DC-2013

Tutorial

on

Metadata Provenance

Dr. Kai Eckert

Postdoctoral Researcher
Mannheim University
Data and Web Science Group
Project DM2E (EU funded)
Digitised Manuscripts to Europeana

Member of
W3C Provenance Incubator Group
W3C Provenance Working Group
DCMI Metadata Provenance Task Group
Agenda

09:30  Introduction and agenda

09:40  Foundations

  Introduction to Provenance and Metadata
  RDF and RDFS (very short)
  Metadata (RDF) Provenance (What is the problem?)
Agenda

10:20 Part 2: Identification of RDF Data
   What’s in the standards? A brief review of Reification.
   Linked Metadata (Use the LD Principles)
   Named Graphs
   RDF 1.1

11:00 Coffee Break
Agenda

11:30   Metamodels in Practice
        OAI-ORE
        The Europeana Data Model
        OAI-ORE “vs.” Named Graphs
        Linked Data Publishing with VoID

12:00   Linked Data Provenance
        State-ful or State-less Data
        Versioning
        Identity and Provenance Context
Agenda

12:45   Expressing Provenance in RDF
       Brief outlook to the PROV-O tutorial

13:00   End
Further Readings, Handout

Eckert, Kai
*Metadata Provenance in Europeana and the Semantic Web*

*Berliner Handreichungen zur Bibliotheks- und Informationswissenschaft*
Number 332

Berlin : Institut für Bibliotheks- und Informationswissenschaft
der Humboldt-Universität zu Berlin, 2012

ISSN 14 38-76 62

Available online:

http://edoc.hu-berlin.de/series/berliner-handreichungen/2012-332

Part 1: Foundations
Agenda

Introduction to provenance and metadata
RDF and RDFS
Metadata (RDF) Provenance (What is the problem?)
Provenance

Definition of PROVENANCE

1 : ORIGIN, SOURCE

2 : the history of ownership of a valued object or work of art or literature
   
   See provenance defined for English-language learners »

Examples of PROVENANCE

• Has anyone traced the provenances of these paintings?
• The artifact is of unknown provenance.

Origin of PROVENANCE

French, from provenir to come forth, originate, from Latin provenire, from pro- forth + venire to come — more at PRO-, COME

First Known Use: 1785
Definition: Provenance

Provenance of a resource is a **record** that describes entities and processes involved in **producing** and **delivering** or otherwise influencing that resource.

Provenance provides a critical foundation for assessing **authenticity**, enabling **trust**, and allowing **reproducibility**. Provenance assertions are a form of contextual **metadata** and can themselves become important records with their own provenance.

W3C Provenance Incubator Group (2010)
Metadata

**Metadata**

*Definition of METADATA*

: data that provides information about other data

*First Known Use of METADATA*

1983
Definition: Metadata

Metadata is structured data that is used to describe the properties of a resource.
Example

Christie, Agatha

- 159 S. (The crime club)

HM 2153
C555
B668

Missing
All things described by RDF are called *resources*, and are instances of the class rdfs:Resource. This is the class of everything. All other classes are subclasses of this class.

Information about resources is expressed in *statements* about the resource.
RDF Resources

RDF URI Reference

URIs are globally unique and every URI identifies one and only one resource.

Literal

Identify values such as numbers and dates
Typed or plain

(Blank node

A resource that exists, but is not identified by a URI)
A statement…

… is a triple of subject, predicate, and object,

… generally describes one property of one identifiable resource by assigning a value.

The subject is always a resource (a blank node or identified by a URI).

The object can be another resource or a literal.
@prefix dcterms: <http://purl.org/dc/terms/>  
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>  
@prefix swb: <http://swb.bsz-bw.de/DB=2.1/PRS=rdf/PPNSET?PPN=>

swb:078273714  
a rdf:resource ;  
dcterms:title “The body in the library”;  
dcterms:creator http://d-nb.info/gnd/118520628;  
dcterms:issued “1976”;  
dcterms:publisher “Collins”;  
dcterms:format “print”.

Example
RDF visualized as graph
Linked Data

Linked Data Principles:

1) Use URIs as names for things.
2) Use HTTP URIs so that people can look up those names.
3) When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL).
4) Include links to other URIs, so that they can discover more things.

http://www.w3.org/DesignIssues/LinkedData.html
Linked Data

Information resources

Resources that are delivered via the Web:
Web pages, images, PDF files, ...

Non-information resources

Resources that are not on the Web:
Books, concepts, persons, ...
Linked Data

Dereferencing a URI from RDF data

Non-information resources

Using http redirects (303 redirect)

Delivers information on the resource in RDF format

Information resource

Depending on content negotiation and using http redirects

Delivers the resource itself

or

Delivers information on the resource in RDF format
Metadata in a linked data environment

Now metadata on a given resource...

... can come from many sources,
... can contain redundant statements,
... can contain false or contradictory statements,
... can be created by many means and processes.

One would like to keep track of those statements

But provenance — as defined — only deals with resources.
Thus: We need a notion of metadata as a resource.
Example: Data enrichment

http://swb.bsz-bw.de/DB=2.1/PRS= rdf/PPNSET?PPN=078273714

http://d-nb.info/gnd/118520628

"Collins"
"156 pages"
"paper"
"print"
"1976"
"The body in the library"
Add different abstracts

"It’s seven in the morning. The Bantrys wake to find the body of a young woman in their library. She is wearing evening dress and heavy make-up, which is now smeared across her cheeks. But who is she? How did she get there? And what is the connection with another dead girl, whose charred remains are later discovered in an abandoned quarry? The respectable Bantrys invite Miss Marple to solve the mystery... before tongues start to wag."@en

http://swb.bsz-bw.de/DB=2.1/PRS=rdf/PPNSET?PPN=078273714

"The body of a dancing hostess from a seaside resort turns up in the library of a married colonel. Miss Marple is her customary uncanny self in aiding the local police find the killer."@en
Add subject information

http://swb-bsz-bw.de/DB=2.1/PRS=rdf/PPNSET?PPN=078273714

http://d-nb.info/gnd/4047289-9

http://d-nb.info/gnd/4165727-5

"Kriminalgeschichte"

skos:prefLabel

dcterm:subject

"Privatbibliothek"

skos:prefLabel
Metadata in a linked data environment

One would like to keep track of those statements

But provenance — as defined — only deals with resources

We need metadata provenance:

What dataset does a given statement belong to?
Who (or what) is responsible for it?
"It’s seven in the morning. The Bantrys wake to find the body of a young woman in their library. She is wearing evening dress and heavy make-up, which is now smeared across her cheeks. But who is she? How did she get there? And what is the connection with another dead girl, whose charred remains are later discovered in an abandoned quarry? The respectable Bantrys invite Miss Marple to solve the mystery... before tongues start to wag."

"The body of a dancing hostess from a seaside resort turns up in the library of a married colonel. Miss Marple is her customary uncanny self in aiding the local police find the killer."
Part 2: Identification of RDF Data
What’s in the standards? A brief review of Reification.
Linked Metadata (Use the LD Principles)
Named Graphs
RDF 1.1
Expressing provenance in RDF

RDF offers a way to describe statements: Reification

- New resource to represent a statement
- Subject, predicate and object as properties of this resource
- Additional information using additional properties
Example

```
exproducts:item10245  exterms:weight  "2.4"^^xsd:decimal .
```

```
exproducts:triple12345  rdf:subject     exproducts:item10245 .
exproducts:triple12345  rdf:object      "2.4"^^xsd:decimal .
```

Example

Limits

No link between statement and reification:
Only by matching subject, predicate, object.

No grouping possible:
Excessive numbers of statements, e.g. identical creator for 100 statements leads to 500 additional statements.

Reification can be used to talk about specific statements, but is not practicable to provide the provenance of a whole dataset.
Linked Metadata
Linked Data Principles

1) Use URIs as names for things.
2) Use HTTP URIs so that people can look up those names.
3) When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL).
4) Include links to other URIs, so that they can discover more things.

http://www.w3.org/DesignIssues/LinkedData.html
Linked Metadata

How do we get the metadata provenance?

Usual best practice: deliver it with the metadata.
Embedded Linked Metadata (Method 1)

Drawback:

What about the provenance of the provenance?
There is no URI for the metadata provenance.
Then we give the metadata provenance a URI!

Problem: How to tell that we want the provenance.

Content negotiation is not working any more, as both contents are RDF.

Missing: A request header that asks for provenance.
The Link Header (Method 2)

Response header sent by `ex:eiffeltower-meta`:

```
Link: <http://example.org/eiffeltower-metameta>; rel=meta
```

Drawback: Additional (head) request needed.
Additional Statements (Method 3)

Provide a **reference** to the provenance data:

For example:

```
ex:eiffeltower-meta rdfs:seeAlso ex:eiffeltower-metameta.
```

**Drawback:** `rdfs:seeAlso` very general. There is no commonly accepted property for provenance.
Linked Metadata Summary

+ Based on Linked Data Principles.
+ Current “best practice.”
- Not suitable for provenance on statement level.
- Requires full control over web server.
- No URI for provenance information, or
- no accepted provenance retrieval mechanism.

Despite the drawbacks: a good starting point, as every provenance mechanism has to fit with the linked data principles.
Named Graphs
Named Graphs

A Named Graph is an RDF graph with an assigned URI as name.

Serialization in TriG:

```trig
ex:eiffeltower-meta {
  ex:eiffeltower rdf:type ex:building.
  ex:eiffeltower ex:height-in-meters "324".
  ex:eiffeltower dcterms:date "1889".
  ...
}
```

Named Graphs will be part of the RDF 1.1 standard, and are supported in SPARQL.
Named Graphs in RDF Stores

RDF-Stores today are usually quad-stores. (not triple-stores, even if we call them that way)

Each triple is assigned to a graph via the fourth quad element.

If the fourth element contains a URI, the URI is interpreted as the name of the graph that contains all triples with the same graph URI.
SPARQL supports Named Graphs:

```sparql
SELECT ?origin ?p ?o WHERE {

  GRAPH ?origin {
  }
}
```

This retrieves all statements about graph URIs containing a certain statement (e.g., provenance).
Named Graphs and Linked Data

A client that fetches linked data via a URI *usually* stores this URI as graph URI in a quad store.

This is **great**, because this way we can talk about the fetched RDF data and store provenance in our RDF store.

This is **only half way there**, because we can not reexpose the provenance information easily.

Because it is not part of RDF.
RDF 1.1
RDF WG

Mission:

*Update the 2004 RDF Recommendations, extending RDF to include features desirable and important for interoperability, but without a negative effect on deployment.*

Required Feature (Charter) among others:

- Support for **Multiple Graphs** and Graph Stores.
- Standardize the Turtle RDF Syntax. Either that syntax or a related syntax should also support **multiple graphs**.

http://www.w3.org/2011/01/rdf-wg-charter
Named Graphs in RDF 1.1 (Work in Progress!)

From RDF 1.1 Concepts and Abstract Syntax (Last Call Working Draft, July 23, 2013):

An RDF Dataset is a collection of RDF graphs and comprises [...] zero or more named graphs.

Each named graph is a pair consisting of an IRI (the **graph name**), and an **RDF graph**.

Note:

The graph name does **not** formally denote the graph.

RDF does **not** place any formal **restrictions** on what resource the graph name may denote, nor on the relationship between that resource and the graph.
RDF Graphs

What is an RDF Graph?

An RDF graph is a set of RDF triples.

That means that a (named) RDF Graph does not contain other (named) graphs.

Consequences:

You can reexpose graphs with names (e.g., with TriG), but: no directions how to interpret the graph URI, and: when the TriG file is fetched, no possibility to store the graphs inside another graph with the URI of the TriG file.
Summary

Half way there, but still enough room for own decisions and developments.

Positive thinking ;-)
Part 3: Metamodels in Practice
Agenda

OAI-ORE
The Europeana Data Model and DM2E
OAI-ORE “vs.” Named Graphs
Linked Data Publishing with VoID
OAI-ORE
Open Archives Initiative - Object Reuse and Exchange

Originally addresses another problem that lacks a solution in RDF:

- How to make a statement about a resource that is only valid in a special context?

  Example: The ordering of resources in an aggregation, like the ordering of articles in a bibliography.

Adaption for provenance:

- All statements are provided within such a context, the context can be identified and further described by provenance statements.
OAI-ORE Graph

ore:ResourceMap ex:rem1

ore:Proxy ex:proxy1

ore:Aggregation ex:agg1

ore:Proxy ex:proxy2

rdfs:Resource, ore:aggregatedResource ex:res1


ore:describes

ore:proxyIn

ore:aggregates

ore:proxyIn

ore:aggregates

ore:proxyFor

ore:proxyFor

"Real world" resources

Proxies

Web resources

Aggregations
OAI-ORE and Linked Data

The Resource Map is just a web resource with an own URI.

The Resource Map is connected to the Aggregation via ore:describes.

The Aggregation and the Proxies provide the scaffolding for the statements that are made in the context of the Aggregation.

Drawback: An application has to be “ORE-aware” to make sense of all this, as the concept of a proxy resource is not known in RDF.
Europeana Data Model
and
DM2E Model
Europeana provides data about cultural heritage objects (CHO) from CH institutions all over Europe.

Provenance requirement: Distinguish metadata from different institutions talking about the same (owl:sameAs) resource.
DM2E Mission Statement

DM2E develops infrastructure to:

1) ingest metadata from data providers,
2) transform metadata to RDF,
3) provide RDF data for consumption by researchers from digital humanities,
4) deliver RDF data to Europeana.
Provenance realized by means of OAI-ORE.

Problems?

Users have to understand **Proxies**.

Users have to understand **Aggregations**.

Wouldn’t named graphs be nicer?
How are proxies and aggregations used?

What is an aggregation?

“Aggregations are used in Europeana to represent the complex constructs that are provided by contributors. An aggregation is associated to the object that it is about, by the property edm:aggregatedCHO.”

Level of aggregation:

1 aggregation per providedCHO.

EuropeanaAggregation aggregates other aggregations (from data providers).
Removing the proxies

Proxies are (proxy-) resources for the actual resources. Every data provider has an “own” resource to describe, as a placeholder.

But: Data providers use different URIs for their resources anyway.

Linking creates owl:sameAs statements and conflates resources.

How can we then reliably maintain different descriptions?

We simply use named graphs to distinguish descriptions from different providers.
A Named Graph per Resource

Corresponds to the EDM aggregations. Finegrained... feasible?
Named Graphs as first class members in the model.

Statements about the aggregation that are only valid for one resource!

If we allow this, the named graph must never get lost!
Nested Graph Problem

Named Graphs are connected to Linked Data Principles.

One Named Graph per document fetched from a URI.

If we provide a dump of the full dataset from one provider, we have several named graphs within one resource that form another named graph...

=> Nested Graph Problem.

RDF does not provide a solution, it is not clear how to deal with such data.
Linked Data Provenance Mantra:

All published statements belong to only one named graph!

*Because everything you publish, IS a named graph.*
Solution: A Named Graph per Provider

This information must not get lost, too. But: It is not only valid for one resource. We are now more flexible regarding the level of aggregation.
One Named Graph per Provided Dataset

Naturally fits to provenance requirements: All statements stem from some dataset.

Positive aspect: Dataproviders do not have to care any more!
Provenance and Versioning on Dataset Level
Overlapping Resource Descriptions
Linked Data Publishing using VoID
What’s inside our store?

RDF Datasets, organized in named graphs.

NG URI scheme:

http://data.dm2e.eu/data/dataset/[provider]/[datasetId]/[version]

VoID (http://www.w3.org/TR/void/):

Each named graph is a void:Dataset.

Additional provenance statements for each dataset.
Make it available

Web documents (with URI) deliver RDF, provenance is included as statements about the URI.

Each Web document is a foaf:Document.

Each Web document contains a statement that links to the void:Dataset:

ex:doc1 void:inDataset ex:dataset1 .
RESTful API (Web documents)

http://data.dm2e.eu/data/…

… `dataset/[provider]/[datasetID]/[version]`
  => (Provenance) information about the dataset

… `resource/[provider]/[identifier]`
  => 303 to latest version

… `dataset/[provider]/[datasetID]/[version]/[identifier]`
  => data about a single resource

… `linkset/[DM2E]/[linksetId]/[version]`
  => generated links

… `linkset/[DM2E]/[linksetId]/[version]/[provider]/[identifier]`
  => links for a specific resource

Hint: Documents contain a `[version]`. 
Provenance in Documents

Generated from provenance information about datasets:

- dc:creator => Data provider
- dc:date => Timestamp
- dm2e:version => version number
- dm2e:nextVersion => link to next version of the document
- dm2e:previousVersion => link to previous version
- dm2e:links => link to a linkset

Optional: PROV statements for full provenance chain.

Maintained by the DM2E infrastructure.

Version means always the version of the underlying dataset.
Consuming the data

Linksets and data enrichments are managed as separate datasets.

Applications have to combine the data as needed. => Preservation of provenance is left to consumer.

How should the data be organized on client side?

- A named graph per retrieved URL, as usual?
- Or a named graph per dataset, which would replicate the organization on the server?

Both is possible, but depending on the application one or the other way might be preferred.
Summary

Many different approaches:

1) Reification
2) “Simple” application of Linked Data principles.
3) Named Graphs
4) OAI-ORE
5) Own models and extensions

In practice, we have to combine them to create flexible solutions.

Unfortunately, the full understandability of linked data provenance is not (yet) guaranteed.
Part 4:
Linked Data Provenance
(Requirements and Limits)
Agenda

State-ful or State-less Data
Versioning
Identity and Provenance Context
State-ful data

Content on web pages can change, they are usually state-less.

Example for a state-less URL:
http://example.org/weather/lisbon

By commitment, the content of a URL can be kept stable, the URL represents a specific state, it is state-ful.

Example for a state-ful URL:
http://example.org/weather/lisbon/2013-09-02

Provenance

From Wikipedia, the free encyclopedia

For other uses, see Provenance (disambiguation).

Provenance, from the French provenir, "to come from", refers to the chronology of the ownership, custody or location of a historical object.[1] The term was originally mostly used in relation to works of art, but is now used in similar senses in a wide range of fields, including archaeology, paleontology, archives, manuscripts, printed books, and science and computing. The primary purpose of tracing the provenance of an object or entity is normally to provide contextual and circumstantial evidence for its original production or discovery, by establishing, as far as practicable, its later history, especially the sequences of its formal ownership, custody, and places of storage. The practice has a particular value in helping authenticate objects. Comparative techniques, expert opinions, and the results of scientific tests may also be used to these ends, but establishing provenance is essentially a matter of documentation.

In archaeology (particularly North American archaeology and anthropological archaeology throughout the world), the term provenience is used in related but a subtly different sense to provenance. Archaeological researchers use provenience to refer to the three-dimensional location or find spot of an artifact or feature within an archaeological site,[2] whereas provenance covers an object's complete documented history. Ideally, in modern excavations, the provenience or find spot is recorded (even when not with great precision), but in older cases only
Wikipedia provides a stable URL for every version of an article, the content of these URLs won't change, the URLs are state-ful.
State-ful RDF and provenance

State-ful URLs make provenance-life easier.

The URL represents the data, so it can be used to identify the fetched data in local systems without problems.

State-less URLs are no show-stopper.

But the fact that the data might have changed in the source should be indicated:

1. Use a local state-ful URL for your data.
2. Link to the state-less URL as source, e.g., via `dct:source` or `prov:wasDerivedFrom`. 
Versioning

Data always changes. Most applications with state-ful URLs will therefore need versioning. The necessary links to other versions can be included with the data.
Versioning vocabulary

**previousVersion**: links to the previous version of this dataset.

**firstVersion**: links to the oldest available version of this dataset.

**version**: serial number of this version, starting with 1.

**versionName**: provides a human-readable name for this version.

**nextVersion**: links to the next version of this dataset.

**latestVersion**: links to the latest available version of this dataset.

**availableVersions**: number of available versions of this dataset.
Avoid changing properties in your data

`nextVersion`: links to the next version of this dataset. Replace with a link to a state-less generic URL:

```
ex:doc1/version1 ex:isVersionOf ex:doc1```

The following information is then linked to the `generic URL`:

`latestVersion`: links to the latest available version of this dataset.

`availableVersions`: number of available versions of this dataset.
The identity of a triple

The provenance of a triple (or any set of triples) defines its identity.

To keep track of metadata provenance, we have to identify triples and make sure we do not lose their identity.
DM2E: A typical Linked Data application
Nested Graph Problem Revisited

Model perspective $\iff$ Implementation perspective

**Modeling:** Named Graphs

**Implementation:** Web documents, void:Datasets, Named Graphs (Quads), ...
Provenance Context

The provenance context is the named graph that establishes the identity of the contained triples.

Other graphs can be used additionally, for publishing or organizing purposes.
Implicit Provenance Context

Proposal derived from practice:

1. By default, the URL of the RDF data is the provenance context.

2. If a void:Dataset is given for the data, then this dataset is the provenance context.

3. You can specify the provenance context explicitly (e.g., by using dm2e:inProvenanceContext)
Properties of the Provenance Context

1. There must always be **one and only one** Provenance Context per statement (as it defines its identity).

2. Every named graph (document, named graph, ...) **either is** a Provenance Context **or is contained completely** within one Provenance Context.

3. It follows then, that the Provenance Context **determines the maximum permissible set** of RDF statements published together.
Practical Implications

No publishing of merged statements from different sources.

Leave the merging to the consuming application.

Pedantic Web: Do not publish the provenance statements together with the data, if they do not share the same provenance.

Break these rules if you have to ;-)
Summary

The problem of metadata provenance is the **stable identification** of data.

The problem gets worse if the data starts to move around, i.e., **when it is consumed and republished**.

There are **limitations** for clean solutions **resulting from the web architecture**.

If you know these limitations, you can create applications that work **perfect** for you...

... and reasonable **well** for all others (i.e., they follow common practices).
Outlook:
Expressing Provenance in RDF
Several ontologies modelling provenance exist

W3C PROV
Open Provenance Model (OPM)
Provenance Vocabulary
Provenir

... 

And we have Dublin Core...
Some of the 55 terms contain only information about the resource itself, but not how or when it was produced

=> Descriptive Terms (What?)

Some terms also contain information on the creation or derivation of the resource

=> Provenance Terms (Who, When, How?)
Who?

Terms

- Contributor
- Creator
- Publisher
- RightsHolder

Range is dct:Agent

- a resource that acts or has the power to act
- Clearly influencing creation of a resource
- RightsHolder is ownership --> provenance in works of art
When?

Terms

Available
Created
Date
DateAccepted
DateCopyrighted
DateSubmitted
Issued
Modified
Valid
When?

Ranges

Date range
- Available, valid

Single date
- All others

Dates are basic provenance information
- Availability and validity often inherent to the resource
- But: provenance related, if active change
How?

Terms

IsVersionOf, hasVersion
IsFormatOf, hasFormat
References, isReferencedBy
Replaces, isReplacedBy
Source
HasPart, isPartOf
accrualMethod
How?

Information on Derivation and Replacement
Information on relations to other resources
Information on processes involved in creation
Definition

“statement of any changes in ownership and custody of the resource since its creation that are significant for its authenticity, integrity, and interpretation.”

--> “classic” provenance of works of art
Summary

More than half of the DC terms deal with provenance related information

Who?
When?
How?

Missing information
Where?
Why?

(only if replacement)
PROV Ontology
Basic contracts

Entities

Resources or “things” to describe

Activities

Creation or changes of entities

Agents

Those responsible for actions
Example

We have data on two books
- “The Glass Palace”, written by Amitav Ghosh
- “Le palais des miroirs”, the French translation, done by Christianne Besse, of the book of Amitav Ghosh
- we want to describe some very basic facts on the provenance of these

Source: Herman (2012)
Example

Source: Herman (2012)
Translation

Source: Herman (2012)
Summary

PROV is a complex abstract model

Initially harder to use than straightforward dcterms:
But is able to express complex relationships

PROV models provenance by describing actions

Activities that affected the state of a resource
Agents that are responsible for the action

PROV can easily model the whole lifecycle of a resource
Dublin Core vs. PROV

Dublin Core

- Very distinct roles, implicit and part of the semantics
- Limited number of terms

PROV

- Explicit modelling of roles
  - Subclassing
  - Qualified classes
Dublin Core is more specialized

<table>
<thead>
<tr>
<th>Property</th>
<th>rdfs:subPropertyOf</th>
<th>prov:wasAttributedTo</th>
</tr>
</thead>
<tbody>
<tr>
<td>dct:creator</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dct:rightsHolder</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dct:publisher</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## PROV can be specialized

<table>
<thead>
<tr>
<th>dcprov:CreationActivity</th>
<th>rdfs:subClassOf</th>
<th>prov:Activity, dcprov:ContributionActivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>dcprov:ContributionActivity</td>
<td>rdfs:subClassOf</td>
<td>prov:Activity</td>
</tr>
<tr>
<td>dcprov:CreatorRole</td>
<td>rdfs:subClassOf</td>
<td>prov:Role, dcprov:ContributorRole</td>
</tr>
<tr>
<td>dcprov:ContributorRole</td>
<td>subClassOf</td>
<td>prov:Role</td>
</tr>
</tbody>
</table>
Example
A mapping between DC and PROV is part of PROV:
http://www.w3.org/TR/prov-dc/
Conclusion

Distinct use cases

- Dublin Core as a simple provenance vocabulary with explicit roles
- PROV as an expressive abstract model
  - Modelling complex provenance chains and relations

Mapping possible

- Much provenance information is within Dublin Core metadata
- Semantic Web applications can make sense of both models, if they understand at least one.
END
Acknowledgements

The PROV example has been taken from:

Ivan Herman. (2012). The W3C PROV vocabulary. (Presentation slides), examples used with permission
http://de.slideshare.net/ivan_herman/the-w3c-prov-vocabulary

This tutorial is a revised version of the Metadata Provenance tutorial held at SWIB 2012, together with Magnus Pfeffer.