Case Study on Linked Data and SPARQL Usage for Web Application Development

There are two web application architectures — **1.** using a relational database, creating an application-specific schema, storing data closed in such DB, presenting information in HTML with no semantics of data, and offering export in other somewhat semantic formats if needed, or **2.** using an RDF store as backend, choosing commonly adopted schemas to express data, presenting information as Linked Data and in XHTML+RDFa to keep data semantics, and exposing the store using a SPARQL enpoint.

Josef Petrák

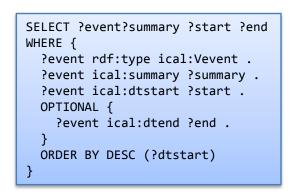
Laboratory of Intelligent Systems (LISp)
University of Economics, Prague me@jspetrak.name

Jan Zemánek, Vojtěch Svátek

Department of Information and Knowledge Engineering University of Economics, Prague jan.zemanek@gmail.com, svatek@vse.cz

Linked Data – every object or resource has its URI in order to be identified. *Information resources* have their Web representation (pages, e-messages, files), but *non-information resources* represent real-world objects. Wikipedia (in the form of DBPedia) and other sources are presented as Linked Data, to be reused.

Syntax – RDF is syntax-independed. Data are expressed as a graph of triples. To exchange information, there are various syntaxes: RDF/XML, Notation 3, RDF/JSON or RDFa for encoding RDF into (X)HTML pages.





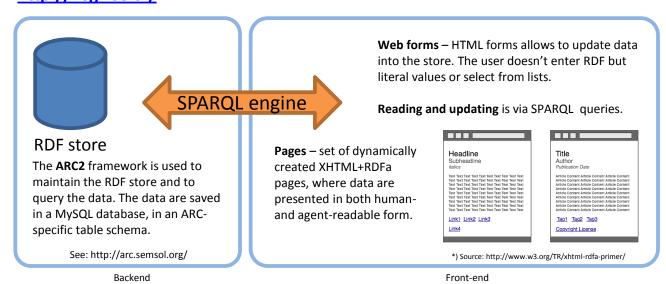


RDF/N3 syntax

Representation – information resources

Research group portal architecture

http://keg.vse.cz/



have their web representation and they are stored in files of various formats. Non-information resources representing real-world objects do not have such a representation. Linked Data are assumed to inform any web agent that the resource does not have a web representation, and point the agent to some representation, ideally a page about that resource.

Semantics —when presenting information in RDF on a web page, the semantic is lost — a web agent understands data as a bundle of headers and paragraphs. A human can still understand the semantic of information. To preserve the semantics for agents, RDFa atributes are used to annotate the web page and to losslessly store information from the original RDF.

Interlinking – Information about resources may be present in more then one RDF store. And what more, they may have different URIs. To share and exchange information, it is necessary to map source URIs using *aliases*. To create one, we use the owl:sameAs property. Then we can ask other stores for additional information.

Example: (see below) a person from a research group has publications in proceedings described in the DBLP store.

```
SELECT ?person ?name ?alias
WHERE {
    ?person rdt:type foaf:Person .
    ?person foaf:name ?name .
    ?person owl:sameAs ?alias .
}
```

Local query: get person information and alias.

Remote query: get information about publications using an alias of a person.

Some of schemas/ontologies widely adopted and considered as near-standards. Using such schemas increases the ability to share data, as other applications can be assumed to understand them. We chose FOAF for the description of people relations, SIOC for interlinked communities, **DOAP** software and artifact projects, **Dublin Core** for library data and RDF Calendar for events and tasks.