Linked Data Provenance: State of the Art and Challenges

Sarawat Anam[1,2], Byeong Ho Kang[1], Yang Sok Kim[1] and Qing Liu[2]

[1]{Sarawat.Anam, Byeong.Kang, YangSok.Kim}@utas.edu.au
School of Computing and Information Systems, University of Tasmania
Sandy Bay, Hobart, Tasmania, Australia
[2]Q.Liu@csiro.au
Autonomous Systems, Digital Productivity and Service Flagship,
CSIRO Computational Informatics, Hobart, Tasmania, Australia

Abstract

Linked Open Data (LOD) is rapidly emerging in publishing and sharing structured data over the semantic web using URIs and RDF in many application domains such as fisheries, health, environment, education and agriculture. Since different schemas that have the same semantics are found in different datasets of the LOD Cloud, the problem of managing semantic heterogeneity among the schemas is increasing. Schema level mapping among the datasets of the LOD Cloud is necessary as instance level mapping among the datasets is not feasible in the process of making knowledge discovery easy and systematic. In order to correctly interpret query results over the integrated dataset, schema level mapping provenance is necessary. In this paper, we review existing approaches of linked data provenance representation, storage and querying, and applications of linked data provenance where mapping is at the instance level. The analysis of existing approaches will assist us in revealing open research problems in the area of linked data provenance where mapping is at the schema level. Furthermore, we explain how schema level mapping provenance in linked data can be used to facilitate data integration and data mining, and also to ensure quality and trust in data.

Keywords: Semantic web, linked data, schema level mapping, mapping provenance, and information extraction.

1 Introduction

Linked Open Data (LOD) is rapidly emerging for publishing and sharing structured data over the semantic web (Berners-Lee et al., 2001) using URIs and RDF based on Tim Berners' Lee's four principles (Bizer et al., 2009). Recently, large amounts of data are available as linked data in various domains such as health, publication, agriculture, and music where mappings between concepts of different datasets are at the instance level. Instance level mapping is defined as the mapping between data elements. For example, HTTP URI http://aims.fao.org/aos/agrovoc/c_12332 is an instance of AGROVOC[1] vocabulary and another HTTP URI http://eurovoc.europa.eu/1744 is an instance of EUROVOC[2] vocabulary and both URIs represent the same literal label "Maize". Mapping between two instances is represented by owl:sameAs which is an OWL predicate used to declare that two instances of different datasets denote one and the same thing. Therefore, users can get information from both datasets using the URI of either AGROVOC or EUROVOC.

In order to benefit both the Artificial Intelligence and Semantic Web Communities, mapping among the datasets is necessary for some applications such as querying, reasoning, data integration, data mining and knowledge discovery (Jain et al., 2010b). These applications are not feasible if mappings between the datasets are at the instance level as instance level mapping has limitations such as lack of expressivity, schema heterogeneity, entity disambiguation, and ranking of results (Jain et al., 2010b). The problems can be solved by mapping the datasets at the schema level. Schema level mapping is done between source schema and target schema. Schema level (class and property) mapping can be published by OWL[3] and RDF Schema[4] where OWL provides properties such as *owl:equivalentClass* and *owl:equivalentProperty*, and RDF Schema provides properties *rdfs:subClassOf* and *rdfs:subPropertyOf*. In order to extract data from data sources that use a specific term, property (schema level) mapping in Linked data is necessary. For example, The HTTP URI http://dbpedia.org/ontology/City and another HTTP URI http://linkedgeodata.org/ontology/City are schemas of DBPedia[5] vocabulary and LinkedGeoData[6] vocabulary respectively and both represent the same literal label "City". Mapping between schemas is represented by owl:equivalentProperty which is an OWL predicate used to declare that two schemas of different datasets denote one and the same thing. There is previous research on mapping concepts at the schema level. Jain et al. (2010a) developed a system, BLOOMS, that aligns schemas of Wikipedia and Wikipedia category hierarchy. In the system, links are generated between class hierarchies (taxonomies), which are rdfs:subClassOf relations. Auer et al. (2009) have completed both schema and instance

[1] http://aims.fao.org/standards/agrovoc/
[2] http://eurovoc.europa.eu/drupal/?q=abouteurovoc
[3] http://www.w3.org/TR/owl-features/
[4] http://www.w3.org/TR/rdf-schema/
[5] dbpedia.org
[6] linkedgeodata.org

level mappings between DBPedia and Linked Geo datasets for data integration and aggregation.

However, schema level mapping among different datasets of LOD Cloud by itself will not help to make knowledge discovery easy and systematic. This is because if the schemas are mapped and the mapping information is not stored, then users need to browse the datasets up to schema level to know about the schemas that are mapped. The problem can be solved by storing the mapping information for further reuse. The preservation of mapping information is called mapping provenance (Velegrakis et al., 2005) where provenance (Cheney et al., 2009) is defined as a term which provides information about a source or a derivation history. Provenance information helps applications to interpret some queries such as who creates the mapping, how the mapping is derived from diverse sources, from where mapping is derived, why the mapping is acquired, and when the mapping is performed.

In this paper, we survey the existing provenance techniques in the context of Linked Data in terms of provenance metadata representation, storage and query. We distinguish linked data provenance by dividing into two levels: instance level and schema level. After surveying the literature of linked data provenance, we find that linked data provenance has been mostly computed at the instance level. We also find that various provenance techniques are used for representing, storing and querying the instance level mapping provenance of Linked Data in order to assess quality and trust. In the literature, there is very little research that describe provenance of schema level mapping in linked data. In this research, we describe how schema level mapping information can be represented, stored and queried by the existing provenance representation techniques. We conclude with some open research problems based on the usage of schema level mapping provenance of linked data in areas such as data integration, data mining, quality assessment and trustworthiness of data.

2 Basic Definitions

2.1 Resource Description Framework (RDF)

RDF consists of a number of triples. Each triple contains three parts in the form <*s*,*p*,*o*> where *s*, *p* and *o* denote subject, predicate and object respectively. The triples are represented by URIs. Object can also be a literal value. RDF is used to create structured data which form the data source, *D*. *D* is typically represented by a directed labelled graph, g. The edges, e of the graph are directed from *s* to *o* (*s*→*o*) and labelled with *p*.

2.2 Linked Data

A RDF graph consists of a set of RDF triples where a predicate represents a relationship between a subject and an object. The definitions of these relationships and classes of entities are represented in vocabularies. The definitions of vocabulary can be represented as RDF data and the vocabularies can be published as linked data (Hartig and Zhao, 2010). Linked data refers to a set of best practices for publishing and sharing structure data over the semantic web according to the four principles of Tim Berners-Lee (Bizer et al., 2009). The fourth principle of linked data says that RDF links are included in a RDF graph which points to RDF data from other data sources on the web. An RDF link is a part of a RDF triple which makes a relationship between a subject and an object where the subject comes from one data source and the object comes from another data source.

The development of a Web of Data, built by applying Linked Data (LD) principles is the frontier of data integration and sharing by creating links between data from different vocabularies. In linked data, vocabularies consist of a large number of entities. Each entity is called a concept. The process of converting vocabulary into linked data is very challenging. The reason behind this is the differences in formats, structure, semantics and concept labels with different languages. Though RDF is a generic data model for describing resources using triples, it does not provide any domain-specific terms for relating classes of things in the world to each other (Bizer, 2011). SKOS (Simple Knowledge Organization System) (Miles et al., 2005) is a standard vocabulary to express thesauri, taxonomies, subject heading systems, and topical hierarchies within RDF, and it is used for converting any source data in to linked data. RDFS (RDF Schema) and OWL (Web Ontology Language) provide vocabularies to describe conceptual models in terms of classes and properties, and these are used for representing subsumption relationships between concepts (for instance doctors are also persons) (Bizer, 2011).

2.3 Mapping

Mapping is a set of logical specifications that express correspondences between semantically related entities of datasets through the application of a matching algorithm. The mapping function is defined as: $M = (E_S, E_t, a, s, r)$, where E_S is a source entity, E_t is a target entity, a is a matching algorithm, s is a similarity measure between entities (ranging from 0 to 1) and r is a relation (e.g., equivalence (=), overlapping (∩), mismatch (⊥), or more general/specific (⊆, ⊇)) holding between E_S and E_t (Shvaiko and Euzenat, 2005).

2.4 Provenance

Provenance is defined as a term which provides the description of the origins of data and the processes by which data are derived and existed in the database (Buneman et al., 2001). Provenance is necessary in order to (1) know the origin of data, (2) trace errors by debugging processes, (3) establish quality, relevance, trust, (4) reuse other's experiment, and know complex transformations. There are many application areas where provenance information needs to be preserved such as scientific computing, data-warehousing, data integration, curated databases, grid-computing and workflow management (Glavic and Alonso, 2009).

2.4.1 Granularity of Provenance

Tan (2007) distinguishes two granularities of provenance – workflow provenance (coarse-grained) and data provenance (fine-grained). Workflow provenance records the metadata about different types of processes and services which take part during execution. Metadata of

the processes and services can be a software program, a hardware and the instruments used for the experiment (Omitola et al., 2010). For example, during examining the provenance information of integrated datasets, users may trust the information if they know what data integration algorithm was used and which datasets were integrated (Omitola et al., 2011). Davidson et al. (2007) provide an overview of tracking and storage of provenance information in scientific workflow systems.

Data provenance stores the origin and derivation history of the data which are transformed at the time of executing a process. This provenance stores the particular features of the original datasets which are combined to produce a feature that are found on the integrated dataset (Omitola et al., 2011). For example, in order to know the values of latitude/longitude of geospatial data, users can find out the original sources from where the values were taken by using provenance information. A particular area of research on data provenance is the provenance in databases which considers the provenance of query results. Buneman et al. (2001) distinguish provenance of databases by two ways: why-provenance and where-provenance; why-provenance refers to the source data that were involved for the existence of the data derived from query result; where-provenance refers to the location in the source databases from where the data of a query result was extracted. Green et al. (2007) introduce how-provenance which describes how the source data were involved in the calculation of a data entity from a query result. Previous research (Simmhan et al., 2005a, Tan, 2007) have been completed in representing provenance of data creation in a DBMS or a workflow management system, but the provenance of data access is not always required for these systems (Hartig and Zhao, 2010). The provenance of both data creation and data access is necessary to be captured for the web of linked data (Hartig and Zhao, 2010).

3 Provenance of Linked Data

Provenance representation and storage are two major challenges of provenance of linked data.

3.1 Provenance Representation

Provenance representation of linked data describes how to represent provenance information using suitable approaches. There are two approaches for representing provenance information: annotation approach and inversion approach (Omitola et al., 2010). **In the annotation approach** (also known as eager approach) (Cheney et al., 2009), metadata of the derivation history of a data (or annotations), descriptions about the source data and the processes are stored. As a consequence, the stored information helps to find out the provenance of the output data, without examining the source data. **In the inversion approach** (also known as lazy approach) (Cheney et al., 2009), extra information or annotation is not carried out to the output data. In this approach, examining the source data, the output data and the transformation derives provenance.

There are some advantages and disadvantages of the above approaches (Cheney et al., 2009). The main advantage of the annotation approach is that it is useful if the source data becomes unavailable after transformation. But the problem of this approach is that it takes more time and space for executing and storing the annotations than inversion approach. As the inversion approach does not use annotations, this approach does not incur any performance or storage overhead when data is transformed from source to target. The disadvantage of the inversion approach is that it cannot compute provenance when source database is unavailable. In computing linked data provenance, the annotation approach is more favourable as it provides richer information of the data and the dataset (Omitola et al., 2010). In order to support the annotation approach, some vocabularies have been used in the available literature to describe the provenance information of the data. In **Section 4**, we will describe all the linked data provenance representation languages.

3.2 Provenance Storage

Storage of provenance information varies according to the level of granularity at which it is collected. If the provenance information is stored according to fine-grained for a big dataset for which provenance is computed at each triple level, then provenance becomes very large which exceeds the actual data size, and it needs large data storage space. For coarse-grained, if the depth of provenance increases, then the size of annotation increases exponentially (Simmhan et al., 2005b). However, it is possible to reduce the storage space by only storing the information which is important for a particular purpose (Omitola et al., 2010). Provenance information can be stored in the same dataset, or in a different location according to tSPARQL (Hartig, 2009). If the provenance information is stored in the same dataset, then the extracting provenance information is not efficient to answer queries, and it also needs large amounts of provenance information to be stored. Provenance information can be stored by itself or with other metadata. Therefore, it is important to decide which storage system will be used to store the provenance information.

4 Provenance Representation Languages

Some linked data provenance representation languages have been proposed in literature. These are described below:

4.1 Vocabulary of Interlinked Datasets (VoID)

Vocabulary of Interlinked Datasets (VoID) (Alexander et al., 2009) is a vocabulary and a set of instructions that provides terms and patterns for describing RDF datasets, and RDF links between datasets. This vocabulary reuses some existing vocabularies in order to store the provenance information. This vocabulary has two main classes: a **dataset** (void: Dataset) is a set of RDF triples (subject, predicate and object) that is published, maintained or aggregated by a single provider; available as RDF; and accessible on the web through dereferenceable HTTP URIs or a SPARQL Endpoint. A **Linkset** (void:Linkset) is a set of RDF triples (subject, predicate and object), which is used to describe that the subject of one dataset is interlinked with the object of

another dataset. In order to express the interlinking between datasets, VoID description states the location of interlinking triples by using void:subset; provides information about source dataset and target dataset by using void:subjectsTarget and void:objectsTarget respectively, and gives RDF links between two datasets using void:linkPredicate.

Some properties of other vocabularies such as Dublin Core[7], FOAF[8] or SCOVO[9] can be reused with VoID.

4.2 Provenance Extension to VoID (VoIDP)

Heath et al. (2008) advise linked data publishers to reuse the existing vocabularies wherever possible. If anyone fails to describe the provenance information of data using the existing vocabularies, then he/she can define new terms. The advantage of reusing the existing vocabulary is that it brings together diverse domains within RDF, and it makes data more reusable. As VoID vocabulary cannot describe queries like "how data were derived, who carried out the transformation, and what processes were used for the transformations?", so VoID is extended to VoIDP (Omitola et al., 2010, Omitola et al., 2011) which has the capability to describe the above queries. VoID provides classes and properties which are designed by reusing existing vocabularies such as Provenance Vocabulary (Hartig and Zhao, 2009), The Time Ontology in OWL (Hobbs and Pan, 2004) and The Semantic Web Publishing Vocabulary (Bizer, 2006). The classes and properties of VoIDP are described by Omitola et al. (2010).

4.3 Provenance Vocabulary for Linked Data

Hartig and Zhao (2010) develop a vocabulary in order to describe provenance of linked data with RDF. They also provide the way of publishing the provenance description as linked data on the web. They define the provenance vocabulary as OWL ontology and partition it into core ontology and supplementary modules such as Types, Files and Integrity Verification. The provenance vocabulary for linked data consists of three parts: general terms, terms for data creation, and terms for data access. Three classes for the general types of provenance elements: Actor, Execution and Artifact are included in the general terms. This term consists of some sub-classes and properties, and describes general provenance elements of linked data using RDF. The term, data creation dimension describes how a data item is created. Data access dimension illustrates how to retrieve the data items from the web. Though this vocabulary provides a basic framework to create and access provenance of linked data, but to support every aspect and details of provenance information, it is necessary to use other specialized vocabularies with this vocabulary.

4.4 W3C PROV Ontology

PROV ontology (PROV-O)[10] is a lightweight ontology standardized by the W3C Provenance Working Group. PROV is a specification that expresses provenance record about the description of entities and the activities which are used for the derivation and existence of a given entity. Provenance can be viewed from three different perspectives according to W3C PROV Model Primer[11] such as agent-oriented, object-oriented and process-oriented. **Agent-oriented** provenance focuses on the people or organizations who are involved in generating or manipulating the information in question. **Object-oriented** provenance focuses on tracing the origins of an entity which contributes to the existence of another entity. **Process-oriented** provenance focuses on the actions and steps taken to generate an entity. PROV-O provides classes, properties and restrictions for representing and interchanging provenance information generated in different systems and under different contexts. Three classes of PROV-O are the followings: An *prov:Entity* that may be real or imaginal is a physical, digital and conceptual kind of thing with some fixed aspects. An *prov:Activity* is a process or a service which includes transforming, modifying, or generating an entity over a period of time. An *prov:Agent* takes the responsibility for an activity that occurs, for the existence of an entity, or for another agent's activity.

5 Linked Data Provenance Techniques

We distinguish linked data provenance by dividing into two levels: instance level and schema level mapping.

5.1 Provenance in Instance Level Mapping

Many research works have been done for capturing provenance in linked open data. Patni et al. (2010) develop sensor provenance ontology using the concepts of Provenir upper level ontology defined in PMF (Sahoo et al., 2009a), and the ontology is used for building a framework, call Sensor Provenance Management System (PMS). The system captures, represents and stores provenance of linked open data in sensor domain according to Sahoo et al. (2009b). The system first captures provenance information associated with the sensor by obtaining the time related information from MesoWest (Bizer, 2006) and the location related information by querying GeoNames (Team, 2010) using SPARQL query language. Then the system uses Sensor Provenance ontology for representing the provenance information. After that, Virtuoso RDF store[12] is used for storing the provenance information. In order to find out a sensor and the observation data over time and geographical space, the stored provenance information is queried by SPARQL in two ways: query for provenance metadata to get the provenance information about a data entity, and query for data using provenance information which returns a set of data entity.

Hartig and Zhao (2009) propose an approach for assessing quality of web data using provenance information. They represent two types of provenance information: data creation and data access. The types are classified in three categories: actors, executions and artifacts. An actor performs the execution of an action or a process which produces a specific data item called an

[7] dcterms: <http://purl.org/dc/terms/>
[8] foaf: <http://xmlns.com/foaf/0.1/>
[9] scovo: <http://purl.org/NET/scovo#>

[10] http://www.w3.org/TR/2013/NOTE-prov-overview-20130430/

[11] http://www.w3.org/TR/2013/NOTE-prov-primer-20130430/
[12] http://www.openlinksw.com/

artifact. The artifact considers timeliness as quality criteria and use provenance information by quantitative method for assessing quality of data. The assessment approach takes three steps. First, elements of provenance information are collected. Second, the influences of these elements are decided. Finally, quality of data is calculated by applying a function. Later the quality value is associated with certainty values in order to deal with missing provenance information.

Hartig and Zhao (2010) describe an approach for integrating provenance information of data creation and data access in to web of linked data. For doing this, they use VoID (Alexander et al., 2009) to represent general provenance information for the described datasets, and develop a provenance vocabulary for representing detailed provenance information of linked data. In order to access the linked dataset on the web, they consider three aspects: (1) adding provenance to linked data objects, (2) adding provenance to RDF dumps, and (3) providing provenance information at SPARQL Endpoints so that a query service can execute SPARQL queries over the dataset. They also extend several linked data publishing tools such as Triplify[13], Pubby[14] and D2R server[15] for publishing the provenance metadata (Hartig et al., 2010). They examine two databases: FlyBase and FlyTED, and create three linked datasets from the two datasets and publish their provenance information using provenance vocabulary and VoID. They also map the linked datasets at the instance level and express their mapping using owl:sameAs link predicate. Then they demonstrate quality and trustworthiness of linked data by using timeliness criteria. They calculate trust by assessing quality using only fine-grained provenance. However, for a big dataset that contains a large number of triples, encoding fine-grained provenance at the triple level occurs much more than actual data (Omitola et al., 2010). It is possible to reduce the storage space by only storing the important information for a particular purpose. In addition, it is necessary to calculate coarse-grained provenance of the integrated dataset.

Zhao et al. (2009) maintain data links between related data items from heterogeneous biological linked data sources. They then capture provenance information about why data items of different sources are linked with each other, how each data link is evolved, when the linked is created, who creates the link, which version of the databases are used, and when the link is updated. For this, they use named graphs to make a provenance statement about the linked data. In order to represent provenance information, they use existing vocabularies such as Dublin Core[7] and dw namespace[16]. By using RDF named graphs and the RDF query language SPARQL, they analyse that trust can be brought to the data web by providing evidence for links, or tracing how the data links are updated and maintained.

Carroll et al. (2005) serialize a linked dataset as a collection of Named Graphs i.e. RDF graphs named with a URI. In this case, each of these graphs could contain provenance metadata about itself. The provenance in linked data is used for calculating trust. Here trust is calculated based on the content of the graphs and the users' task, rather than the users themselves. Named Graphs provide greater precision and potential interoperability as it has a clearly defined abstract syntax and formal semantics. The collection of Named Graphs could contain an additional Named Graph that describes the provenance of the other graphs. However, The Named Graph framework has some limitations. It may contain a few triples or many. Therefore, it does not give a good control on the granularity of the collection of data items in order to attach provenance (Omitola et al., 2010).

Hartig (2008) develops a trust model and trust assessment methods in order to assess the trustworthiness of RDF data on the Web. The trust model defines trust values for representing trustworthiness of RDF data on a statement level. Here trustworthiness of RDF statements is calculated based on a trust value which is unknown or a value in the interval [-1, 1]. The trust values 1,-1 and 0 represent belief, disbelief and lack of belief or disbelief respectively. For assigning subjective trust values in every statement, a trust function is defined that represents the trustworthiness of the statement specific to an information consumer. Besides, a trust aggregation function is developed for calculating trust value for a set of related RDF statements. The trust function is implemented by provenance-based and opinion-based methods. Then a trust aware query language, tSPARQL (Hartig, 2009) is developed which adds TRUST AS and ENSURE TRUST clauses. These two clauses are used to determine trust requirements and to query the trustworthiness of RDF data.

Theoharis et al. (2011) develop data provenance models for Semantic Web data. They discuss implicit provenance information of SPARQL queries in order to compute annotations reflecting various dimensions of data quality such as trustworthiness, reputation and reliability. Here the authors prove that abstract provenance models for the relational data model can be leveraged for positive SPARQL queries over RDF data. They also find out some limitations of abstract provenance models in capturing the semantics of the SPARQL OPTIONAL operator that implicitly introduces negation.

Hartig et al. (2009) develop an approach in order to execute SPARQL queries over the Web of Linked Data. The approach traverses RDF links to discover data that is relevant for a query during the query execution itself. The approach has some limitations such as the retrieval of unforeseeable large RDF graphs from the Web.

5.2 Provenance in Schema Level Mapping and Challenges

In this section, at the conceptual level, we describe schema level mapping provenance representation, storage and querying, and future applications of provenance based on the architecture of **Fig. 1**.

[13] Triplify: http://example.org/triplify
[14] Pubby: http://www4.wiwiss.fu-berlin.de/pubby/
[15] D2R server: http://www4.wiwiss.fu-berlin.de/bizer/d2r-server/
[16] http://www.datawebs.net/

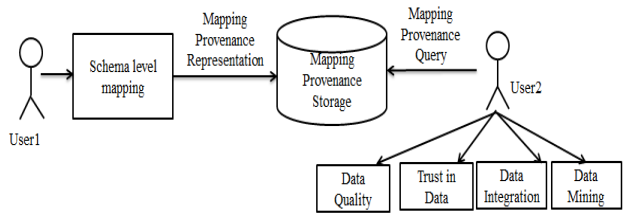

Fig. 1. Architecture of proposed schema level mapping provenance

In **Fig. 1**, it is said that user1 performs schema level mapping in Linked Data. He/she then represents and stores the mapping provenance using suitable provenance languages and storage system respectively. User2 will query the mapping provenance and retrieve data in order to do some applications such as data integration, data mining, quality assessment and trustworthiness of data.

Bizer and Schultz design *R2R Mapping Language* (Bizer and Schultz, 2010) for publishing and discovering dataset-level and vocabulary-level mappings in linked data. The language can only support fine-grained, self-contained term mappings, and it does not consider coarse-grained mapping information for publishing. But in our research, we consider schema level mapping provenance information at both granularities (Tan, 2007)– coarse-grained (workflow provenance) and fine-grained (data provenance) so that extracting provenance information becomes efficient to answer queries. At the coarse level, it is necessary to record the metadata about the different types of processes and services which take part during execution in order to increase trustworthiness. For example, if users are aware of some information such as the datasets that are mapped, the algorithm which is used for mapping, the human agent who operates the mapping, and the time when the mapping is created, then they may trust the information as trustworthiness comes by disclosing as much information as possible (Omitola et al., 2010). Besides, it is necessary to store the particular features of datasets as fine-grained provenance. For instance, in order to reuse provenance information, users may want to get some information such as the schemas of the datasets that are mapped and the linkPredicate which is used for mapping.

We define schema level mapping provenance as: $MP = (D_s, D_t, E_s, E_t, L_p, M_s, A_p)$, where D_S is a source dataset, D_t is a target dataset, E_S is a source schema, E_t is a target schema, L_p is a link predicate which is owl:equivalentProperty, M_s is a mapping system and A_p is the additional provenance information such as the human agent who drives the mapping system for mapping schemas and the time when the mapping is performed.

In the following, we sketch the way of using the existing provenance languages for representing schema level mapping provenance of the datasets. We then provide the way of storing provenance information in a separate location for making knowledge discovery easy and systematic without browsing the dataset individually. We define queries to extract mapping provenance information and also to provide necessary and sufficient knowledge of the original data sources for data extraction where query results may be derived from multiple mapped datasets.

Use Cases

At the conceptual level, we take two datasets DBPedia and LinkedGeoData. In computing linked data provenance, we choose annotation approach as it provides richer information of the data and the dataset (Omitola et al., 2010). In order to support annotation method, we use W3C PROV vocabulary for representing the provenance information. Another vocabulary, FOAF[17] which links people and information using the Web, is also used with PROV-O. Finally, we represent and store schema level mapping information as provenance in TURTLE format in the following way:

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix dbp: <http://dbpedia.org/ontology/> .
@prefix lgd: <http://linkedgeodata.org/ontology/> .
@prefix :     <http://example.org#> .
<>
  a prov:Bundle, prov:Entity;
  prov:wasAttributedTo :sarawat ;
  prov:wasDerivedFrom <http://dbpedia.org/>,
                      <http://linkedgeodata.org/>;
  prov:wasAssociatedWith :BLOOMS ;
  prov:generatedAtTime "2014-07-03 20:01:08"^^xsd:dateTime  .

:sarawat
  a foaf:Person, prov:Agent;
  foaf:givenName "Sarawat"^^xsd:string;
  foaf:mbox <mailto:sarawat.anam@utas.edu.au> .

:BLOOMS
  a prov:SoftwareAgent;
  rdfs:label "BLOOMS"^^xsd:string .

dbp:City  owl:equivalentProperty  lgd:city .
dbp:Country  owl:equivalentProperty  lgd:country .
dbp:Airport  owl:equivalentProperty  lgd:aerodrome .
dbp:River  owl:equivalentProperty  lgd:waterway .
dbp:LightHouse  owl:equivalentProperty  lgd:lighthouse .
dbp:Stadium  owl:equivalentProperty  lgd:stadium .
```

Fig. 2. Mapping Provenance of DBPedia and LinkedGeoData

In our research, in **Fig. 2**, we use the following classes and properties of PROV Vocabulary for representing provenance. *prov:Bundle is* an Entity and is a named set of provenance descriptions. *prov:Entity* that may be real or imaginal is a physical, digital and conceptual kind of thing with some fixed aspects. A *prov:Agent* takes the responsibility for an activity that occurs, for the existence of an entity, or for another agent's activity. The property *prov:wasAssociatedWith* is used to describe an Agent's responsibility for an Activity, and this property is used to provide information about the *BLOOMS System (Jain et al., 2010a). prov:SoftwareAgent* is a software agent that runs the mapping system. The property *prov:wasAttributedTo* is used to describe an Agent's responsibility for an Entity. We use *foaf:givenName* to describe the name of the human Entity who performs the mapping. *prov:wasDerivedFrom* is used to provide information about the datasets that are used for mapping. *prov:generatedAtTime* is used to provide the information

[17] foaf: http://xmlns.com/foaf/0.1/

of time (*2014-07-03 20:01:08*) when the mapping between the datasets is completed.

The rest of the information describes which properties of DBPedia are mapped to the properties of LinkedGeoData. Here, the mapping between the properties is expressed by owl:equivalentProperty which declares that two properties of different datasets denote one and the same thing. The properties *City, Country, Airport, River, Lighthouse and Stadium* of DBPedia are mapped to the properties *city, country, aerodrome, waterway, lighthouse and stadium* of LinkedGeoData respectively.

Then we store the provenance information in a separate RDF storage system in order to access and retrieve the information by SPARQL query. Mapping provenance will help users to decide which and how many properties they can select from the mapped datasets, and can retrieve data under the selected properties and use the data for some applications such as data mining and data integration, quality assessment and trustworthiness of data. In order to help retrieving information from mapping provenance file and original sources, we define the following queries using SPARQL query language to retrieve information from mapping provenance file.

Query1: *SPARQL query that asks for retrieving the properties which are mapped from two datasets.*

```
prefix owl: <http://www.w3.org/2002/07/owl#>
select ?s ?o where  {?s owl:equivalentProperty ?o}
```

Query2: *SPARQL query that asks for extracting the name of the datasets which are mapped, the non-human agent that performed the mapping, the human agent who performed the mapping and the completion time of mapping.*

```
prefix prov: <http://www.w3.org/ns/prov#>
select ?datasetName ?system  ?humanAgent  ?time
where  {?s  prov:wasDerivedFrom  ?datasetName;
    prov:wasAssociatedWith ?system;
   prov:wasAttributedTo ?humanAgent;
    prov:generatedAtTime ?time}
```

Getting the property names from mapping provenance file using the above queries, users can extract data from original data sources using the following query:

Query3: *SPARQL query that asks for extracting data from original data sources based on selected properties*.

```
prefix dbp: < dbpedia.org/ontology/>
select distinct ?uri ?uri2
where {?uri rdf:type schema:Country .
          ?uri2 rdf:type schema:Airport }
```

Now we provide some examples of how to extract data from mapping provenance file and original sources using local SPARQL Endpoint named TWINKLE[18] and Virtuoso SPARQL Endpoint respectively.

- An example of extracting data from provenance file is given in **Fig. 3**.

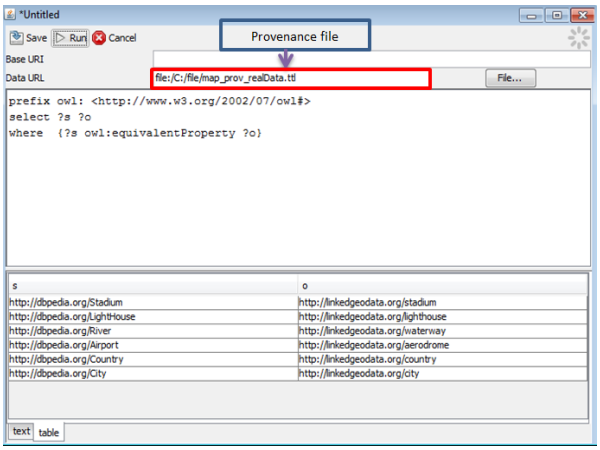

Fig. 3. Provenance data retrieval

- An example of extracting data from original data source, DBPedia using SPARQL Endpoint, *http://dbpedia.org/sparql* is given in **Fig. 4.**

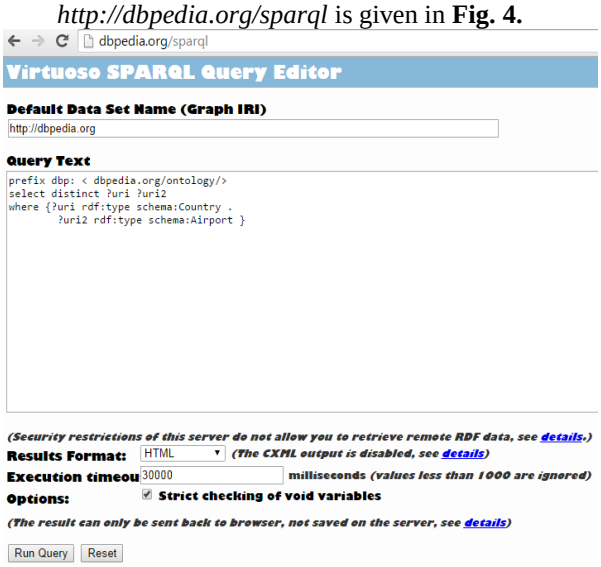

Fig. 4. Data retrieval from DBPedia

Like **Fig. 4,** it is also possible to extract data from the original LinkedGeoData using SPARQL Endpoint, *http://linkedgeodata.org/sparql*. After retrieving data based on mapping provenance using the methods described above, data can be used in the following applications:

- **Quality of Linked Data**

Data quality is an essential characteristic that determines the reliability of data by assessing criteria such as accuracy, completeness, believability and timeliness (Hartig and Zhao, 2009). When large amounts of linked data come from various sources, then users of linked data may face the danger of poor quality data which might contain wrong information. Instance level mapping provenance at the fine-grained has been used to identify outdated information by comparing genes timeliness (Hartig and Zhao, 2010). However, for a big dataset, computing provenance at the instance level may lead to the provenance information be much more than the actual data. The problem can be solved by computing particular provenance information at both granularities: fine-grained and coarse-grained where mapping among datasets is in

[18] http://www.ldodds.com/projects/twinkle/

the schema level. This schema level mapping provenance helps to reduce potential errors of linked data by assessing quality at the schema level.

- **Trustworthiness of Linked Data**

As large amounts of linked data are available in various sources, so users need to understand the trustworthiness of data in order to use it. Trustworthiness of linked data has been calculated by assessing the quality at each triple level using fine-grained provenance (Hartig and Zhao, 2010). However, it is necessary to calculate both fine-grained and coarse-grained provenance of the integrated dataset where mapping between datasets is at the schema level. This is because users make the judgement of trustworthiness of data on the context of information they see (Artz and Gil, 2007). At the coarse level, recording the provenance metadata about different types of processes and services which take part during execution can be used to increase trustworthiness of linked data. Storing provenance information of the particular features of datasets as fine-grained provenance is also necessary in order to increase the trustworthiness of linked data.

- **Linked Data Integration**

Data integration involves combining data residing at heterogeneous sources and providing users with a unified view of these data (Lenzerini, 2002). In data integration, schema mappings are used to translate queries from a source schema in to a target schema from heterogeneous data sources. As linked data is increasing day by day and semantically same types of schema data are found in different dataset, so schema level mapping in linked data is necessary to combine data from multiple datasets by eliminating redundant data. In this context, schema level mapping provenance helps to get the information of schemas without domain knowledge of the data sources.

- **Linked Data Mining**

As huge amounts of linked data are available in the LOD Cloud, it is necessary to find out hidden patterns and trends such as frequency, rarity, and correlation. Some systems have been developed for mining linked data. LiDDM (Narasimha et al., 2011) is an approach which extracts data from multiple linked data sources such as DBPedia, Linked Movie Database, WorldFactBook, and Data.gov using SPARQL, and integrates data using JOIN operation and mines these data using data mining techniques. Extension of RapidMiner which is called RapidMiner semweb plugin (Khan et al., 2010) retrieves data from semantic web. The system uses all the algorithms which are implemented in RapidMiner for processing the extracted linked data.

However, in the above systems, users need to acquire domain knowledge about the schema names of the datasets by browsing the datasets individually in order to retrieve data under the schemas. The problem can be solved by using schema level mapping provenance which helps to select schemas without browsing the datasets in order to extract data from multiple datasets. The large amounts of data retrieved from multiple datasets will help to increase the performance of data mining by applying data mining algorithms. A summary of linked data provenance is given below:

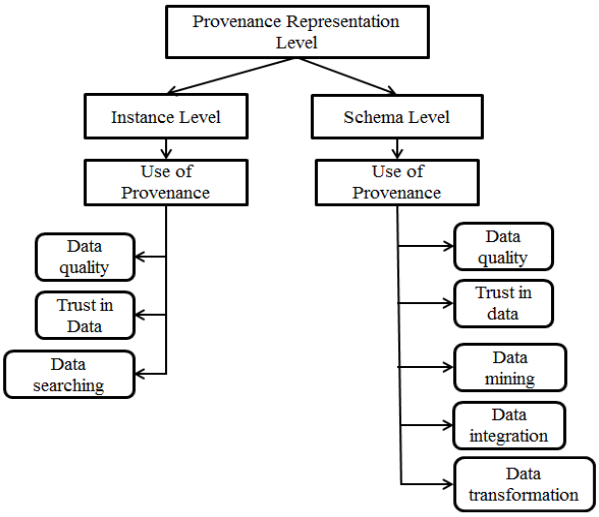

Fig. 5. Provenance representation level and usage of each level

Approaches	Provenance representation level	Granularity of provenance	Provenance representation language	Provenance storage repository	Query	Applications of provenance
Carroll et al.(2005)	Instance	Fine-grained	Named Graphs	RDF	RDFQ	Assessment of trust in data
Hartig (2008)	Instance	Fine-grained	Named Graphs and semantic sitemaps	RDF	tSPARQL	Assessment of trust in data
Hartig and Zhao (2009)	Instance	Fine-grained and coarse-grained	Provenance model	RDF	SPARQL	Assessment of data quality
Zhao et al. (2009)	Instance	Fine-grained	Dublin Core and dw namespace	RDF	SPARQL	Assessment of trust in data
Patni et al. (2010)	Instance	Fine-grained and coarse-grained	Sensor provenance ontology	Virtuoso RDF store	SPARQL	finding out a sensor and observation data over time and geographical space
Hartig and Zhao (2010)	Instance	Fine-grained	Provenance vocabulary	Virtuoso RDF store	SPARQL	Assessment of data quality and trust in data
Theoharis et al.(2011)	Instance	Fine-grained	Abstract provenance models	RDF	SPARQL	Computing trust, reputation and reliability of data
Bizer and Schultz (2010)	Instance and Schema	Fine-grained	R2R Mapping Language	RDF	SPARQL	Data transformation
Proposed approach	Schema	Fine-grained and coarse-grained	W3C PROV Ontology	Virtuoso RDF store	SPARQL	Assessment of data quality, trust in data, data mining and data integration

Table 1: Summary of Instance and schema level mapping provenance techniques

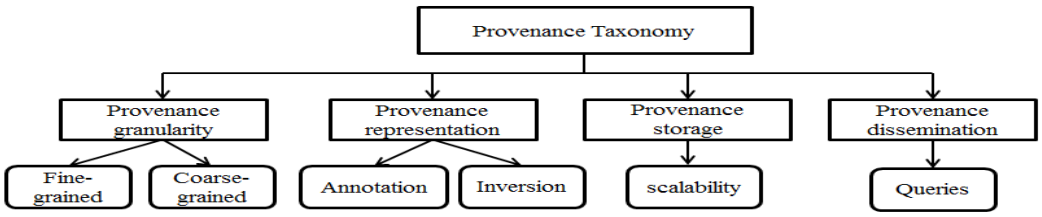

Fig. 6. Provenance Taxonomy

6 Conclusion and Future Works

In this research, we have described a state of the art survey of the current linked data provenance approaches and found some problems of instance level mapping provenance. We have proposed a novel approach of provenance of schema level mapping in linked data and have provided some challenges which can be solved by schema level mapping provenance. At the conceptual level, we have used two datasets DBPedia and LinkedGeoData, and have represented provenance of mapping using suitable provenance languages and stored schemas (properties) mapping information as mapping provenance. We have stored both fine-grained and coarse-grained provenance at the schema level in a separate location. We have also defined queries using SPARQL query language in order to extract provenance information from provenance storage system and data from original sources. In addition, we have shown how to retrieve provenance information using local SPARQL Endpoint and how to extract data from original sources using de-referencable HTTP URI of DBPedia SPARQL Endpoint. In this research, we have only emphasized on the property level mapping because our purpose is to extract data under each property in order to use in some applications. In future, we will compute schema level mapping using datasets from LOD Cloud. Then we will store mapping provenance information in the Virtuoso RDF store[12] and it will have a SPARQL Endpoint which will be accessible by de-referencable HTTP URI in order to query using SPARQL query language. We will extract data based on the provenance information and use the data for ensuring data quality and trustworthiness, doing data mining and data integration.

Acknowledgement

The Intelligent Sensing and Systems Laboratory and the Tasmanian node of the Australian Centre for Broadband Innovation are assisted by a grant from the Tasmanian Government which is administered by the Tasmanian Department of Economic Development, Tourism and the Arts.

References

Alexander, K., Cyganiak, R., Hausenblas, M. and Zhao, J.(2009): Describing Linked Datasets. In *Proceedings of the Second Workshop on Linked Data on the Web, LDOW*.

Artz, D. and Gil, Y. (2007): A survey of trust in computer science and the semantic web. In *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2)**,** 58-71.

Auer, S., Lehmann, J. and Hellmann, S. (2009): Linkedgeodata: Adding a spatial dimension to the web of data. In *The Semantic Web-ISWC*, Springer, 731-746.

Berners-Lee, T., Hendler, J. and Lassila, O. (2001): The semantic web. In *Scientific american*, 284(5)**,** 28-37.

Bizer, C. (2006): Semantic Web Publishing Vocabulary (SWP), User Manual.

Bizer, C. (2011): Evolving the Web into a Global Data Space. In *BNCOD*.

Bizer, C., Heath, T. and Berners-Lee, T. (2009): Linked data-the story so far. In *International Journal on Semantic Web and Information Systems (IJSWIS)*, 5(3)**,** 1-22.

Bizer, C. and Schultz, A. (2010): The R2R Framework: Publishing and Discovering Mappings on the Web. In *COLD*, 665.

Buneman, P., Khanna, S. and Wang-Chiew, T. (2001). Why and where: A characterization of data provenance. In *Database Theory—ICDT*, Springer, 316-330.

Carroll, J.J., Bizer, C., Hayes, P. and Stickler, P. (2005): Named graphs, provenance and trust. In Proceedings of the 14th international conference on World Wide Web, ACM, 613-622.

Cheney, J., Chiticariu, L. and Tan, W.-C. (2009): Provenance in databases: Why, how, and where. In *Foundations and Trends in Databases*, 1(4),379–474.

Davidson, S.B., Boulakia, S.C., Eyal, A., Ludäscher, B., Mcphillips, T.M., Bowers, S., Anand, M.K. and Freire, J. (2007): Provenance in Scientific Workflow Systems. In *IEEE Data Eng. Bull.*, 30(4)**,** 44-50.

Glavic, B. and Alonso, G. (2009): Perm: Processing provenance and data on the same data model through query rewriting. Data Engineering, *In ICDE, IEEE 25th International Conference on*, IEEE, 174-185.

Green, T.J., Karvounarakis, G. and Tannen, V. (2007): Provenance semirings. In *Proceedings of the twenty-sixth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, ACM, 31-40.

Hartig, O. (2008): Trustworthiness of data on the web. In *Proceedings of the STI Berlin & CSW PhD Workshop*.

Hartig, O. (2009): Querying trust in rdf data with tsparql. In *The Semantic Web: Research and Applications*, Springer, 5-20.

Hartig, O., Bizer, C. and Freytag, J.-C. (2009): Executing SPARQL queries over the web of linked data. In *Proceedings of the International Semantic Web Conference*, 293–309.

Hartig, O. and Zhao, J. (2009): Using Web Data Provenance for Quality Assessment. In *international workshop on Semantic Web and Provenance Management*, USA.

Hartig, O. and Zhao, J. (2010): Publishing and consuming provenance metadata on the web of linked data. In *Provenance and Annotation of Data and Processes*, Springer, 78-90.

Hartig, O., Zhao, J. and Mühleisen, H. (2010): Automatic integration of metadata into the web of linked data. In *Proceedings of the Demo Session at the 2nd Workshop on Trust and Privacy on the Social and Semantic Web (SPOT) at ESWC*.

Heath, T., Hausenblas, M., Bizer, C., Cyganiak, R. and Hartig, O. (2008): How to publish linked data on the web. In T*utorial in the 7th International Semantic Web Conference*, Karlsruhe, Germany.

Hobbs, J.R. and Pan, F. (2004): An ontology of time for the semantic web. In *ACM Transactions on Asian Language Information Processing (TALIP)*, 3(1)**,** 66-85.

Jain, P., Hitzler, P., Sheth, A.P., Verma, K. and Yeh, P.Z. (2010a): Ontology alignment for linked open data. In *The Semantic Web–ISWC 2010*. Springer, 402-417.

Jain, P., Hitzler, P., Yeh, P.Z., Verma, K. and Sheth, A.P. (2010b): Linked Data Is Merely More Data. In *AAAI Spring Symposium: Linked Data Meets Artificial Intelligence*.

Khan, M.A., Grimnes, G.A. and Dengel, A. (2010): Two pre-processing operators for improved learning from semanticweb data. In *First RapidMiner Community Meeting And Conference (RCOMM 2010)*.

Lenzerini, M. (2002): Data integration: A theoretical perspective. *In Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, ACM, 233-246.

Miles, A., Matthews, B., Wilson, M. and Brickley, D. (2005): SKOS core: simple knowledge organisation for the web. In *International Conference on Dublin Core and Metadata Applications*, 3-10.

Narasimha, V., Kappara, P., Ichise, R. and Vyas, O. (2011): LiDDM: A Data Mining System for Linked Data. In *Proceedings of the LDOW, LDOW*.

Omitola, T., Gibbins, N. and Shadbolt, N. (2010): Provenance in Linked Data Integration. In *Future Internet Assembly, Ghent, Belgium.*

Omitola, T., Zuo, L., Gutteridge, C., Millard, I.C., Glaser, H., Gibbins, N. and Shadbolt, N. (2011): Tracing the provenance of linked data using VoID. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*, ACM, 17.

Patni, H., Sahoo, S., Henson, C. and Sheth, A.(2010): Provenance aware linked sensor data. In *Proceedings of the Second Workshop on Trust and Privacy on the Social and Semantic Web*.

Sahoo, S., Barga, R., Goldstein, J., Sheth, A. and Thirunarayan, K. (2009a): Where did you come from... Where did you go? In *An Algebra and RDF Query Engine for Provenance Kno. e. sis Center, Wright State University*.

Sahoo, S.S., Weatherly, D.B., Mutharaju, R., Anantharam, P., Sheth, A. and Tarleton, R.L. (2009b): Ontology-driven provenance management in escience: An application in parasite research. In *On the Move to Meaningful Internet Systems: OTM*, Springer, 992-1009.

Shvaiko, P. and Euzenat, J. (2005): A survey of schema-based matching approaches. In *Journal on Data Semantics IV*, Springer, 146-171.

Simmhan, Y.L., Plale, B. and Gannon, D. (2005a): A survey of data provenance in e-science. In *ACM SIGMod Record*, 34(3)**,** 31-36.

Simmhan, Y.L., Plale, B. and Gannon, D. (2005b): A survey of data provenance techniques. In *Computer Science Department, Indiana University, Bloomington IN*, 47405.

Tan, W.C. (2007): Provenance in Databases: Past, Current, and Future. In *IEEE Data Eng. Bull.*, 30 (4)**,** 3-12.

Team, G. (2010). GeoNames, http://www.geonames.org/.

Theoharis, Y., Fundulaki, I., Karvounarakis, G. and Christophides, V. (2011): On provenance of queries on semantic web data. In *Internet Computing, IEEE*, 15(1)**,** 31-39.

Velegrakis, Y., Miller, R.J. and Mylopoulos, J. (2005): Representing and querying data transformations. Data Engineering. In *ICDE 2005, Proceedings, 21st International Conference on, IEEE*, 81-92.

Zhao, J., Miles, A., Klyne, G. and Shotton, D. (2009): Linked data and provenance in biological data webs. In *Briefings in bioinformatics*, 10(2)**,** 139-152.