Linked Data Mashups: A Review on Technologies, Applications and Challenges

Tuan Nhat Tran, Duy Khanh Truong, Hanh Huu Hoang, and Thanh Manh Le

Hue University
3 Le Loi Street, Hue City, Vietnam
hhhanh@hueuni.edu.vn

Abstract. To remedy the data integration issues of the traditional Web mashups, the Semantic Web technology uses the Linked Data based on RDF data model as the unified data model for combining, aggregating and transforms data from heterogeneous data resources to build Linked Data Mashups. There have been tremendous amount of efforts of Semantic Web community to enable Linked Data Mashups but there still lack of a systematic survey on concepts, technologies, applications and challenges. Therefore, this paper gives an overview of Linked Data Mashups and conducts a state-of-the-art survey about technologies, applications and challenges on Linked Data Mashups.

Keywords: Linked Data, Mashups, RDF, Semantic Web.

1 Introduction

The development of generic Web applications is well understood and supported by many traditional computer science domains, such as classical database applications. In current Web application development data integration and access are typically dealt with by fairly sophisticated abstractions and tools which support rapid application development and the generation of reusable and maintainable software components. The task of programming such applications has become the task of combining existing components from well-established component libraries, i.e., customizing and extending them for application-specific tasks. Typically, such applications are built relying on a set of standard architectural styles which shall lower the number of bugs and ensure code that is easy to understand and maintain.

Historically, the process of writing new queries and creating new graphic interfaces has been something that has been left to the experts. A small team of experts with a limit skill-sets would create applications, and all users would have to use what was available, even if it did not quite fit their needs [16]. A *mashup* is an (web) application that offers new functionality by combining, aggregating and transforming resources and services available on the web [16]. Therefore, mashups are an attempt to move control over data closer to the user and closer to the point of use. Although mashups are technically similar to the data integration techniques that preceded them, they are philosophically quite different. While data integration has historically been about

N.T. Nguyen et al. (Eds.): ACIIDS 2014, Part II, LNAI 8398, pp. 253–262, 2014.

allowing the expert owners of data to connect their data together in well-planned, well-structured ways, mashups are about allowing arbitrary parties to create applications by repurposing a number of existing data sources, without the creators of that data having to be involved [5]. Therefore, mashup enabling technologies not only reduce the effort of building a new application by reusing available data sources and systems but also allow the developers to create novel applications beyond imagination of the data creators. However, the traditional web mashups still suffers the heterogeneity of data coming from different sources having different formats and data schema. To remedy the data integration issues of the traditional web mashups, the Semantic Web technology uses the Linked Data based on RDF data model as the unified data model for combining, aggregating and transforms data from heterogeneous data resources to build Linked Data Mashups. Powered by tools and technologies having been developed by the Semantic Web community, there are various applications domains building applications with Linked Data Mashups.

There has not been any work that give a comprehensive survey about technologies and applications of Linked Data Mashups as well the challenges for building Linked Data Mashups. This shortcoming comes from several following reasons. Typical Linked mashups are data-intensive and require the combination and integration of RDF data from distributed data sources. In contrast to that, data-intensive applications using RDF are currently mostly custom-built with limited support for reuse and standard functionalities are frequently re-implemented from scratch. While the use of powerful tools such as SPARQL processors, takes the edge off of some of the problems, a lot of classical software development problems remain. Also such applications are not yet built according to agreed architectural styles which are mainly a problem of use rather than existence of such styles. This problem though is well addressed in classical Web applications. For example, before the introduction of the standard 3-tier model for database-oriented Web applications and its support by application development frameworks, the situation was similar to a lot the situation that we see now with RDF-based applications [1].

This paper aims at giving a systematic view about Linked Data Mashups. It will give an overview of Linked Data Mashups in Section 2. Then, in Section 3, the paper gives a survey about enabling technologies for Linked Data Mashups such as data integration, mashup execution engines, interactive programing and visualization. Section 3 will introduce a series of applications domains for Linked Data Mashups. The challenges of building Linked Data Mashups are discussed in Section 4.

2 Linked Data Mashups

2.1 Linked Data

The term *Linked Data* refers to a set of best practices for publishing and linking structured data on the Web. These best practices were introduced by Tim Berners-Lee in his Web architecture note namely *Linked Data* [17] and have become known as the *Linked Data principles*. These principles are descripted as: the basic idea of Linked Data is to apply the general architecture of the World Wide Web to the task of sharing structured data on global scale [7].

Linked Data principles firstly advocates using URI references to identify, not just Web documents and digital content, but also real world objects and abstract concepts. These may include tangible things such as people, places and cars, or those that are more abstract, such as the relationship type of *knowing somebody*. Linked Data use the HTTP protocol for Web resources access mechanism with the use of HTTP URIs to identify objects and abstract concepts, enabling these URIs to be *dereferenced* (i.e., looked up) over the HTTP protocol into a description of the identified object or concept. Linked Data principle also advocates use of a single data model for publishing structured data on the Web – the Resource Description Framework (RDF), a simple graph-based data model that has been designed for use in the context of the Web [7]. Lastly, Linked Data uses of hyperlinks to connect not only Web documents, but any type of thing. For example, a hyperlink may be set between a person and a place, or between a place and a company. Hyperlinks that connect things in a Linked Data context have types which describe the relationship between the things. For example, a hyperlink of the type *friend of* may be set between two people, or a hyperlink of the type *based near* may be set between a person and a place. Hyperlinks in the Linked Data context are called *RDF links* in order to distinguish them from hyperlinks between classic Web documents.

The RDF data model represents information as node-and-arc-labelled directed graphs. The data model is designed for the integrated representation of information that originates from multiple sources, is heterogeneously structured, and is represented using different schemata [7, 12]. Data is represented in RDF as RDF *triples*. The RDF data model is described in detail as part of the W3C RDF Primer[1].

2.2 Linked Data Mashups

Linked Data Mashups are created in the similar fashion as web mashups whilst they use a unified data model, RDF model for combining, aggregating and transforming data from heterogeneous data resources. Using a single data model for data manipulation operations enables a simpler abstraction of application logics for mashup developers. The RDF data model is driven by vocabularies or ontologies which play the role as the common understanding among machines, developers, domain experts and users.

A Linked Data Mashup is composed from different piece of technologies. The first type of technologies is data integration which covers data transformation, storage, accessing and application APIs based on RDF data model. The second type of technologies is mashup execution engines which provide the execution eviroments for computing the mashup processing workflow. The third type of technologies is interactive programing and visualization which provide a composing and exploring environments for mashup developer to build data processing workflow for a mashup (Section 3).

One simple example of a Linked Data Mashup is an aggregated Sales application that integrates customer relationship management (CRM) and financial data with functionality from the Web and corporate backend data. This example mashup would

[1] W3C RDF Primer, `http://www.w3.org/TR/rdf-primer/`

employ real-time information, streaming content, and Web services to form a coordinated application using all of these data sources. Integrated sales information for the traveling sales person could be available from their smart phone or laptop. The data integration tools are responsible for transforming streams real-time Web information of financial and CRM data and Background information and Request for Information (RFI) documents to Linked Data. Internally, Internal, proprietary customer data about installed products, contracts, and upsell possibilities can be exposed as Linked Data via RDFisers [7]. When all the data are accessible as Linked Data and can be queried via SPARQL, there is a series of front-end application can be built. The facet browsers for Linked Data [3, 14] enable combining financial, CRM and other data with online maps to visually identify, locate and categorise customers for each geographical location. Using Google Maps or Mapquest[2] APIs, each customer site appears on the map and allows the sales person to drill down using the map paradigm to identify customer sites to expose new sales or possible upsell opportunities. Background information and RFI documents could be generated partly using semantically rich content from DBpedia (http://dbpedia.org/), the semantically structured content from Wikipedia. Integrated and updated glossary definitions of domain vernacular, references to partners and competitors could come together as competitive analysis documents. Prospective customers could read marketing evaluations combined with general reference content, and links to trusted independent blogger opinions, all from a single document. Customer data can be integrated with the maps, reference information, and sales database to provide personalised content for customers.

3 Technologies Enabling Linked Data Mashups

3.1 Data Integration

Data integration technologies for Linked Data Mashups involve all solutions and tools to enable data from heterogeneous sources accessible as Linked Data. The representative architecture for data integration of Linked Data Mashups is depicted in Fig. 3.

In this architecture, the publishing layer provides all tools to expose traditional data sources in RDF data formats. They include wrappers for the databases, RDFizers for transforming data from other format like XML, JSON, HTML into RDF. Then when all data is accessible as Linked Data it might be stored in storages or accessed via Web APIs such as SPARQL Endpoints, called Web Linked Data. These data might be manipulated and integrated to access in a refined form via a SPARQL query interface by application code in the application layer.

3.2 Mashup Execution Engines

A mashup is usually constructed in a formal language to representing the computing process that generates the output for the mashup. Then the mashup represented in a

[2] http://www.mapquest.com/

such language is executed in an execution engine. In this section, we introduce two popular execution engines, MashMaker [5] and DERI Pipes [1]. MashMaker uses functional programing language whilst DERI Pipes uses Domain Specific Language (DSL) in XML

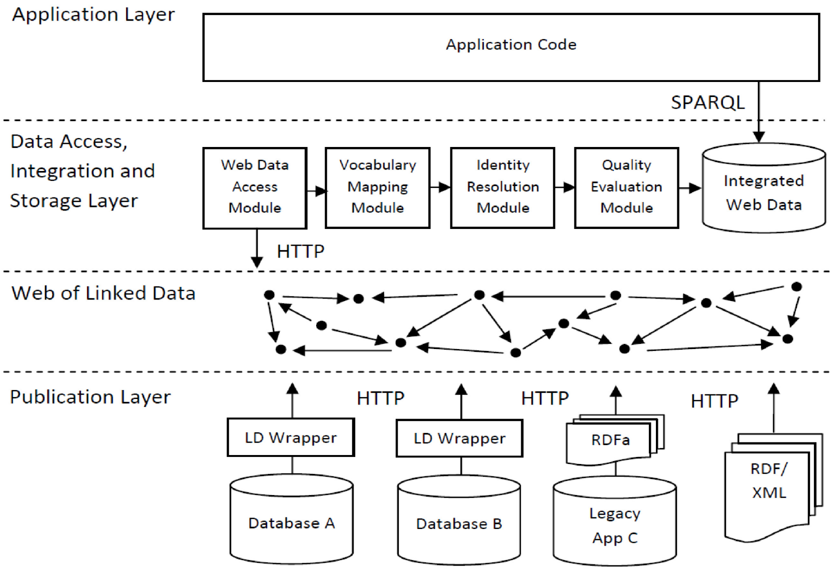

Fig. 1. Data integration architecture for Linked Data Mashups [7]

MashMaker provides a modern functional programming language with non-side effecting expressions, higher order functions, and lazy evaluation. MashMaker programs can be manipulated either textually, or through an interactive tree representation, in which a program is presented together with the values it produces. MashMaker expressions are evaluated lazily. The current consensus in the programming language community seems to now be that lazy evaluation is the wrong evaluation model for conventional programming languages. This is because the bookkeeping overhead of lazy evaluation makes programs run slowly, the complex evaluation behavior makes performance hard to predict, and programmers often have to battle with space leaks due to long chains of lazy thunks. In the case of web mashups, the bookkeeping cost of remembering how to evaluate something is tiny compared to the massive cost of fetching and scraping a web site, thus it is only necessary for a very small number of expressions to be unneeded for the bookkeeping cost to be more than paid back. Even if fetching a web site was cheap, it is important for us to minimize the number of queries we make to a remote server, to avoid overwhelming a server. Typical mashup programs work with relatively small amounts of data that are not directly presented to the user, and so space leaks are far less of a problem.

DERI Pipes [1] proposes a flexible architectural style for the fast development of reliable data intensive applications using RDF data. Architectural styles have been around for several decades and have been the subject of intensive research in other domains such as software engineering and databases. Le-Phuoc et al. [1] bases their work on the classical pipe abstraction and extends it to meet the requirements of (semantic) Web applications using RDF. The pipe concept lends itself naturally to the data-intensive tasks at hand by its intrinsic concept of decomposing an overall data-integration and processing task into a set of smaller steps which can be freely combined. This resembles a lot the decomposition of queries into smaller sub queries when optimizing and generating query plans. To some extent, pipes can be seen as materialized query plans defined by the application developer. Besides, the intrinsic encapsulation of core functionalities into small components, this paradigm is inherently well suited to parallel processing which is an additional benefit for high-throughput applications which can be put on parallel architectures.

3.3 Interactive and Visual Programming

As more and more reusable structured data appears on the Web, casual users will want to take into their own hands the task of mashing up data rather than wait for mash-up sites to be built that address exactly their individually unique needs. Therefore, an interactive and visual programming environment is desired for building Linked Data Mahsups. The techniques and tools like facet-browsing, Web GUI facilitate to interactive mashup developing editors such as Potluck [3], Exhibit [4] and IntelMash Maker [5].

Potluck [3] provides a Web user interface that let’s casual users—those without programming skills and data modelling expertise—mash up data themselves. Potluck is novel in its use of drag and drop for merging fields, its integration and extension of the faceted browsing paradigm for focusing on subsets of data to align, and its application of simultaneous editing for cleaning up data syntactically. Potluck also lets the user construct rich visualizations of data in-place as the user aligns and cleans up the data. This iterative process of integrating the data while constructing useful visualizations is desirable when the user is unfamiliar with the data at the beginning—a common case—and wishes to get immediate value out of the data without having to spend the overhead of completely and perfectly integrating the data first.

Exhibit [14] is a lightweight framework for publishing structured data on standard web servers that requires no installation, database administration, or programming. Exhibit lets authors with relatively limited skills-those same enthusiasts who could write HTML pages for the early Web-publish richly interactive pages that exploit the structure of their data for better browsing and visualization. Such structured publishing in turn makes that data more useful to all of its consumers: individual readers get more powerful interfaces, mashup creators can more easily repurpose the data, and Semantic Web enthusiasts can feed the data to the nascent Semantic Web.

IntelMash Maker [5] does this by making mashup creation part of the normal browsing process. Rather than having a reasonably skilled user create a mashup in advance as a mashup site that other users browse to, MashMaker instead creates personalized mashups for the user inside their web browser. Rather than requiring that a user tell a mashup tool what they want to create, MashMaker instead watches what information the user looks at, correlates the user’s behaviour with that of other users, and guesses a mashup application that the user would find useful, without the user even having to realize they wanted for a mashup.

4 Application Domains

4.1 DBpedia Mashups

If you see Wikipedia as a main place where the knowledge of mankind is concentrated, then DBpedia [12]—which is extracted from Wikipedia—is the best place to find the machine representation of that knowledge. DBpedia constitutes a major part of the semantic data on the web. Its sheer size and wide coverage enables you to use it in many kind of mashups: it contains biographical, geographical, bibliographical data; as well as discographies, movie metadata, technical specifications, and links to social media profiles and much more. Just like Wikipedia, DBpedia is a truly cross-language effort, e.g., it provides descriptions and other information in various languages. DBpedia is an unavoidable resource for applications dealing with commonly known entities like notable persons, places; and for others looking for a rich hub connecting other semantic resources.

4.2 Mashups for Internet of Things

Internet of Things (IoT) has been creating vast amount of distributed stream data which can be modelled using RDF data model called Linked Stream Data. Linked Stream Data is becoming new valuable data sources for Linked Data Mashups. Therefore, the Web of Things (WoT) together with mashup-like applications is gaining popularity with the development of the Internet towards a network of interconnected objects, ranging from cars and transportation cargos to electrical appliances.

A long the same line, cities are alive: they rise, grow, and evolve like living beings, WoT allows a wide rage of Smart City applications. In essence, the state of a city changes continuously, influenced by a lot of factors, both human (people moving in the city or extending it) and natural ones (rain or climate changes). Cities are potentially huge sources of data of any kind and for the last years a lot of effort has been put in order to create and extract those sources. This scenario offers a lot of opportunities for mashup developers: by combining and processing the huge amount of data (both public and private) is possible to create new services for urban stakeholders—citizens, tourists, etc. called urban mashups [9].

Another application domain for IoT is emergency management [10]. Emergency management applications support a command staff in disruptive disaster situations, such as earthquakes, large-scale flooding or fires. One crucial requirement to emergency management systems is to provide decision makers with the relevant information to support their decisions. Mashups can help here by providing flexible and easily understandable views on up-to-date information.

4.3 Tourism Mashups

Web 2.0 has revolutionized the way users interact with information, by adding a vast amount of services, where end users explicitly and implicitly, and as a side effect of their use, generate content that feeds back into optimization of these services.

The resulting (integrated) platforms support users in and across different facets of life, including discovery and exploration, travel and tourism. Linked Data Mashup enables the creation and use of Travel Mashups, defined based on the varied travel information needs of different end users, spanning temporal, social and spatial dimensions [8]. The RDF-based travel mashups are created for bridging these dimensions, through the definition and use of composite, web- and mobile-based services. Their applications elicit the information need of an end user exploring an unfamiliar location, and demonstrates how the Topica Travel Mashup leverages social streams to provide a topical profile of Points of Interest that satisfies these user's requirements.

4.4 Biological and Life Science Domains

Semantic Web technologies provide a valid framework for building mashups in the life sciences. Ontology-driven integration represents a flexible, sustainable and extensible solution to the integration of large volumes of information. Additional resources, which enable the creation of mappings between information sources, are required to compensate for heterogeneity across namespaces. For instance, [6] uses an ontology-driven approach to integrate two gene resources (Entrez Gene and HomoloGene) and three pathway resources (KEGG, Reactome and BioCyc), for five organisms, including humans. Satya et al. [6] created the Entrez Knowledge Model (EKoM), an information model in OWL for the gene resources, and integrated it with the extant BioPAX ontology designed for pathway resources. The integrated schema is populated with data from the pathway resources, publicly available in BioPAX-compatible format, and gene resources for which a population procedure was created. The SPARQL query language is used to formulate queries over the integrated knowledge base to answer the three biological queries. Simple SPARQL queries could easily identify hub genes, i.e., those genes whose gene products participate in many pathways or interact with many other gene products. The identification of the genes expressed in the brain turned out to be more difficult, due to the lack of a common identification scheme for proteins.

5 Open Challenges

Even there has been a plenty of technology and research achievements of Linked Data community to enable Linked Data Mashups, there are a number of challenges to address when building mashups from different sources. The challenges can be classified into four groups: Entity extraction from text, object identification and consolidation, abstraction level mismatch, data quality.

Transforming Text Data to Symbolic Data for Linked Data Entities. A large portion of data is described in text. Human language is often ambiguous - the same company might be referred to in several variations (e.g. IBM, International Business Machines, and Big Blue). The ambiguity makes cross-linking with structured data difficult. In addition, data expressed in human language is difficult to process via software programs. Hence overcoming the mismatch between documents and data to extract RDF-based entities is still emerging challenges.

Object Identification and Consolidation. Structured data are available in a plethora of formats. Lifting the data to a common data format is thus the first step. But even if all data is available in a common format, in practice sources differ in how they state what is essentially the same fact. The differences exist both on the level of individual objects and the schema level. As an example for a mismatch on the object level, consider the following: the SEC uses a so-called Central Index Key (CIK) to identify people (CEOs, CFOs), companies, and financial instruments while other sources, such as DBpedia, use URIs to identify entities. In addition, each source typically uses its own schema and idiosyncrasies for stating what is essentially the same fact. Thus, methods have to be in place for reconciling different representations of objects and schemata.

Abstraction Levels. Data sources provide data at incompatible levels of abstraction or classify their data according to taxonomies pertinent to a certain sector. Since data is being published at different levels of abstraction (e.g. person, company, country, or sector), data aggregated for the individual viewpoint may not match data e.g. from statistical offices. Also, there are differences in geographic aggregation (e.g. region data from one source and country-level data from another). A related issue is the use of local currencies (USD vs. EUR) which have to be reconciled in order to make data from disparate sources comparable and amenable for analysis.

Data Quality. Data quality is a general challenge when automatically integrating data from autonomous sources. In an open environment the data aggregator has little to no influence on the data publisher. Data is often erroneous, and combining data often aggravates the problem. Especially when performing reasoning (automatically inferring new data from existing data), erroneous data has potentially devastating impact on the overall quality of the resulting dataset. Hence, a challenge is how data publishers can coordinate in order to fix problems in the data or blacklist sites which do not provide reliable data. Methods and techniques are needed to; check integrity, accuracy, highlight, identify and sanity check, corroborating evidence; assess the probability that a given statement is true, equate weight differences between market sectors or companies; act as clearing houses for raising and settling disputes between competing (and possibly conflicting) data providers and interact with messy erroneous Web data of potentially dubious provenance and quality. In summary, errors in signage, amounts, labeling, and classification can seriously impede the utility of systems operating over such data.

6 Conclusion

In this paper we have investigated state-of-the-art approaches in Linked Data Mashups in terms of technologies and application domain. From analytical reviews on approaches, we have drawn up open challenges for Linked Data Mashups. This review is a first steps of our research aiming at pointing out research trends in building up real applications. They are based on the open linked data in order to make a new leash of intelligent applications that utilise and facilitate the advantages of RDF data model and Linked Data for new generation of Linked Data Mashups application line.

References

[1] Le-Phuoc, D., Polleres, A., Hauswirth, M., Tummarello, G., Morbidoni, C.: Rapid prototyping of semantic mash-ups through semantic web pipes. In: Proceedings of the 18th International Conference on World Wide Web (WWW 2009) (2009)

[2] Huynh, D.F., Karger, D.R., Miller, R.C.: Exhibit: lightweight structured data publishing. In: Proceedings of the 16th International Conference on World Wide Web (WWW 2007), pp. 737–746. ACM, New York (2007)

[3] Huynh, D.F., Miller, R.C., Karger, D.R.: Potluck: Data Mash-Up Tool for Casual Users. In: Aberer, K., et al. (eds.) ISWC/ASWC 2007. LNCS, vol. 4825, pp. 239–252. Springer, Heidelberg (2007)

[4] Liu, D., Li, N., Pedrinaci, C., Kopecký, J., Maleshkova, M., Domingue, J.: An approach to construct dynamic service mashups using lightweight semantics. In: Harth, A., Koch, N. (eds.) ICWE 2011. LNCS, vol. 7059, pp. 13–24. Springer, Heidelberg (2012)

[5] Ennals, R., Brewer, E., Garofalakis, M., Shadle, M., Gandhi, P.: Intel Mash Maker: join the web. SIGMOD Rec. 36(4), 27–33 (2007)

[6] Sahoo, S.S., Bodenreider, O., Rutter, J.L., Skinner, K.J., Sheth, A.P.: An ontology-driven semantic mashup of gene and biological pathway information: Application to the domain of nicotine dependence. J. of Biomedical Informatics 41(5), 752–765 (2008)

[7] Heath, T., Bizer, C.: Linked Data: Evolving the Web into a Global Data Space, 1st edn. Synthesis Lectures on the Semantic Web: Theory and Technology, vol. 1(1), pp. 1–136. Morgan & Claypool (2011)

[8] Cano, A.E., Dadzie, A.-S., Ciravegna, F.: Travel Mashups. In: Semantic Mashups (2013)

[9] Dell'Aglio, D., Celino, I., Della Valle, E.: Urban Mashups. In: Semantic Mashups (2013)

[10] Sosins, A., Zviedris, M.: Mashups for the Emergency Management Domain. In: Semantic Mashups (2013)

[11] Kenda, K., Fortuna, C., Moraru, A., Mladenić, D., Fortuna, B., Grobelnik, M.: Mashups for the Web of Things. In: Semantic Mashups (2013)

[12] Héder, M., Solt, I.: DBpedia Mashups. In: Semantic Mashups (2013)

[13] Papadakis, I., Apostolatos, I.: Mashups for Web Search Engines. In: Semantic Mashups (2013)

[14] Huynh, D.F., Karger, D.R., Miller, R.C.: Exhibit: lightweight structured data publishing. In: Proceedings of the 16th International Conference on World Wide Web (WWW 2007), pp. 737–746. ACM, New York (2007)

[15] Quan, D., Huynh, D., Karger, D.R.: Haystack: A Platform for Authoring End User Semantic Web Applications. In: Fensel, D., Sycara, K., Mylopoulos, J. (eds.) ISWC 2003. LNCS, vol. 2870, pp. 738–753. Springer, Heidelberg (2003)

[16] Endres-Niggemeyer, B.: Semantic mashups. Springer (2013) ISBN 978-3-642-36403-7

[17] Berners-Lee, T.: Linked Data - Design Issues (2006), http://www.w3.org/DesignIssues/LinkedData.html