LINKEDLAB: A DATA MANAGEMENT PLATFORM FOR RESEARCH COMMUNITIES USING LINKED DATA APPROACH

Fariz Darari and Ruli Manurung

Information Retrieval Laboratory, Faculty of Computer Science, Universitas Indonesia, Kampus Baru UI Depok, Jawa Barat, 16424, Indonesia

E-mail: fadirra@gmail.com

Abstract

Data management has a key role on how we access, organize, and integrate data. Research community is one of the domain on which data is disseminated, e.g., projects, publications, and members. There is no well-established standard for doing so, and therefore the value of the data decreases, e.g. in terms of accessibility, discoverability, and reusability. LinkedLab proposes a platform to manage data for research communites using Linked Data technique. The use of Linked Data affords a more effective way to access, organize, and integrate the data.

Keywords: *data management, linked data, research community, semantic web*

Abstrak

Manajemen data memilki peranan kunci dalam bagaimana kita mengakses, mengatur, dan mengintegrasikan data. Komunitas riset adalah salah satu domain di mana data disebarkan, contohnya distribusi data dalam proyek, publikasi dan anggota. Tidak ada standar yang mengatur distribusi data selama ini. Oleh karena itu, *value* dari data cenderung menurun, contohnya dalam konteks *accessibility*, *discoverability*, dan *usability*. LinkedLab merupakan sebuah usulan *platform* untuk mengelola data untuk komunitas riset dengan menggunakan teknik Linked Data. Kegunaan Linked Data adalah sebuah cara yang efektif untuk mengakses, mengatur, dan mengitegrasikan data.

Kata Kunci: *komunitas penelitian, linked data, manajemen data, semantic web*

1. Introduction

Data management has an important role so that data can be accessed, organized, and integrated effectively. One approach for data management is Linked Data, a technique to publish and link structured data on the web. The Linked Data is based on four principles [1] that are, use IRIs as identifiers for things, use HTTP IRIs so that people can look them up, when someone dereferences an IRI, provide useful standards-based information, e.g., RDF, include links to other IRIs, so that they can discover more things.

The use of Linked Data approach provides many advantages, among of them are the ease of data access, simpler integration, and more sophisticated data processing by machines. These benefits increase the value of data with respect to the corresponding domains [2].

One of the data domains is research community. The information concerning research communities usually contains interlinked data, both internally within the scope of a research community itself, and externally with other organizations. For instance, data about research products and publications is related with data of the corresponding research project. Furthermore, the data about the project itself is also linked to the data of research members. These data and their relationships unfortunately are often hidden away in the form of HTML pages on research community platforms. Consequently, the data is difficult to disseminate and integrate with other data. The data is still locked in silos. Also, the data model is ambiguous and not explicit, which implies the problem of understanding the domain knowledge.

One possible approach to tackle these problems is proposed in [3]. Semantic MediaWiki (SMW) is used to build a semantic portal for the AIFB institute. The use of SMW combines the benefits of the semantic and social applications. Semantics is added to the platform content created

This paper is the extended version from paper titled "LinkedLab: A Linked Data Platform for Research Communities" that has been published in Proceeding of ICACSIS 2012.

and maintained collaboratively, thus creating a flexible, extensible, and structured knowledge representation.

Another possible approach is done using VIVO [4]. VIVO can be loaded with research activities, interests, and accomplishments, enabling the discovery of researchers across institutions. The use of shared ontology, i.e. the VIVO ontology, makes data integration process more straightforward and effective.

LinkedLab is built as a platform that could provide a data management solution based on the Linked Data approach. The data is not held in silos anymore, enabling easier reuse and integration by external applications, for instance, building a list of publications and projects done by various research communities. Also, more structured data enables machines to process data more easily. One of the complex queries that can be handled by Linked Data effectively is a query for a student who is looking for a supervisor on her specific research topic, concerning a specific project funded by a particular organization. Communication between research members also becomes unambiguous due to the use of ontologies in Linked Data. Based on these observations, the challenges of data management in research community as mentioned before can be solved by Linked Data, which serves as the basis for LinkedLab platform.

LinkedLab is a data management platform for research communities using Linked Data approach. It is designed to publish, edit, consume, and integrate data among research communities. LinkedLab is divided into three parts: the ontology, nodes, and integrator. The LinkedLab ontology will be the data schema. It defines a shared vocabulary to describe concepts and links in the research community domain. The LinkedLab nodes represent research communities distributed on different systems. The processes of data publication, editing, and consumption occur here. Last, the integrator is used to merge and query data over multiple nodes. The integration process makes use of the LinkedLab ontology that annotates data on each node. The overview of the LinkedLab architecture is depicted in figure 1.

The architecture refers to the Semantic Web cake **[**5**].** The LinkedLab itself is on the topmost layer, i.e. the user interface and applications layer. It is built using components from the layers specified below the user interface and applications layer, e.g., query, ontology, and data interchange. Each of them has been standardized by the W3C (World Wide Web Consortium) and is applied as SPARQL, OWL, and RDF, respectively. By using these building blocks, LinkedLab can apply the advantages of Linked Data in a standardized way. Semantic Web layer cake is depicted in figure 2. LinkedLab also combines both centralized and decentralized ways to manage data. The centralized notion is due to the shared ontology used as the semantic glue of data. The data itself is distributed over research communities, so each research community will have full control over its own data. This combination can be done due to the similar view of the data domain [6], i.e., research community.

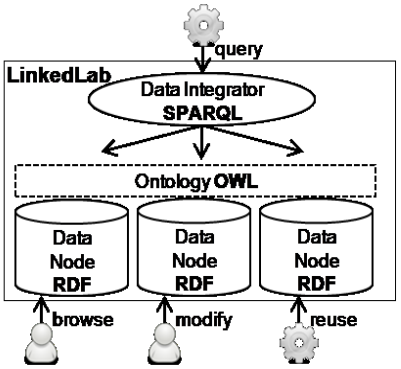

Figure 1. LinkedLab architecture.

An ontology defines a set of representational primitives, i.e., classes and properties, with which to model a domain of knowledge [7]. OWL (Web Ontology Language) is one of the knowledge representation languages developed to be consistent with the Web architecture. The LinkedLab ontology is meant to describe concepts and relationships related to research community domain such as research activities, outputs, and organizational profiles. Also, it is designed to be the semantic glue during data integration process among LinkedLab nodes. The LinkedLab ontology development method follows [8]. To determine the ontology scope, researcher list the corresponding competency questions, e.g., (1) Who are the members of a research group and what are their roles? (2) What papers were written by a research member and who are her co-authors? (3) What projects were done by a research laboratory and who funded them?

There are already various existing ontologies such as FOAF and BIBO, which can be reused. Each of these ontologies covers different aspects in the LinkedLab ontology. FOAF describes people, while BIBO specifies documents. Ontology reuse reduces the cost related to the ontology development and provides an interoperability benefit with other systems.

The research community domain basically can be divided into seven main terms: Person, Document, Project, Topic, Product, Event, and Organization [9]. Furthermore, researcher list

additional related terms such as Role, Membership, and Authorship. Researcher also observe the properties for each of these terms.

Then, researcher define OWL classes from the term list, following a top-down approach. For instance, as for the term Event, researcher first define the general event class Event, followed by more specific classes such as Conference, Meeting, Workshop, and so forth.

The remaining terms are then translated into OWL properties. There are two categories of OWL properties; object properties, which link between individuals, and datatype ones, which link an individual to data values. For instance, the class Event has object properties organizer, which denotes the organizer of the event, and based_near, which states the location of the event. Moreover, it also has datatype properties, such as startDate, endDate, and description. Researcher also observe the characteristics of properties such as domain, range, inverse, transitivity, and functionality. For instance, isSuperEventOf and isSubEventOf properties are inverse each other and both have Organization as their domain and range.

There is an issue researcher faced regarding the modeling of *n*-ary relations, since OWL properties are basically binary. Some relations in the LinkedLab ontology such as authorship and membership link an individual to multiple values. Therefore, researcher follow an approach that introduces a new class for each relation [10]. For instance, to model authorship, researcher create a class Authorship, and then researcher define three new properties such as author, document, and rank. The purpose of this is to describe the authorship of documents using individuals of the class Authorship. This approach is also implemented in the VIVO ontology, which is reused in the LinkedLab ontology, and can be illustrated in figure 3.

The node represents the data of a single research community, whose structure is given by LinkedLab ontology. It has three key functions: data publishing, editing, and consumption. All these functions follow the Linked Data principles mentioned earlier. Data publishing that the data publication process on the LinkedLab nodes must satisfy Linked Data principles. The data is identified using IRIs and can be dereferenced via HTTP IRIs together with 303 redirects and content negotiation [1].

For instance, the identifier for a researcher named "John Doe", can be given as "http://example.org/id/john-doe". So, if one dereferences the IRI, she will receive either the HTML page, which is human readable, or the RDF data, which is machine readable, depending on the content type request. All data published is annotated by LinkedLab ontology. To link the data to an external IRI, LinkedLab nodes also allow to add to the RDF description, the equivalence between two instances using owl:sameAs. Each node is equipped with a SPARQL endpoint as well, to which one can query over the data.

Data editing, that the editing process on the nodes covers creating, updating, and deleting data. Since LinkedLab nodes will be used by non-expert users, i.e., users with no prior knowledge of Linked Data, it is worth to make the editing process simple. One of the solutions is to use semantic forms. Such semantic forms look similar to regular forms, except they are used to edit RDF data. The form fields are driven by the LinkedLab ontology and the values are taken from the stored data. For instance, if we want to edit an instance of the Organization class, the corresponding properties such as name, homepage, and subOrganizationOf will appear as form fields.

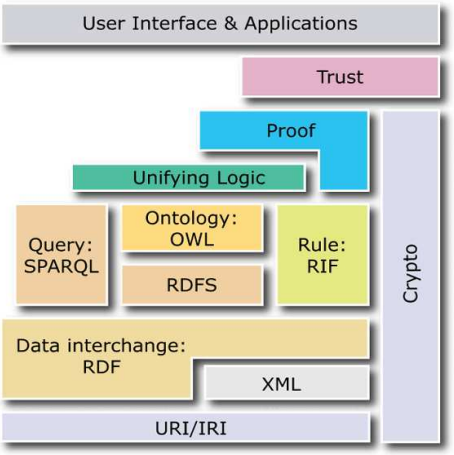

Figure 2. Semantic Web layer cake.

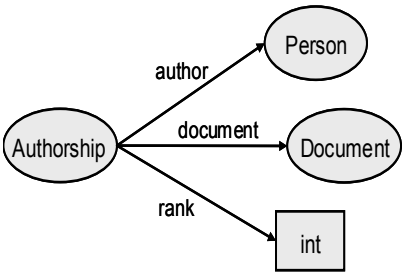

Figure 3. Authorship modeling.

Data consumption, that there are two benefits of using Linked Data for data consumption: openness and standardized data representation and access. A LinkedLab node is able to consume both its own data and external data, e.g., the LOD cloud [2]. The consumption patterns are varied such as browsing, searching, and visualizing data. Further, by importing external data, one can give an extended description for instances on nodes, e.g., import data from DBPedia, a part of LOD cloud, to give more information about Information Retrieval research topic.

The integrator is meant to integrate data among LinkedLab nodes, so that one can do a query to the merged data. The use of Linked Data gives easier data integration, as it relies on standardized data models and access mechanisms.

A SPARQL server employing federated extension is needed to build the integrator. The federated extension will send parts of a SPARQL query into a set of remote endpoints. The extension uses the SERVICE keyword to operate, whose pattern can be seen in figure 4.

```
#variables to appear
SELECT ...
#query pattern
WHERE
{
...
#subquery pattern to be sent to remote
endpoints
      SERVICE ?endpoint{
        ...
        }
 }
```

Figure 4. Federated query pattern.

The pattern above involves a variable, i.e., ?endpoint, which is bound to the data in the default graph concerning remote endpoints on LinkedLab nodes. The graph makes use of the voiD vocabulary [11]. An example of a description of endpoint can be depicted in figure 5.

```
<http://example.org/> a void:Dataset ;
 void: sparqlEndpoint
 <http://example.org/sparql>.
```

Figure 5. Endpoint description using turtle syntax.

2. Methodology

The implementation of LinkedLab involves five steps: ontology implementation, Semantic MediaWiki reuse, ontology integration, data population, and data integration. The ontology implementation gives an OWL file as the output. As for the data population, researcher employ Semantic MediaWiki that is already integrated with the LinkedLab ontology. Semantic MediaWiki serves as the basis for the LinkedLab nodes. The data integration is done by Fuseki, a SPARQL server that already supports federated query extension.

Researcher implement LinkedLab ontology using Protégé-OWL ontology editor **[**12**]**. Researcher reuse external ontologies such as FOAF and BIBO and merge them with an internal vocabulary containing terms in a separate namespace, i.e., lab:, that are not defined already in the external ontologies. The LinkedLab ontology is serialized using RDF/XML syntax. Figure 6 shows the implementation of OWL properties in Protégé-OWL.

Semantic MediaWiki (SMW) is a wiki application embedded with semantic functionalities [3]. The supported semantic functionalities have strong correspondences with the requirements to implement LinkedLab nodes. All data that is created within SMW can be published via Linked Data.

The IRI pattern of SMW to acquire the RDF data of an instance is http://{{site_name}}/wiki/index.php/Special:URI Resolver/{{instance_name}}. To model *n*-ary relations, SMW can be equipped with the Semantic Internal Objects extension. There is also Semantic Forms extension, which provides a form based interface to edit SMW data. As for the data consumption, SMW has various built-in features such as semantic search on Special:Ask page, semantic browsing using Special:Browse page, and can be extended with the Semantic DrillDown, Semantic Result Formats, and External Data extensions for faceted browsing, data visualization, and external data reuse, respectively. SMW can also be synchronized with a so-called triple store, a kind of database optimized to store RDF data, thus enabling data integration via SPARQL endpoint. Researcher reuse Joseki as the triple store for LinkedLab nodes. Based on all above considerations,

researcher decide to employ Semantic MediaWiki as an underlying platform for LinkedLab nodes.

Ontology integration means a process to merge the LinkedLab ontology into LinkedLab nodes. Researcher make a program supporting this process that is called OWL2SMW. The program reuses both Protégé-OWL API and Pywikipedia bot. There are two steps of operating OWL2SMW. First, the program converts the LinkedLab ontology into SMW syntax serialization using Protégé-OWL API, following the conversion **guidelines from [13]. Next, the Pywikipedia bot** creates corresponding pages on the Semantic MediaWiki, e.g. class, property, and instance pages.

LinkedLab provides two ways of doing data population. First, if the data is already in OWL format, one can use OWL2SMW. The program transforms the input data into SMW syntax serialization. For instance, [[Category:{{class name}}]] is used to denote classes, while [[{{property name}}::{{property values}}]] is used to denote properties. Furthermore, Pywikipedia bot will generate instances and their properties as wiki pages. Alternatively, one can also manually input data using Semantic Forms extension that is embedded on the node.

Researcher reuse Fuseki for the LinkedLab integrator. The configuration is stored as RDF file and is loaded as default graph, containing description about endpoints from which the data of LinkedLab nodes can be queried. The query for merged data can be done using the provided query form or HTTP GET request. The result format of the query can be chosen between XML, JSON, or CSV.

3. Results and Analysis

The evaluation is done by performing some use cases regarding Linked Data application such as data publishing, editing, consumption, and integration. Researcher use research community data from the Information Retrieval Lab at Universitas Indonesia as testing data to build the LinkedLab node. The data is extracted from the lab website using Perl. Researcher also build two other LinkedLab nodes using dummy data particularly for data integration evaluation.

The publishing feature is the most important one in LinkedLab node. Researcher evaluate it by checking if the data is already published conforming to the RDF publishing best practices [2], i.e., employing 303 redirects and content negotiation correctly. cURL is used to get the data published in the node, as it could simulate how Linked Data browser works.

One of the scenarios is to get the RDF data of ICACSIS 2010, an instance of Conference class. First, researcher send an HTTP request to its IRI, as shown in figure 7.

```
Curl -I -H "Accept: application/rdf+xml"
http://localhost/wiki/index.php/Special:URI
Resolver/ICACSIS2010
```

Figure 7. HTTP request for RDF data.

The node then replies with the header 303 See other and parameter location: http://localhost/wiki/index.php?title=Special:ExportRDF/ICACSIS2010&xmlmime=rdf. The request will be redirected to the location of the RDF data, which describes the homepage, organizer, and start date of the conference. The received RDF data can be seen in figure 8.

The editing process can be done through Semantic Forms. For instance, to edit the instance ICACSIS 2010, one needs to go to http://localhost/wiki/index.php/Special:FormEdit/swrc:Conference/ICACSIS2010. The form displayed in figure 9 shows the properties of the instance.

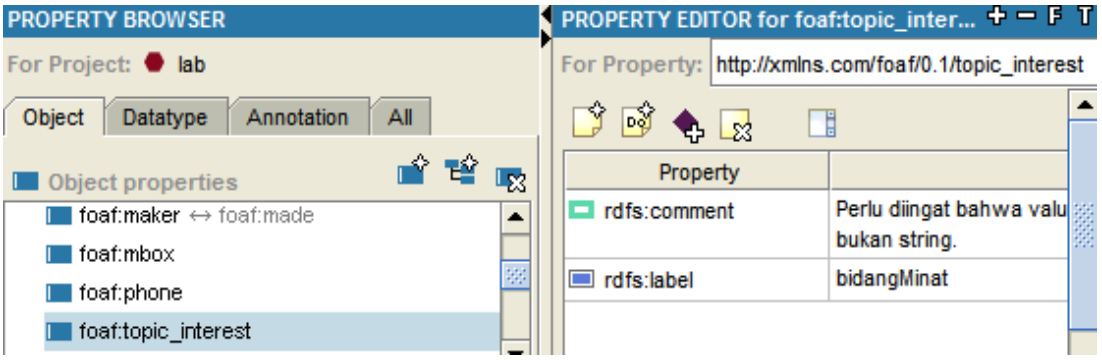

Figure 6. Protégé-OWL property editor.

```
<rdf:type
rdf:resource= “&swrc;Conference”/>
 <swivt:wikiNamespace
rdf:datatype=
“http://www.w3.org/2001/XMLSchema#integer”>
0</swivt:wikiNamespace>
 <foaf:name
rdf:datatype=
“http://www.w3.org/2001/XMLSchema#string”>I
CACSIS 2010</foaf:name>
 <foaf:homepage
rdf:resource=
“http://icacsis.cs.ui.ac.id/”/>
 <bibo:organizer
rdf:resource=
“&wiki;Faculty_Of_Computer_Science_UI”/>
 <swrc:startDate
rdf:datatype=
“http://www.w3.org/2001/XMLSchema#dateTime”
>2010-10-20T00:00:00</swrc:startDate>
```

Figure 8. Snippet of RDF data.

Researcher evaluate one of the data consumption features, i.e., Semantic Search. SMW provides Semantic Search to query its own data. The query language is SMW QL [13]. One can access Special:Ask page for Semantic Search. Researcher execute queries that utilize the ontology-based data structure on the LinkedLab node. For instance, if researcher want to find a person who has written a paper on ICACSIS 2010, the respective query can be seen as in figure 10.

Data integration among LinkedLab nodes is implemented using SPARQL with its federated extension. The executed query must follow the federated query pattern as mentioned before. For instance, to get all publications of knowledge representation, including the name of its first author, researcher invoke the query below (figure 11).

Figure 9. Semantic forms.

```
[[Category:foaf:Person]] [[-
vivo:linkedAuthor::<q>[[vivo:linkedInformat
ionResource::<q>[[bibo:presentedAt::ICACSIS
2010]]</q>]]
```

Figure 10. Semantic search query.

The first subquery before the SERVICE block gets all endpoints registered on the default graph. Next, the SERVICE keyword invokes the subqueries inside the block to each remote endpoint. After retrieving the publications from each endpoint, the results are then merged.

```
SELECT ?dataset ?title ?author_name
WHERE {
?dataset void:sparqlEndpoint ?ep.
SERVICE ?ep
{
  ?authorship vivo:linkedAuthor ?person
  ?authorship vivo:authorRank 1 .
  ?person foaf:name ?author_name .
  ?authorship
Vivo:linkedInformationResource ?pub .
  ?pub dcterms:title ?title .
  ?pub dcterms:subject ?topic .
  ?topic owl:sameAs
<http://dbpedia.org/resource/Knowledge_repr
esentation>
   }
}
```

Figure 11. Federated query.

4. Conclusion

LinkedLab serves as a solution for data management in research community domain. Linked Data principles are used as the foundations for the platform development, i.e., for data publishing, editing, consumption, and integration. Linked Data brings easier reuse and integration, more sophisticated data processing, and a common understanding that can improve the communication among research members.

As for the next development, one could extend the LinkedLab ontology with other relevant ontologies such as resume and CV ontologies. Moreover, there is a need to implement more use cases for data consumption, e.g., faceted browsing and data visualization of merged data taken from the LinkedLab integrator.

Acknowledgements

Researcher thank to Evi Yulianti, S.Kom., for the help providing data during evaluation phase.

References

[1] C. Bizer, T. Heath, & T. Berners-Lee, “Linked Data - The Story So Far,” *International Journal on Semantic Web and Information Systems (IJSWIS)*, vol. 5, pp. 1-22, 2009.

[2] T. Heath & C. Bizer, *Linked Data: Evolving the Web into a Global Data Space,* 1st ed., Morgan & Claypool, 2011.

[3] D.M. Herzig & B. Ell, "Semantic MediaWiki in Operation: Experiences with Building a Semantic Portal" *In Proceedings of International Semantic Web Conference*, pp. 114-128, 2010.

[4] VIVO, VIVO Ontology, VIVO, http://vivoweb.org/, 2011, retrieved December 15, 2011.

[5] World Wide Web Consortium (W3C), Semantic Web Layer Cake, W3C, http://www.w3.org/2007/03/layerCake.png, 2011, retrieved December 15, 2011.

[6] H. Stuckenschmidt & F.V. Harmelen, *Information Sharing on the Semantic Web*. Springer-Verlag Berlin Heidelberg, Germany, 2005.

[7] T. Gruber, in: L. Liu, & M. T. Ozsu (Eds.), *Ontology*, *Encyclopedia of Database Systems*, Springer-Verlag, Germany, pp. 1963-1965, 2009.

[8] N.F. Noy & D.L. McGuinness, Ontology Development 101: A Guide to Creating Your First Ontology, Protégé, http://protege.stanford.edu/publications/ontology_development/ontology101.html, 2001, retrieved December 16, 2011.

[9] Y. Sure, S. Bloehdorn, P. Haase, J. Hartmann, & D. Oberle, "The SWRC Ontology - Semantic Web for Research Communities" *In Proceedings of the 12th Portuguese Conference on Artificial Intelligence - Progress in Artificial Intelligence (EPIA 2005),* pp. 218-231, 2005.

[10] N.F. Noy & A. Rector, Defining N-ary Relations on the Semantic Web, W3C, http://www.w3.org/TR/swbp-n-aryRelations/. 2006, retrieved December 16, 2011.

[11] K. Alexander, R. Cyganiak, M. Hausenblas, & J. Zhao, Describing Linked Datasets with the VoID Vocabulary, W3C, http://www.w3.org/2001/sw/interest/void/, 2011, retrieved December 17, 2011.

[12] Protégé, What is Protégé-OWL?, Protégé, http://protege.stanford.edu/overview/protege-owl.html, 2011, retrieved December 20, 2011.

[13] J. Bao, L. Ding, & J. Hendler, Knowledge Representation and Query in Semantic MediaWiki: A Formal Study, Technical Report, Tetherless World Constellation (RPI), New York, 2008.

[14] J. Bao & D. Calvanese, OWL 2 Web Ontology Language Overview, W3C, http://www.w3.org/TR/owl2-overview/, 2009, retrieved December 19, 2011.

[15] E. Prud'hommeaux & C. Buil-Aranda, SPARQL 1.1 Federated Query, W3C, http://www.w3.org/2009/sparql/docs/fed/service, 2011, retrieved December 18, 2011.