Integrating Provenance into the Web of Data

Olaf Hartig and Jun Zhao

Our main motivation is to enable provenance-based information quality assessment of Linked Data. This can only be achieved when provenance-related metadata is an integral part of the Web of data. Therefore, we aim to provide the necessary means to describe provenance of Linked Data and to make these descriptions available as a Linked Data again. Our contributions are:

- A vocabulary to describe provenance of Linked Data
- Automatic metadata generation in publishing tools

The Provenance Vocabulary

| Overview |
|-----------------|-----------------|
| • Suitable to provide general descriptions of all cases of Linked Data publishing |
| • Defined as an OWL ontology |
| • Partitioned into: |
|  - Core ontology |
|  - Supplementary modules (types, integrity verification) |

Design Principles

- Usability and understandability
- No specific granularity prescribed
e.g. a prv:DataItem can be a whole linked dataset, an RDF graph, or a single triple
- More detailed descriptions based on other vocabularies can be added easily
- Provide schema-level links to related vocabularies
e.g. prv:Actor owl:equivalentClass foaf:Agent

Example

<http://example.org/triplify> a prv:NonHumanActor ;
  rdfs:comment "Instance of Triplify V0.5" ;
  prv:operatedBy <http://olafhartig.de/foaf.rdf#olaf> .

_:a a prvTypes:TriplifyConfiguration , prv:CreationGuideline;
  prv:createdBy [
    a prv:DataCreation ;
    prv:performedBy <http://olafhartig.de/foaf.rdf#olaf> ] .

This example describes the provenance of a data item representing data about a specific gene. The data item was created by the execution of a process, performed at a given time (Mar. 1, 2010) and performed by a specific Triplify service. This service was operated by Olaf, identified by the URI http://olafhartig.de/foaf.rdf#olaf, who also provided the configuration file used by the service. The data item was created from a source data item. During the creation the Triplify service retrieved this source data item by accessing a public JDBC access point.