The use of Linked Data technologies in practice

Jelena Jovanovic

jeljov@gmail.com

Overview

- Common applications cases for Linked Data (LD) technologies
- General LD Application Architecture
- Summary: the kinds of problems LD technologies should be used for

Common applications cases for LD technologies

Agile data integration in large business/ government (eco)systems

- USA Ministry of Defense: connecting various information systems that are under the Ministry’s jurisdiction
- Pharmaceutical company Biogen Idec: data integration within supply-chain management (> 30 companies)
- Chevron: integration of (oil- and gas-related) data from various sources with heterogeneous data formats
- Amdocs: improving customer experience by integrating (then analyzing) customer data; specifically targets ind. branches with large customer base (e.g., telecoms, health care)

Content annotation, classification and search

- Libraries, museums, archives and other institutions with huge collections of scientific and artwork
  - Library of Congress
  - British Library
  - ZBW German National Library of Economics
  - Amsterdam Museum
  - Europeana (European cultural heritage)
  - …

Dynamic Semantic Publishing

- BBC
  - Since 2009, BBC has been developing and using its LD publishing platform (slides)
  - In 2010, their entire World Cup website was powered by LD technologies (article)
  - In 2012, BBC further developed its LD platform to power its 2012 Summer Olympics website (article)
  - LD platform is still in ongoing development and use (article)

Dynamic Semantic Publishing

- LD technologies are used in production systems of other well known media companies and organizations:
  - Time Inc., Financial Times, Gannet Inc., Elsevier, the Library of Congress
- Rational for employing LD technologies
  - huge amount of unstructured content
  - diverse and abundant cross-content relationships and annotations (metadata)
  - both content and links keep changing

Search, discovery, recommendations

- Seevl
  - provides search, discovery and exploration of music entities and their connections
  - pulls data from multiple sources: MusicBrainz, Wikipedia, Freebase, BBC, NY Times
  - unifies the data as LD using Music Ontology, to build a giant graph of music entities
  - uses knowledge graph to recommend artists and provide explanations for the recommendations
  - offers RESTful access to its knowledge graph – see: https://developer.seevl.fm/

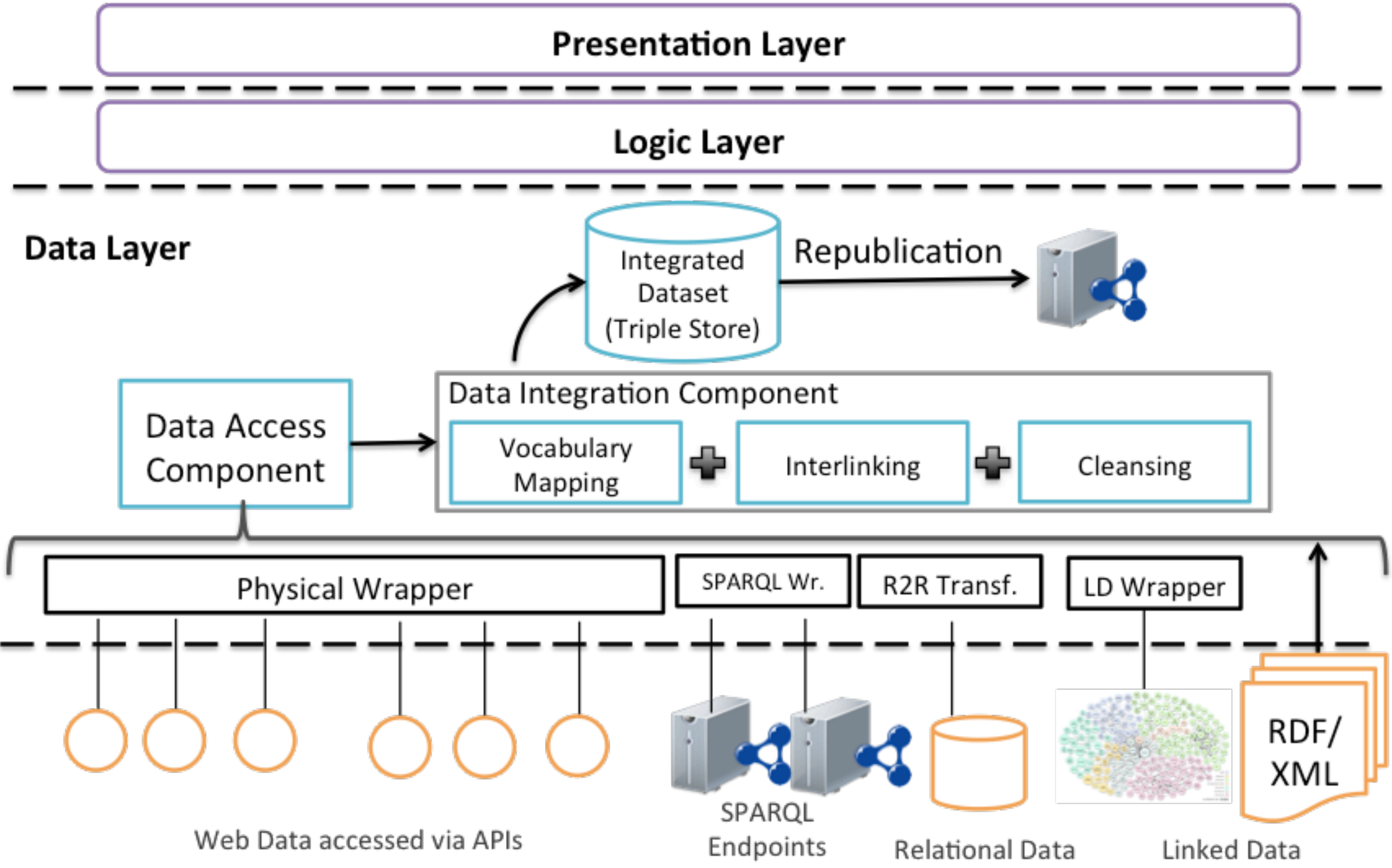
Advancing e-commerce

- Benefits for online retailers
  - Rich/interactive snippets tend to lead to better visibility in search engine results and higher click through rates
  - Better position and higher visibility in vertical search engine results pages (e.g., Google Shopping)
  - Seamless, just-in-time introduction of occasion-specific product categories (e.g., the case of BestBuy)

Advancing e-commerce

- Benefits for online shoppers
  - ▫ Search, discovery and recommendation of niche and long tail products
  - ▫ Faceted product search at the Web scale
  - ▫ Better, more personalized, product recommendation
    - based on the product-specific features and/or user’s shopping history

General LD Application Architecture



Presentation Layer

Logic Layer

Data Layer

Integrated Dataset (Triple Store)

Republication

Data Access Component

Data Integration Component

Vocabulary Mapping

Interlinking

Cleansing

Physical Wrapper

SPARQL Wr.

R2R Transf.

LD Wrapper

RDF/ XML

Web Data accessed via APIs

SPARQL Endpoints

Relational Data

Linked Data

Source: http://www.euclid-project.eu/sites/default/files/images/chapter5_image033b.png

Summary: kinds of problems LD technologies should be used for

Open ended problems

- **Data model is not entirely known**
    - you are not fully sure if you'll need to extend your data model at some point in the (near) future
- **Usage model is not entirely known**
    - you are not fully sure that your users have provided you with all the kinds of views/reports they might need
- **User base is not entirely known**
    - you are not sure if your system would need to be extended to other categories of users
    - adding new users means *new kinds of data* and *new usages of data*

Open ended problems

In a nutshell:

the more open-ended a problem is,

the more beneficial LD technologies will be

Additional kinds of problems LD technologies are suitable for

- The requirement to integrate data from heterogeneous sources using open standards
  - frequent in company mergers and acquisitions
  - LD technologies allow for explicity defining the context of and rationale for the data mapping needed for integration

> "*Whenever you integrate content, you have the same problem: the integration entails mapping data to data and does not capture any intrinsic understanding of the process or context of how the data is related. It's essentially dumb mapping.*" [1]

[1] http://www.javaworld.com/javaworld/jw-03-2012/120326-modernizing-it.html

Additional kinds of problems LD technologies are suitable for

- The need to work with unstructured content
  - reports and other kinds of office documents,
  - Web pages,
  - news articles,
  - scientific publications,
  - …

When LD technologies are *not* the best solution

- Large scale data (Big Data)
  - still not as efficient as more traditional solutions
  - NoSQL datastores and related technologies are currently used to solve this issue
- High volume transactional applications
  - couple of thousands of read/write requests per second to a single server
- Large scale computations
  - high-scale numeric computations on a huge amount (terabytes) of numeric data

Recommendation:

Panel: Semantic Technologies for Big Data Analytics - Challenges and Opportunities

video available at:

http://videolectures.net/eswc2013_montpellier/

Combine the best of both worlds

- LD applications are storage agnostic, and can be easily combined with other technologies
  - E.g., a relational database that is used as the high-volume transactional server can be wrapped as a SPARQL endpoint and thus integrated in a broader LD application
  - Or, a subset of data can be pulled from a large scale data warehouse, and mapped to the chosen vocabulary/ ontology to be used in a LD app

To sum up:
when to consider using LD technologies

- Your use case(s) involve documents and other forms of unstructured data
- You expect to add more kinds of data in the (near) future
- You expect to add more views on the data in the (near) future
- You expect to expand your application to serve more kinds of users in the (near) future
- The data scale is less than petabytes
- The transaction volume is modest (i.e., hundreds versus thousands of transactions per second)
- Your application require only modest numeric calculations

Source: http://www.cambridgesemantics.com/semantic-university/what-makes-good-semantic-web-application