AN ONTOLOGY-BASED DATA WAREHOUSE FOR THE GRAIN TRADE DOMAIN

Mhamed Itmi[1] and Boulares Ouchenne[2]

LITIS Laboratory, INSA Rouen Normandy,
University of Rouen Normandy. France

ABSTRACT

Data warehouse systems provide a great way to centralize and converge all data of an organization in order to facilitating access to the huge amounts of information, analysing and decision making. Actually, the conceptual data-models of data warehouses does not take into account the semantic dimension of information. However, the semantic of data models constitute an important indicator to help users to finds its way in any applications that use the data warehouse. In this study, we will tackle this problem trough using ontologies and semantic web techniques to integrate and model information. The contributions of this paper are an ontology for the field of grain trade and a semantic data warehouse which uses the ontology as a conceptual data-model.

KEYWORDS

Ontology Engineering, Data Warehouse, Sparql Endpoint.

1. INTRODUCTION

HAROPA Port of Rouen has a leading position in France with around 50% market share of wheat exported by sea and 44% on barley. According to the highly dependent campaigns of the production and the meteorological conditions, the grain traffic represents in tonnage between 25% and 30% of the traffics of Rouen.

HAROPA port of Rouen benefits from the proximity of a market of 22 million consumers in a radius of 200kms. The expertise in transport, handling and logistics of its operators, combined with its geographical location, explain its strategic interest for all types of goods. It is mainly known for the export of cereals.

With the aim of consolidating HAROPA's leadership as a major European port for the export of cereals and to strengthen its competitiveness and to capture new markets for the actors of the cereals sector of the Seine axis, we have participated in a study which led us to work on an ontology dealing with the grain trade.

Nowadays, organization's data sources are scattered across multiple systems, not necessarily compatible. These data sources are designed to be effective for the functions on which they are specialized. They are of- ten unstructured for analysis and designed with the primary objective of preserving information. As critical business information has to be served with a fast response time and well structured for decision making. The data warehouse aims to aggregate and enhance data from different sources to allow the user to get access easily, quickly and ergonomically to the

David C. Wyld et al. (Eds) : AIAPP, CRIS, CoSIT, SIGL, SEC - 2018
pp. 93–101, 2018. © CS & IT-CSCP 2018 DOI : 10.5121/csit.2018.80408

information. The process of implementing a data warehouse is a very complex task that pushes designers to acquire wide knowledge of the domain, thus requiring a high level of expertise. The design of the conceptual model is the key step of the process of designing data warehouses. This model is the basis for the implementation of the data warehouse. The conceptual model of a data warehouses is generally [1,2,3] represented by a standard 3NF data model, star models, snowflake model, or constellation model. These models prescribes the information to be represented in a database stored on a physical medium. These data models are only powerful at the structural level and lack the ability to specify the semantic relations contained in complex data for modelling and analysis. To tackle this problem, we will use ontologies to facilitate the integration of heterogeneous data sources by resolving semantic heterogeneity between them [4,5,6]. The main advantages of using ontologies is to define the semantic vocabulary of data and to obtain implicit knowledge thanks to performing reasoning on it.

The remaining part of the paper is structured as follows: section 2 introduces the industrial case study including and describes the engineering process performed to develop our ontology. Section 3 presents some of the content of our ontology. Section 4 describes the architecture of our proposed model for the data warehouse, and summarizes implementation we have conducted. Section 5 summarizes the work and draws conclusions.

2. THE CONTEXT OF THE STUDY AND THE GENERAL APPROACH

The case study presented in this paper is based on an industrial research and development project. The goals of this study is to design a reference ontology for representing information on the grain trade activity at the port of Rouen. This ontology will serve as a data model for a data warehouse containing information collected as part of an R&D project. There are many reasons for us to adopt the choice of implementing a data warehouse:

1. Centralize and converge all types of data collected from several for- mats (reports, databases, Excel files, etc.) to semantic data interlinked in order is facilitate access to information, analysis and decision making.

2. Allows a balanced perspective of the organizations. Indeed, insofar as each relevant indicator (for example juridical) is directly or indirectly correlated to another (for example economic), it affects directly or indirectly its objectives.

3. Saves time and money. In fact, users can quickly access data from a huge number of sources (all in one place). So, they can quickly make the best decisions.

4. Improves the quality and consistency of data. As its implementation includes the conversion of data from many sources in a common format. So we can have more confidence in the accuracy of such data.

5. Provides historical information. Indeed, it can store large amounts of historical data so we can analyse the different periods and temporal trends in order to make predictions.

2.1. The General Approach

Figure 1 shows an overview of our approach. The first step in the design of a data model of the warehouse. Once the model is developed and validated, the second phase begins. It consists on the creation of the data warehouse as a dataset (RDF [7] database). The third and final step in the population of the data warehouse. It takes place in several steps and constitute the data migration

phase after they have undergone selection and reformatting operations in order to be homogenized.

This phase is an important step insofar as it is estimated at about 60 per cent of the implementation time of the warehouse. We have identified several types of data sources to populate the warehouse. For each type of data we have implemented integrators and, for the interaction between users and the data warehouse, we developed web interfaces for navigation and update.

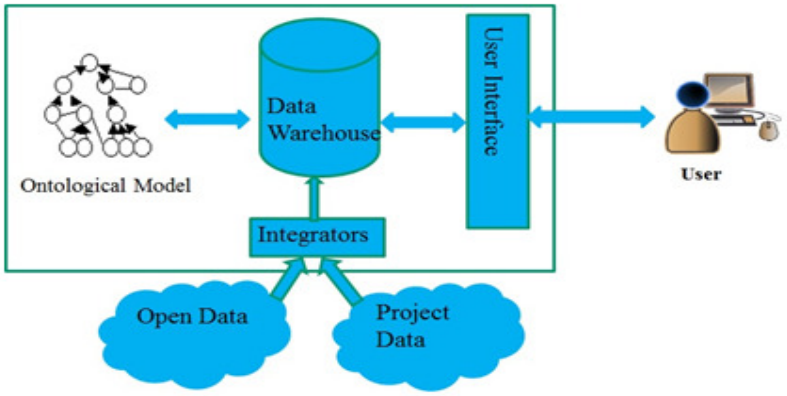

Figure 1. Overview of the approach

3. OVERVIEW OF THE ONTOLOGICAL MODEL

This starting point of the process of the ontology development is to define its domain and scope. In our case, the domain and scope of the ontology is the grain trade activity at the port of Rouen. The process for developing our ontology follows an iterative and incremental approach [8,9]. We have followed the following steps:

Step 1: Creating a conceptual data model that identifies and structure the basic concepts implemented by our ontology. The task of Conceptual Modelling plays a crucial role in the process of our ontology development. Conceptual models translate and specify the main data requirements in an abstract representation about our domain.

Step 2: Identifying pre-existing RDF or OWL schemas which propose classes and properties equivalent to those identified in the previous step to expand and/or refine them. Reusing existing ontologies, may even be a requirement if our data warehouse needs to interact with other applications already use specific ontologies or controlled vocabularies.

Step 3: Enriching the ontology by adding features to certain relation-ships, such as the fact that a property is transitive, reflexive, symmetric, etc.

During the design of this ontology, we have strived apply the commonly recommended techniques in the Community [10]. For instance, the definition for each object property a reverse property thus facilitating the manipulations and aligning our ontology with other reference ontologies (such as the FOAF Ontology [11])). Also, we opted for reusing only terms we need from external ontologies, without importing them explicitly. Finally, our ontology has been described in OWL2, we have taken advantage of possibilities of this language in terms of expressiveness [12]. In particular, we defined chains of properties to infer new relationships without recourse to a language dedicated to the expression rules.

3.1. Overview of the Ontological Model

The model we have designed includes all information relating to the activity of grain trading at the port of Rouen. It represents graphically the entities, the various actors and contractual relations between them (Sale, Rent, Fobbing, etc.). In the remainder of this article, each part of our ontology is presented as an UML class diagram where:

- UML classes are OWL classes.
- UML class attributes represent OWL data properties (data types have been added to not to complicate the presentation).
- Association relationships between OWL classes represent OWL object properties.
- Each identifier is prefixed by a domain name. Only those prefixed by (realgrain) are actually introduced in our ontology and he others are reused from other ontologies.

To be readable, our ontological model is divided into three parts: the actors, the different types of contracts and the various kinds of sale contracts.

3.1.1. Representation of Actors

This section describes the different information describing people, organizations and infrastructure involved in the export of grain field, the nature of that involvement and the semantic relationships between them. Figure 2 shows the class diagram of this first part of the ontology. For instance, considering the example of a Silo which is a structure used to store the grain before its shipments to final clients. The Silo class defined in our ontology is a subclass of

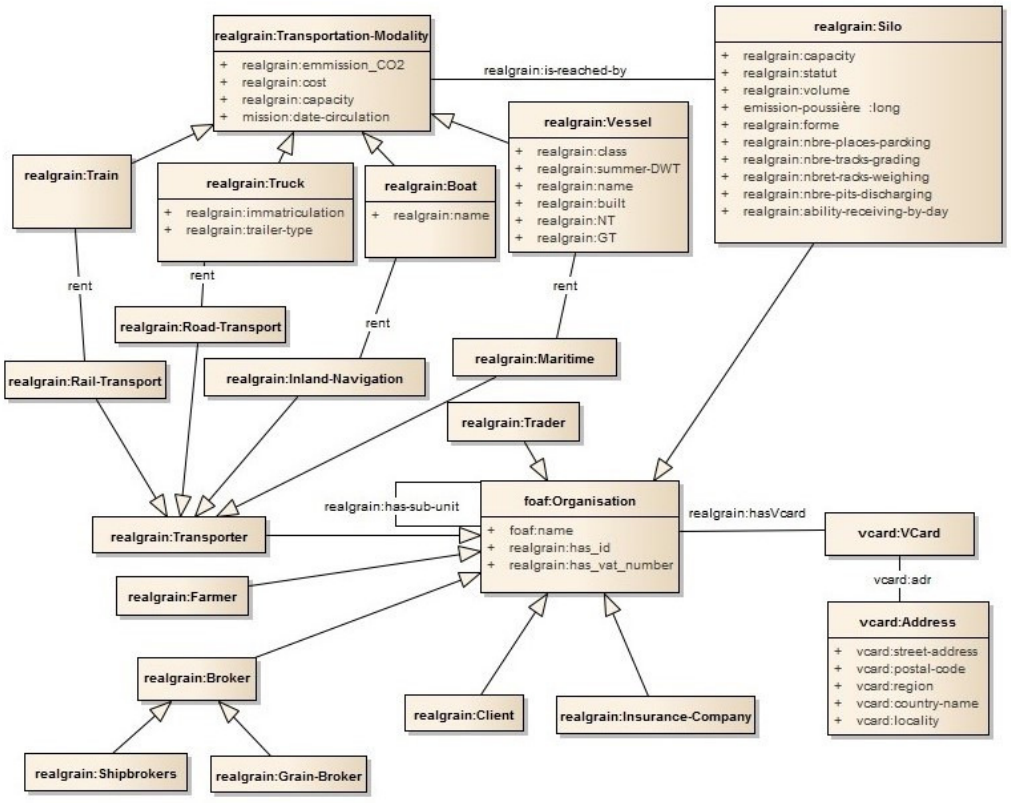

Figure 2. Actors

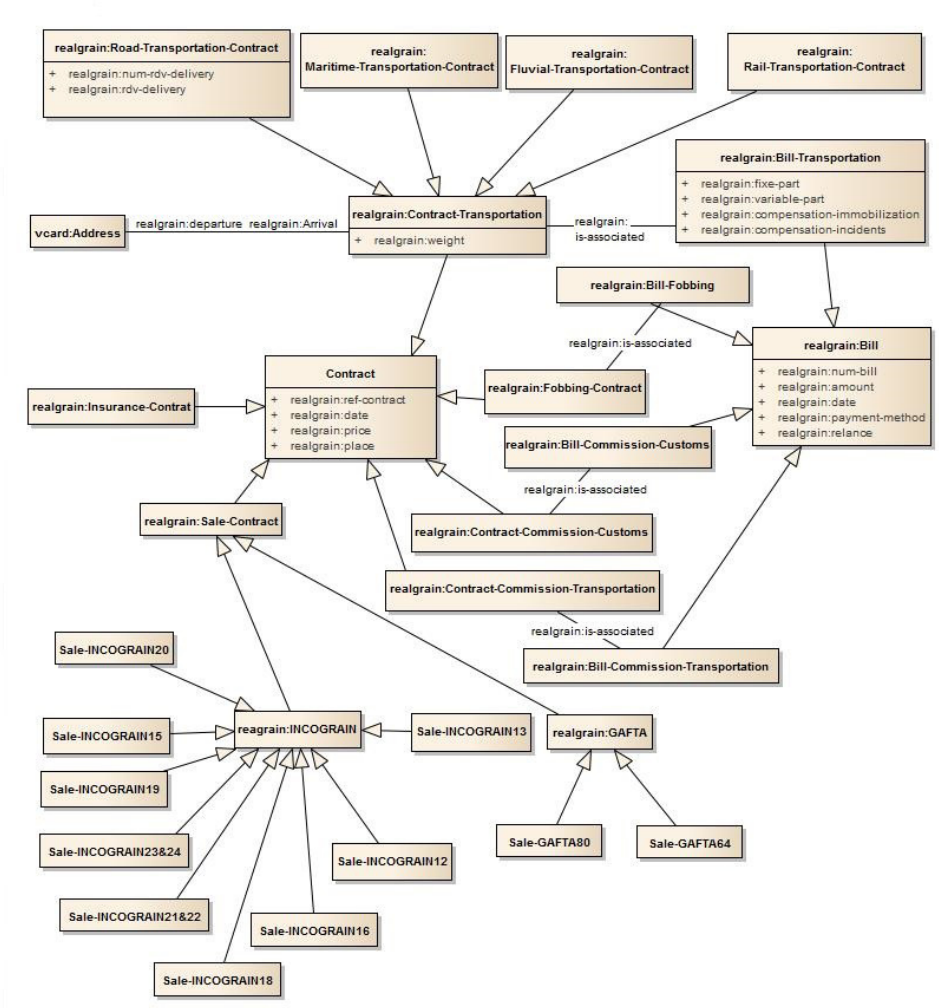

Figure 3. Contracts

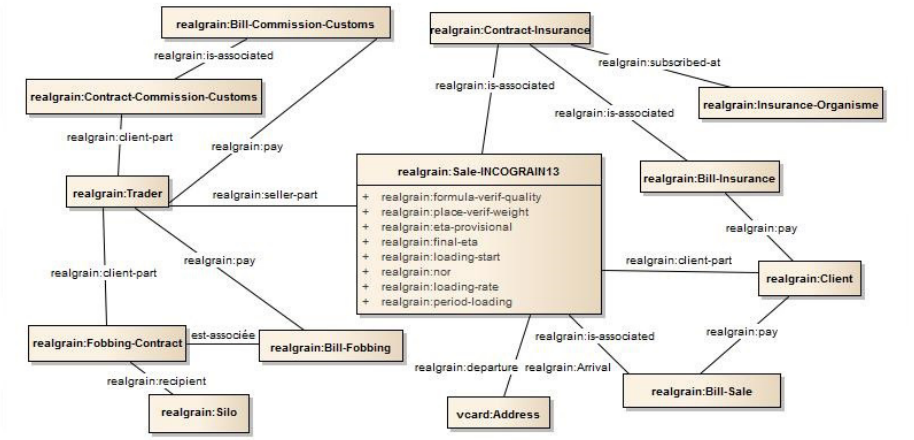

Figure 4. Example of an INCOGRAIN Contract

the foaf:Organization class defined in the FOAF ontology [11]. We have introduced and reused some data properties describe relationships between instances (individuals) and data values. We have also introduced and reused some Object properties to describe relationships between two instances (individuals). They link individuals from a domain to individuals a range. For example:

- foaf:name: is a data property reused from the FOAF ontology to represent the name of the organization.
- realgrain:capacity: is data property introduced to represent storage capacity of the Silo.
- vcard:adr: is an object property reused from the Vcard ontology to link the postal address to an information contact.
- realgrain:is-reached-by is an object property introduced to represent the different kinds of transportation modalities to reach the Silo.

3.1.2. Contracts

This section describes the different contracts, the information which contains and semantic relationships between them. Figure 3 shows the class diagram of the second part of ontology, and the hierarchical relationships between the different classes. Considering the example of sale contracts. For export sales of French cereals, buyers and sellers use standard contracts of purchase and sale. There are mainly two kinds of contracts GAFTA and INCOGRAIN which facilitate transactions and reduce sources of disputes between buyers and sellers. The choice of one of them determines the place of arbitration (Paris and London), the language of the proceedings (French or English) and applicable law (French or Anglo-Saxon). For The INCOGRAIN, there are twelve contract type available in several languages. Each contract is identified by a number which determine the client, the seller, the mode of transportation used for bringing the grain, etc.

3.1.3. Sales Contracts

For international sales of cereals from the port of Rouen, there are two agreement models of contracts CAF and FOB. The main difference between an FOB and a CAF agreement is the point at which responsibility and liability transfer from seller to buyer. With a FOB shipment, this occurs when the shipment reaches the port or other facility designated as the point of origin. With a CAF agreement, the seller pays costs and assumes liability until the grain reach the port of destination chosen by the buyer. Figure 4 shows the class diagram of the standard contract INCOGRAIN-13 and semantic relationships between different actors.

4. SYSTEM ARCHITECTURE AND IMPLEMENTATION

Figure 5 shows the general architecture of the system. It includes four basic elements:

1. An RDF database (triple store), for RDF triples data storage that relate objects among them through the SPARQL query language [13]. Today, there is a list of implementations that provide the RDF triple store functionalities, including Apache Jena, Virtuoso, Owlim, Neo4J, GraphDB, Opengraph, etc. Based on the results of qualitative and quantitative study of existing RDF stores [14], we opted for the use of Apache Jena TDB [15], which is an open source framework (based on Java) for creating semantic web oriented applications.

2. A SPARQL endpoint [16], which allows applications to query information from the triple store using the SPARQL query language. The solution adopted to implement the

SPARQL endpoint is the free software Jena Fuseki [17,6]. This solution offers external applications the possibility of exploiting our data triple store by questioning directly the deposit trough SPARQL queries or even to combine it with other triple stores. Fuseki also offers two ways to interact with the user from a web application via an HTTP or with application programming interface (API). Our SPARQL endpoint provides an intuitive interface to write the SPARQL query and to select the formats of the results (JSon, XML, CSV, etc).

3. A Java Web application deployed on a WildFly application server [18] (a set of Servlets and JSP pages) has been developed to provide a simple way to view modify and enrich the data warehouse. The web interface offers two levels of navigation (conceptual level: General view of the model, instance level: details of the instances).

4. A set of transformation tools (called integrators) allowing to automatically convert data (databases, Excel files, etc.) to RDF triples (complying with the ontology previously defined).

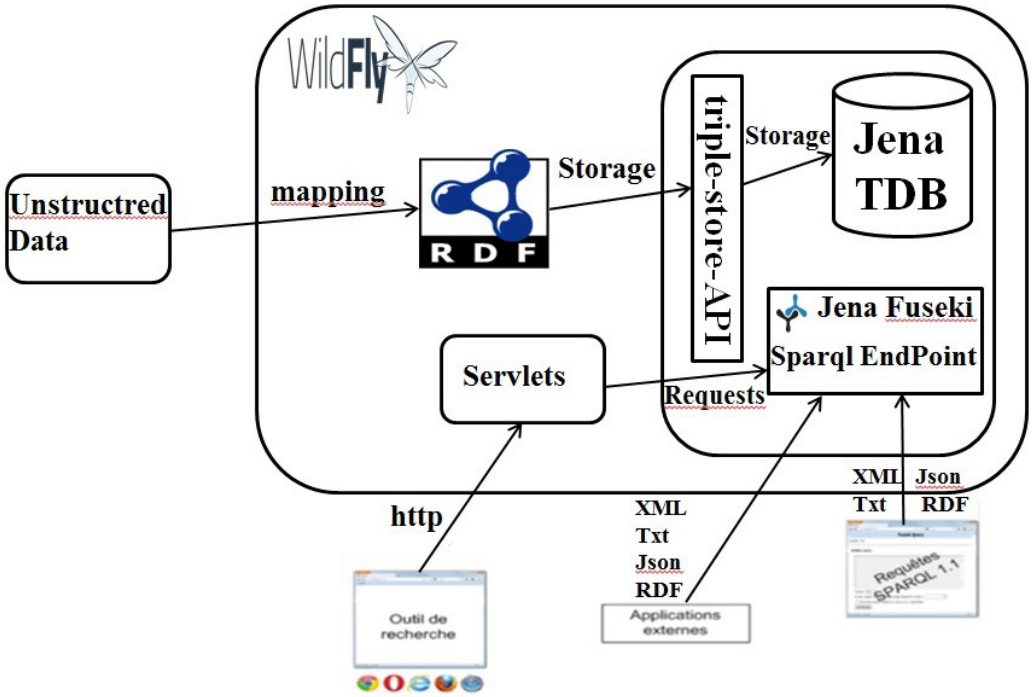

Figure 5. Detailed Architecture

5. CONCLUSIONS

This work has shown how the use of Semantic Web technologies could greatly improve the design of semantic data warehouses and facilitate the integration of data semantics by giving a formal semantics to data elements. Firstly, we have proposed a domain ontology which has given a semantic representation to all data related to the activity of grain trade. This ontology has been designed by applying the best practices proposed in [10]. Then, we have developed integrators enabling to automatically map data collected from several formats to RDF triples. Finally, a SPARQL endpoint has been proposed to provides access to the data warehouses. In the future, we plan to enrich our data warehouse with external sources available on open data platforms and to

take advantage of the power of reasoning engines from Web Semantic technologies to manage smartly the content.

ACKNOWLEDGEMENTS

This research is supported by the European Union (EU) with the European Regional Development Fund (ERDF) and Normandy Region.

REFERENCES

[1] Luca Cabibbo and Riccardo Torlone. A logical approach to multidimensional databases. In Proceedings of the 6th International Conference on Extending Database Technology: Advances in Database Technology, EDBT '98, pages 183– 197, London, UK, UK, 1998. Springer-Verlag.

[2] Aris Tsois, Nikos Karayannidis, and Timos Sellis. Mac: Conceptual data modelling for OLAP. In 3rd International Workshop on Design and Management of Data Warehouses (DMDW 2001, page 2001, 2001.

[3] Surajit Chaudhuri and Umeshwar Dayal. An overview of data warehousing and OLAP technology. SIGMOD Rec., 26(1):65–74, March 1997.

[4] Jesus Pardillo and Jose-Norberto Mazo´n. Using ontologies for the design of data warehouses. CoRR, abs/1106.0304, 2011.

[5] Khouri Selma, Boukhari Ilyès, Bellatreche Ladjel, Sardet Eric, Jean Stéphane and Baron Michael. Ontology-based structured web data warehouses for sustainable interoperability: Requirement modeling, design methodology and tool. Comput. Ind., 63(8):799–812, October 2012.

[6] Ouchenne, B. & Itmi, M. OntoEDIFACT: An Ontology for the UN/EDIFACT Standard DBKDA 2017 : The Ninth International Conference on Advances in Databases, Knowledge, and Data Applications, IARIA XPS Press, 2017, 91-96

[7] Resource Description Framework (RDF). (2016). https://www.w3.org/RDF/.

[8] Andre Menolli, H. Sofia Pinto, Sheila Reinehr, and Andreia Malucelli. An incremental and iterative process for ontology building.

[9] Mariano Fernandez, Asuncion Gomez-Perez, and Natalia Juristo. Methontology: from ontological art towards ontological engineering. In Proceedings of the AAAI97 Spring Symposium Series on Ontological Engineering, pages 33–40, Stanford, USA, March 1997.

[10] Jean Charlet, Bruno Bachimont, and Raphal Troncy. Ontologies pour le web smantique. Revue I3, page 31p, 2004.

[11] Dan Brickley and Libby Miller. FOAF Vocabulary Specification 0.99. http://xmlns.com/foaf/spec/.

[12] Christine Golbreich and Evan K. Wallace. OWL 2 Web Ontology Language New Features and Rationale. https://www.w3.org/TR/owl2-new-features/.

[13] Steve Harris and Andy Seaborne. SPARQL 1.1 Query Language. https://www. w3.org/TR/2013/REC-sparql11-query-20130321/.

[14] Bernhard Haslhofer, Elaheh Momeni Roochi, Bernhard Schandl, and Stefan Zander. European rdf store report. Technical report, University of Vienna, Vienna, March 2011.

[15] Apache Jena - TDB. https://jena.apache.org/documentation/tdb/.

[16] Lee Feigenbaum Kendall Grant Clark and Elias Torres. SPARQL Protocol for RDF. https://www.w3.org/TR/rdf-sparql-protocol/.

[17] Apache Jena Fuseki. https://jena.apache.org/documentation/fuseki2/.

[18] WildFly application server. http://www.wildfly.org/.