim. A Web-Scale Study of the Adoption and Evolution of the Schema.org Vocabulary over Time. In *Proceedings of the 5th International Conference on Web Intelligence, Mining and Semantics*, page 15. ACM, 2015.

[72] Kostadin Mishev, Angjel Kjosevski, Nikola Kalemdzhievski, Nikola Koteli, Milos Jovanovik, Kosta Mitreski, and Dimitar Trajanov. Publishing Skopje Air Quality Data as Linked Data. In *Proceedings of the 12th Conference for Informatics and Information Technology*, pages 273–277. Faculty of Computer Science and Engineering, Skopje, 2015.

[73] Elena Mishevska, Bojan Najdenov, Milos Jovanovik, and Dimitar Trajanov. Open Public Transport Data in Macedonia. In *Proceedings of the 11th Conference for Informatics and Information Technology*, pages 161–166. Faculty of Computer Science and Engineering, Skopje, 2014.

[74] Martin Mitrevski, Milos Jovanovik, Riste Stojanov, and Dimitar Trajanov. Open University Data. In *Proceedings of the 9th Conference for Informatics and Information Technology*, pages 9–13. Faculty of Computer Science and Engineering, Skopje, 2012.

[75] Bojan Najdenov, Milos Jovanovik, and Dimitar Trajanov. VEO: an Ontology for CO_2 Emissions from Vehicles. *ICT Innovations 2014, Web Proceedings*, pages 269–278, 2014.

[76] Bojan Najdenov, Hristijan Pejchinoski, Kristina Cieva, Milos Jovanovik, and Dimitar Trajanov. Open Financial Data from the Macedonian Stock Exchange. In *ICT Innovations 2014*, pages 115–124. Springer International Publishing, 2015.

[77] Bojan Najdenov, Goran Petkovski, Milos Jovanovik, Riste Stojanov, and Dimitar Trajanov. Automated Linked Data Generation from the Transport Administration Domain. In *23rd Telecommunications Forum (TELFOR), 2015*, pages 827–830, 2015.

[78] Eric Prud'hommeaux, Carlos Buil-Aranda, Andy Seaborne, Axel Polleres, Lee Feigenbaum, and Gregory Todd Williams. SPARQL 1.1 Federated Query. `https://www.w3.org/TR/sparql11-federated-query/`, March 2013. Accessed: 2016-01-22.

[79] José Luis Redondo-García, Vicente Botón-Fernández, and Adolfo Lozano-Tello. Linked Data Methodologies for Managing Information about Television Content. *International Journal of Interactive Multimedia and Artificial Intelligence*, 1(6), 2012.

[80] Anisa Rula and Amrapali Zaveri. Methodology for Assessment of Linked Data Quality. In *Proceedings of the 1st Workshop on Linked Data Quality, co-located with 10th International Conference on Semantic Systems (SEMANTiCS 2014)*, 2014.

[81] François Scharffe, Ondřej Zamazal, and Dieter Fensel. Ontology Alignment Design Patterns. *Knowledge and Information Systems*, 40(1):1–28, 2014.

[82] Max Schmachtenberg, Christian Bizer, and Heiko Paulheim. Adoption of the Linked Data Best Practices in Different Topical Domains. In *The Semantic Web–ISWC 2014*, pages 245–260. Springer, 2014.

[83] William Smith, Alan Chappell, and Courtney Corley. Medical and Transmission Vector Vocabulary Alignment with Schema.org. Technical report, Pacific Northwest National Laboratory (PNNL), Richland, WA (US), 2015.

[84] Mirko Spasic, Milos Jovanovik, and Arnau Prat-Prez. An RDF Dataset Generator for the Social Network Benchmark with Real-World Coherence. In *Proceedings of the Workshop on Benchmarking Linked Data (BLINK), 15th International Semantic Web Conference (ISWC)*, pages 18–25, 2016.

[85] Riste Stojanov, Marjan Georgiev, Vladimir Zdraveski, Milos Jovanovik, and Dimitar Trajanov. Live Objects - Collaborative Window in the Corporate Documents. In *New Trends in Database and Information Systems II*, Advances in Intelligent Systems and Computing, pages 71–81. Springer International Publishing, 2015.

[86] Damjan Temelkovski, Milos Jovanovik, Igor Mishkovski, and Dimitar Trajanov. Towards Open Data in Macedonia: Crime Map Based on the Ministry of Internal Affairs' Bulletins. In *Proceeding from the 9th Conference for Informatics and Information Technology*, pages 14–18. Faculty of Computer Science and Engineering, Skopje, 2012.

[87] Will Tracz. Where Does Reuse Start? *ACM SIGSOFT Software Engineering Notes*, 15(2):42–46, 1990.

[88] Dimitar Trajanov, Riste Stojanov, Milos Jovanovik, Vladimir Zdraveski, Petar Ristoski, Marjan Georgiev, and Sonja Filiposka. Semantic Sky: A Platform for Cloud Service Integration Based on Semantic Web Technologies. In *Proceedings of the 8th International Conference on Semantic Systems (I-SEMANTICS 2012)*, pages 109–116. ACM, 2012.

[89] Marc Twagirumukiza. Schema.org New Release 3.0 with the health-lifesci.schema.org Extension. `https://googleblog.blogspot.mk/2011/06/introducing-schemaorg-search-engines.html`, May 2016. Accessed 10 October 2016.

[90] Boris Villazón-Terrazas, Luis M Vilches-Blázquez, Oscar Corcho, and Asunción Gómez-Pérez. Methodological Guidelines for Publishing Government Linked Data. In *Linking Government Data*, pages 27–49. Springer, 2011.

[91] Dydimus Zengenene, Vittore Casarosa, and Carlo Meghini. Towards a Methodology for Publishing Library Linked Data. In *Bridging Between Cultural Heritage Institutions*, pages 81–92. Springer, 2014.