

# An Overview of the Linked Data AppStore

~ Demo/Poster Paper ~

Dumitru Roman, Claudia D. Pop, Roxana I. Roman, Bjørn M. Mathisen,  
Leendert Wienhofen, Brian Elvesæter, and Arne J. Berre

SINTEF, Oslo, Norway

Contact: [dumitru.roman@sintef.no](mailto:dumitru.roman@sintef.no)

**Abstract.** This demo/poster paper provides an overview of a Software-as-a-Service platform prototype for data integration on the Web – The Linked Data AppStore (LD-AppStore). It builds upon Linked Data technologies, targets data scientists/engineers and data integration application developers, and aims to provide a solution for simplifying tasks such as data transformation, querying, entity extraction, data visualization, crawling, etc. This paper focuses on the overall architecture of the LD-AppStore, basic data operations supported by the current prototype, and outlines the demonstration of the prototype.

## 1 Motivation

In recent years a significant amount of data has been made available as Open and/or Linked Data, however applications utilizing such data have been rather few.<sup>1</sup> Reasons include, amongst others, the technical complexity and economical cost of integration, publishing, interlinking and providing reliable access to the data, and lack of simplified and unified solutions for data consumption, and lack of tools and infrastructures where datasets and 3<sup>rd</sup> party components can be made easily available to application developers to reuse, combine and develop novel data-driven applications. At present, Linked Data publishers and application developers need to rely on generic platforms (like the Amazon Web Services or Google App Engine cloud providers), and build, deploy and maintain complex Linked Data software and data stacks from scratch. Tools addressing various aspects of data integration process, though available in a Linked Data context, are difficult to use for more complex, interesting data integration tasks. This results in a high cost of data integration at large scale, a rather complicated and time consuming process. New innovative ways of simplifying data integration in a Linked Data context are needed.

---

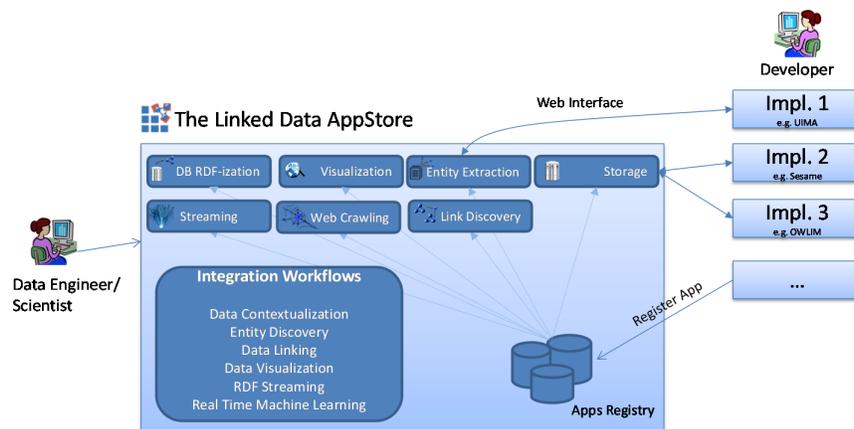
<sup>1</sup> As of Sept 2014, for example, the official EU public open data portal (<http://publicdata.eu/>) contains more than 48,000 datasets but lists less than 80 applications using the data. The situation is not much different for other open data portals (see e.g. <http://www.datacatalogs.org/>).

To simplify the data integration process, and support data publishers and application developers, this paper provides an overview of a Software-as-a-Service platform—*The Linked Data AppStore (LD-AppStore)*—for data scientists/engineers aiming to enable them to use, in a rather simplified manner, tools/services for tasks such as data transformation, entity extraction, data visualization, crawling, etc. At the same time, data integration application developers have the possibility of exploiting the use of their tools/services by plugging them into the LD-AppStore.

## 2 The LD-AppStore Platform Overview

The LD-AppStore is meant to be a service where data engineers can get access to various types of data operations, such as data transformation, storage, querying, linking, visualization, etc., which they can apply on their data, and have access to various tool/service implementations of those data operations – implementations provided by developers. The LD-AppStore serves as a registry of data operations and their implementations.

Figure 1 provides a high level overview of the LD-AppStore architecture.



**Figure 1. LD-AppStore Architecture Overview**

The upper part of the picture depicts components for basic data operations, currently being considered: RDF-ization of relational databases (mapping relational tables to RDF graphs), data visualization (visualization of RDF graphs), entity extraction (extracting entities from various sources), data storage (storage of RDF data manipulated in the platform), link discovery (finding links between data in RDF graphs), crawling (searching through RDF graphs), and data streaming (querying streams of RDF data). A set of Web APIs have been designed for these data operations. The set of tools/services that implement these basic data operations are made available through the registry functionality of the platform (lower right part of the figure). When using a specific data operation, the data engineer may select which implementation of that operation he/she wants to use. The Linked Data tool/service developers have access to the platform for

registering their implementations, i.e., the implementations of the Web APIs corresponding to the data operations APIs. The lower left part depicts a set of data integration workflows meant to seamlessly combine the basic data operations in workflows (configurable by the data engineers) that can eventually provide further useful insights into the data on which they are applied.

In the current design, the platform offers six different types of basic data operations for which Web APIs have been designed: *DB-RDFization* (for mapping data from relational databases to RDF); *Entity Extraction* (for extracting entities from various sources); *Data Visualization*; *Storage* (for storing/querying data); *Streaming* (for querying streams of data); *Link Discovery* (for discovering relations between different datasets); and *Web Crawling* (for searching Linked Data).

### 3 The LD-AppStore Prototype and Demonstration

The current implementation of the LD-AppStore that will be demonstrated consists of the backend infrastructure for registering applications/tools implementing the APIs of data operations, the graphical frontend infrastructure through which data engineers can access the various data operations and the tools/services that implement them, as well as a set of tools that have been modified to implement the above mentioned APIs. Figure 2 provides a screenshot of the LD-AppStore homepage.

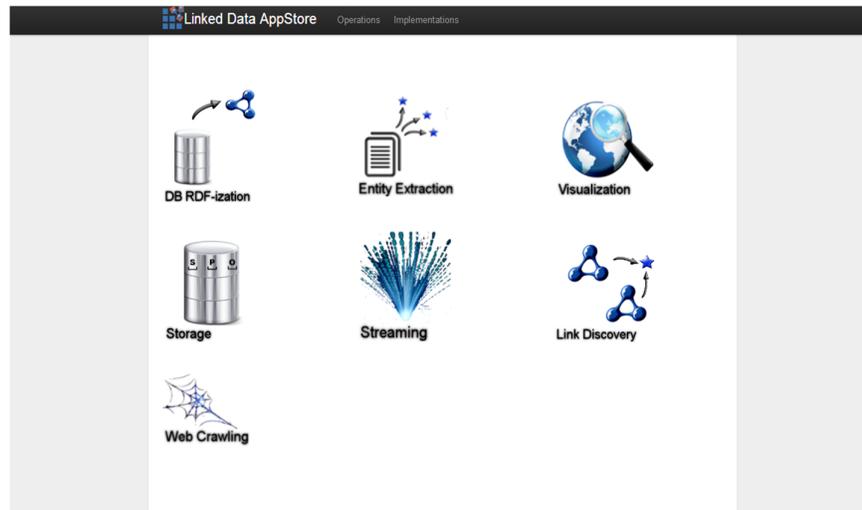


Figure 2. Screenshot of the LD-AppStore homepage.

The platform offers the possibility to register new tools/services as implementations for various operations. For each of the already registered tools a programmatic Web interface has been made which follows the one for its corresponding operation. In this way, implementation independence has been obtained, as long as each of the new added tools implement the operation's interface. The following tools have been integrated in

the current prototype: DB2Triples<sup>2</sup> for the DB-RDFization operation; The Unstructured Information Management Architecture (UIMA)<sup>3</sup> for the entity extraction operation; LodLive<sup>4</sup> for the visualization operation; OpenRDF Sesame<sup>5</sup> for storage operations; Continuous SPARQL (C-SPARQL)<sup>6</sup> for the streaming operation; The Silk framework<sup>7</sup> for the link discovery operation; and LDSpider<sup>8</sup> for the crawling operation.

The demonstration will show the current implementation focusing on overall the capabilities of the prototype and exemplify the registration and use of existing tools (e.g. DB2Triples) in the LD-AppStore.

**Related Approaches.** The LD-AppStore follows the research line of bundling well-established technologies and tools for publishing and consuming Linked Data in order to ease data integration on the Web. Notable approaches developed in this area include toolchains such as the Linked Data Stack<sup>9</sup> and the LarkC platform<sup>10</sup>. Such approaches do not provide an as-a-service hosted solution where 3<sup>rd</sup> party tool developers can plug-in their implementations for different data operations and where data publishers can configure and execute workflows of data operations implementations on their data --- which is what LD-AppStore targets. DaPaaS<sup>11</sup>, COMSODE<sup>12</sup>, and LinDA<sup>13</sup> are a number of recent EU funded research projects addressing the problem of simplifying access, integration, and usage of open data based on Linked Data technologies, primarily focusing on data publication and consumption aspects. The projects are in early stages of development with their approaches not entirely defined yet, however ideas from the LD-AppStore are finding traction in the DaPaaS project.

**Acknowledgements.** This work was partly funded by the following projects: BigFut (SINTEF internally funded project 102003299), DaPaaS (FP7 610988)<sup>14</sup>, SmartOpenData (FP7 603824)<sup>15</sup>, and InfraRisk (FP7 603960)<sup>16</sup>.

---

<sup>2</sup> <https://github.com/antidot/db2triples>

<sup>3</sup> <https://uima.apache.org/>

<sup>4</sup> <http://en.lodlive.it/>

<sup>5</sup> <http://www.openrdf.org/>

<sup>6</sup> <http://streamreasoning.org/download/>

<sup>7</sup> <http://wifo5-03.informatik.uni-mannheim.de/bizer/silk/>

<sup>8</sup> <https://code.google.com/p/ldspider/>

<sup>9</sup> <http://stack.linkeddata.org/>

<sup>10</sup> <http://www.larkc.eu/>

<sup>11</sup> <http://dapaas.eu/>

<sup>12</sup> <http://www.comsode.eu/>

<sup>13</sup> <http://linda-project.eu/>

<sup>14</sup> <http://project.dapaas.eu/>

<sup>15</sup> <http://www.smartopendata.eu/>

<sup>16</sup> <https://www.infrarisk-fp7.eu/>