RDF based Linked Open Data Management as a DaaS Platform

LODaaS (Linked Open Data as a Service)

Seonho Kim, Ivan Berlocher, Tony Lee
Saltlux, Inc.
Seoul, South Korea
e-mail: {shkim, ivan, tony}@saltlux.com

***Abstract*—In this paper we discuss the architecture and the processes for Resource Description Framework (RDF) based Linked Open Data as a Service (LODaaS), considering practical use cases. LODaaS is different from the usual Linked Data Platform (LDP) or Data Warehouse (DW), and as such, has to consider its own stakeholders and the processes of data publishing and consuming based on agreed ontology schema. The datasets should be transformed, published and consumed following the schema so that datasets could be shared, linked together and queried by 3rd party services without difficulty regarding data structures and queries. We implemented the Personalized and Localized Urban Quality Index (PLUQI) application for a data consumption use case, utilizing DaPaaS as a LODaaS platform. To implement this, s we designed an ontology schema, collected and published the datasets into the DaPaaS platform and reused these via the endpoint for PLUQI web service.**

Keywords—Linked Data Platform, RDF, Data as a Service, Data Integration, Open Data.

I. INTRODUCTION

W3C proposed the recommendation of Linked Data Platform (LDP) [1], which is a Linked Data specification defining a set of application integration patterns for building RESTful HTTP services that handle RDF documents. It provides a set of best practices and a simple approach for a read-write Linked Data architecture based on HTTP access to web resources that describe their state using the RDF data model [2].

Regarding the applications that use Linked Data, the data should be integrated for the domain specific purpose for which they aim to, but the recommendation does not cover this point. The graph-based RDF data model is flexible to integrate data from multiple data sources, but for data consumers who need to write a SPARQL query to retrieve data from the LDP, it is difficult to retrieve what they exactly want, because they do not have any preliminary information on what they should expect to get from it. Therefore, data consumers have to explore and look into the data first before writing queries. In addition, the data should be inter-connected, but composed of many namespaces that may have their own data schema for each, but not published on the LDP. Also, even if they are well constructed, the data integration itself has three kinds of heterogeneity problems to be solved: syntactic heterogeneity, structural heterogeneity, and semantic heterogeneity [3].

Generally, these problems are not handled or c resolved by most of the related research or solutions, because many of them are focusing on constructing a general knowledge base like Freebase, or have a specific purpose such as Linked Closed Data [4], Linked Government Data [5], or Linked Enterprise Data [6], so that they do not have to consider multiple purposes or requirements from the third party application developers or data consumers.

In this paper we propose the Linked Open Data as a Service (LODaaS) concept and its architecture, with the processes to manage the Linked Data for which the LDP could be used as Data as a Service (DaaS).

Section II describes the concepts and the considerations of the LODaaS, relevant stakeholders and core features. Section III describes the use case – PLUQI, which is a web-based service based upon the LODaaS. Section IV summarizes the roles and the features of LODaaS considered in this paper.

II. METHODOLOGY

A. Stakeholders

LODaaS has three relevant stakeholders: data owners, data publisher, and data consumers, as represented in Figure 1. The data owner is the one who owns the datasets to be published as open data, and the data publisher transforms the datasets and imports them into the LODaaS repositories. Both roles enable the reuse of data by the data consumer.

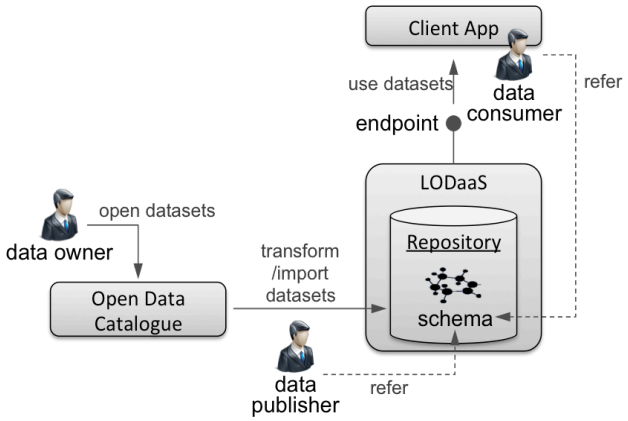

Figure 1. Relevant stakeholders of LODaaS

However, some of the open datasets are required to be transformed and imported according to the domain specific needs in the agreed format so that the data consumers can reuse them. This means that datasets should be transformed based on the agreed ontology schema so that the data consumers can simply query following the schema, rather than concern themselves with the actual datasets. So, the ontology schema should be designed accordingly, considering data consumer's perspectives and this is one of the data publisher's roles.

The data publisher should define the ontology schema to transform the datasets and import them to the repository. This means that the data publisher has to manage the repository, including the schema, to meet the needs of the services using the published data. Therefore, the schema should not be complex and not changed often in perspective of service management to provide the datasets via the endpoint, and to keep the processes of transformation/import efficiently.

B. Features

DaaS is an emerging subset of the "as a service" (XaaS) models for Cloud Computing services, where data is accessed, queried and updated on demand through a predefined service interface (usually a RESTful service). DaaS is based on Service-Oriented Architecture (SOA), and offers data services to be consumed by the third party applications in a unified format and to import data from data publishers.

Many of DaaS platform architecture includes several common components that LODaaS also does. Therefore, this section describes the processes and the requirements for each of the components as listed below:

- Data importing / transforming
- Data publishing
- Data retrieving / querying

1) Data Importing / Transforming

Data importing and transforming features are particularly important as a DaaS that provides RDF based Linked Data, because each of the resources should be named with URIs and the names should reuse the vocabularies to have unique ones for the same concepts or resources to be interlinked between the resources. In essence, the data publishing feature has to be combined with data importing feature to transform dataset provided by data publishers into RDF format, and store them into the repository.

To differ from general data warehouse platforms, this architecture should consider the characteristics of Linked Data and Open Data as described in TABLE I, in the perspective of Extract-Transform-Load (ETL) processes.

TABLE I. COMPARING ETL PROCESSES BETWEEN PLATFORMS

Process	Platform	
	Data Warehouse	***LODaaS***
Extract	extracting data from homogenous or heterogeneous data sources	extracting data from open data sets
Transform	deriving data to be loaded into end target	deriving data to be loaded into end target + URI mapping
Load	loading data into the end target	loading data into the triple store

2) Data Publishing

Data publishing process covers the processes for data transforming and providing machine-accessible data to the public [7]. Therefore, the architecture to support those processes should be designed. There are several related tools. D2R Server [8] provides the environment to publish relational data as Linked Data, Pubby [9] provides Linked Data interface for clients with HTML/RDF browsers, not only providing SPARQL endpoints. Paget [10] is a framework for building Linked Data applications, and PublishMyData [11] supports publishing Linked Data on the cloud and having access to it.

A case study of Linked Open Government Data (LOGD), "Publications Office of the European Union" indicates RDF based store on dedicated ontology (CDM, Common Data Model) as a key resource of its business model [12]. In addition, they have achieved the integration of their content and metadata based on the CDM. It is an example of how the Linked Data can be inter-connected based on the specific needs of users, which is useful because it may be adjusted to LDP to be used for domain specific purpose. This implies that LODaaS should be available for multiple purposes based on the ontology schemas (for each of them), and data should be published to the data layer of LODaaS following the schema. The data layer could be designed in two ways: to have independent repositories for each purpose, or to have integrated repository (Figure 2). However, regardless of the design, each datasets should be mapped to its corresponding ontology schema to be transformed and imported to the repository.

The first option is a more efficient way to manage data, because there is no confusion between ontology schemas when it performs ontology structural inference, and normally the application using LODaaS uses limited datasets stored in a specific repository instead of having the entire datasets. But if it is clear that there is no contradiction or conflicts, then the latter option is the preferred way to have expanded dataset based on Linked Data.

In summary, LODaaS architecture should support data transforming and importing, and the ontology schema should be loaded in the repository providing data interfaces.

 ISBN: 978-1-61208-445-9

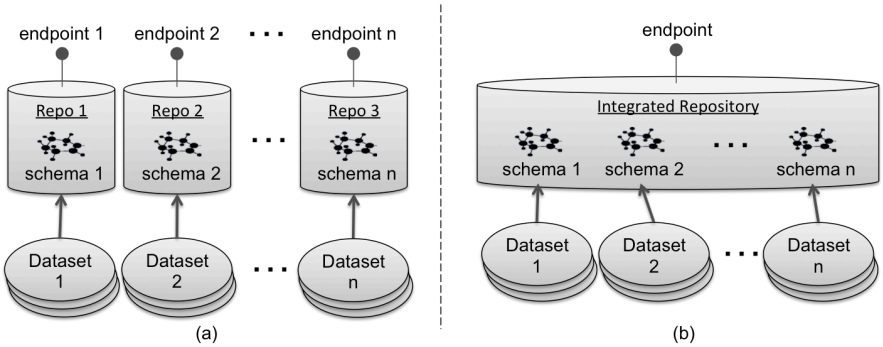

Figure 2. Two ways to locate repositories for LODaaS: (a) using independent repositories (b) using integrated repository

3) Data Retrieving / Querying

Linked Data relies on documents containing data in RDF format [13], therefore, LODaaS should provide SPARQL endpoints so that the clients can access the data through the interface. The clients need to write SPARQL query to retrieve data, which implies they need to use specific namespaces and URIs for resources. The data publishers and the clients as data consumers need to share the ontology schema information defined in the corresponding repositories or separated documentations, and this is where the ontology schema design is needed considering use cases, so that the data publishers and the consumers can be independent from each other. Details for this point is described in Section III (implementation), including the use case we have tried.

III. IMPLEMENTATION

A. Use case: PLUQI

Personalized and Localized Urban Quality Index (PLUQI) is the use case for this research defined as a customizable index model and mobile/Web application that can represent and visualize the level of well-being and sustainability for given cities based on individual preferences.

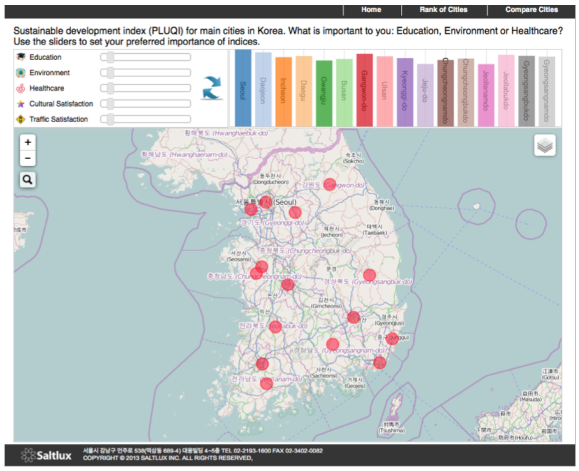

Figure 3. PLUQI Web Application

Figure 3 is a screenshot of the service webpage showing PLUQI indices. This is based on the open datasets published on DaPaaS, whose goal is to develop an integrated Data-as-a-Service (DaaS) and Platform-as-a-Service (PaaS) platform for open data applications. The proposed DaPaaS architecture will support unified accessibility to heterogeneous open datasets by using semantic technologies.

The users of PLUQI app will be provided with information about general satisfaction information from numbered index for the main regions/cities in Korea calculated based on the open datasets.

B. Processes & Architecture

To develop the PLUQI use case, we use the DaPaaS as a LODaaS to get open datasets about locational satisfaction, such as education, environment, transportation, etc. The steps for data publishing and consuming are as follows:

1. Collecting open datasets suitable for PLUQI from Korean open data catalogues
2. Transforming the open datasets into RDF format using Grafter, which is a Linked Data manufacturing tool for tabular data (Grafter [14], developed by Swirrl)
3. Importing and publishing the transformed data into DaPaaS
4. Consuming the published data from PLUQI app deployed in the DaPaaS via its SPARQL endpoint

The stakeholders can be summarized as shown in TABLE II.

TABLE II. STAKEHOLDERS FOR PLUQI USE CASE

Stakeholder	Participant
Data owner	Public sector (the person in charge of publishing open government data)
Data publisher	DaPaaS user (developer to transform and import the datasets, managing repositories)
Data consumer	PLUQI web application (developer to write queries and the user of PLUQI app)

We created a new repository of collected and transformed open datasets for PLUQI, from DaPaaS publisher portal GUI, and the SPARQL endpoint is provided. We also imported the PLUQI ontology schema to follow the architecture represented in Figure 2.

C. Ontology Schema Design

The PLUQI ontology schema helps data publishers and consumers to be independent from each other. The ontology schema can be represented as shown in Figure 4. *Quality_Index* can have *Value* to define measures described in open datasets, such as number of high schools as for educational satisfaction, which belongs to *Level_of_opportunity*. A *Value* has its *Location*, so we can describe that each location (regions/cities) has data of each categories represented as sub-classes of *Quality_Index*.

Following the design, data publishers can publish their datasets mapping for the categories, and the data consumers can query the data mapped for the categories they want to take. This means data consumers can get all the published data for *Quality_Index* without being too concerned about what kind of datasets exists for the categories. Especially, PLUQI should be available to get up-to-date indices and the data without changing queries used in the app regardless of what new datasets are published on DaPaaS.

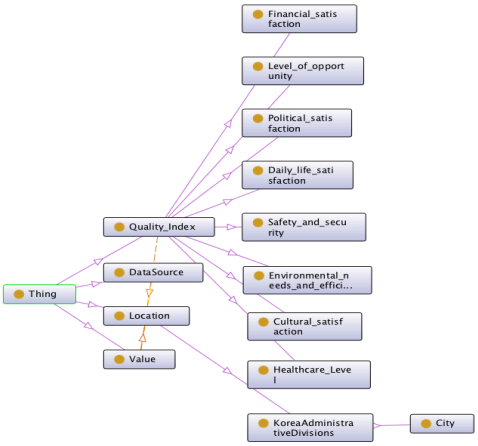

Figure 4. PLUQI Ontology Schema

IV. CONCLUSION

LODaaS has different features from the ones in LDP or DW, because it deals with open datasets, which are already open to the public and allows publishing their datasets. In addition, unlike Open Data catalogues, such as CKAN, LODaaS supports domain specific needs to provide useful data, so that the data is well transformed and managed considering the needs and creating a link between them.

To meet this need, we described the considerations for the processes of publishing open datasets to form Linked Data for specific use-case, and the architectures to manage and consume the data. It is difficult to be 'generally opened' and also 'targeting domain specific needs' at the same time, but this issue could be resolved by the agreed ontology schema between the stakeholders participating in the use case and by separating the repositories for each of them.

Our future work will be devoted to collect various data not only open datasets but also from social media which will be integrated with open datasets based on the PLUQI ontology schema to be published on DaPaaS.

ACKNOWLEDGMENT

This work was supported by the Industrial Strategic Technology Development Program (10044494, WiseKB: Big data based self-evolving knowledge base and reasoning platform) funded by the Ministry of Science, ICT & Future Planning (MSIP, Korea), and DaPaaS project which is funded by the European Commission under the 7th Framework Programme, Project No. 610988, http://dapaas.eu/.

REFERENCES

[1] W3C Linked Data Working Group, "Linked Data Platform 1.0W3C Proposed Recommendation 11 December 2014." https://dvcs.w3.org/hg/ldpwg/raw-file/default/ldp.html [retrieved: 2015.02.24]

[2] N. Mihindukulasooriya, R. García-Castro, and M. E. Gutiérrez, "Linked Data Platform as a novel approach for Enterprise Application Integration," COLD 2013.

[3] M. Gagnon, "Ontology-based integration of data sources." Proceedings of 10th International Conference on 10th International Conference on Information Fusion(FUSION2007), 2007, pp.1-8.

[4] M. Cobden, J. Black, N. Gibbins, Les Carr, and N. R. Shadbolt, "A Research Agenda for Linked Closed Dataset," COLD 2011. [Online]. Available from: http://eprints.soton.ac.uk/272711/ [retrieved: 2015.02.24]

[5] D. Wood, "Linking government data," Springer, 2011.

[6] D. Wood, "Linking Enterprise Data, 1st edition," Springer, 2010.

[7] W3C Government Linked Data Working Group, "Best Practices for Publishing Linked Data", http://www.w3.org/TR/ld-bp/ [retrieved: 2015.02.24]

[8] D2R Server. [Online]. Available from: http://www4.wiwiss.fu-berlin.de/bizer/d2r-server [retrieved: 2015.02.24]

[9] Pubby. [Online]. Available from: http://wifo5-03.informatik.uni-mannheim.de/pubby [retrieved: 2015.02.24]

[10] Paget. [Online]. Available from: https://code.google.com/p/paget [retrieved: 2015.02.24]

[11] PublishMyData [Online]. Available from: http://www.swirrl.com/publishmydata [retrieved: 2015.02.24]

[12] European Commission, "Study on business models for Linked Open Government Data", 2013. [Online]. Available from: https://joinup.ec.europa.eu/sites/default/files/85/31/25/Study_on_business_models_for_Linked_Open_Government_Data_BM4LOGD_v1.00.pdf. [retrieved: 2015.02.24]

[13] C. Bizer, T. Heath, and T. Berners-Lee, "Linked Data - the story so far," International Journal on Semantic Web and Information Systems, vol. 5, no. 3, pp. 1–22, 2009.

[14] Grafter [Online]. Available from: http://grafter.org [retrieved: 2015.02.24]

 ISBN: 978-1-61208-445-9