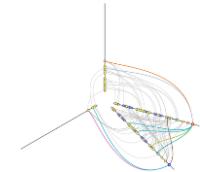


Provenance: An Introduction to PROV

Luc Moreau & Paul Groth & Trung
Dong Huynh



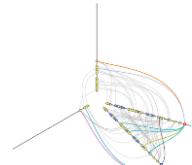


Acknowledgements



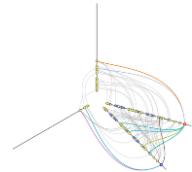
COMMIT /





Outline

- Notion of Provenance
- Examples of Provenance
- W3C Provenance Working Group
- PROV
- Provenance Recipes
- Tools / Management
- Summary
- Hands-on



References



PROV Model Primer
W3C Working Group Note 30 April 2013



PROV-O: The PROV Ontology



PROV-DM: The PROV Data Model
W3C Recommendation 30 April 2013

/

10430/

This version:

<http://www.w3.org/TR/2013/REC-prov-dm-20130430/>

Latest published version:

<http://www.w3.org/TR/prov-dm/>

Implementation report:

<http://www.w3.org/TR/2013/NOTE-prov-implementations-20130430/>

Previous version:

<http://www.w3.org/TR/2013/PR-prov-dm-20130312/> (color-coded, in progress)

Editors:

[Luc Moreau](#), University of Southampton
[Paolo Missier](#), Newcastle University

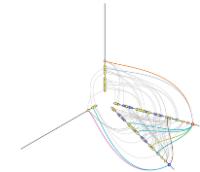
Contributors:

MORGAN & CLAYTON



SYNTHESIS LECTURES ON SEMANTIC WEB TECHNOLOGIES

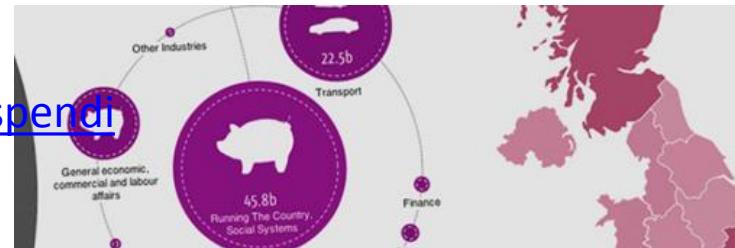
Luc Moreau
University of Southampton
Paul Groth
VU University Amsterdam

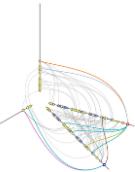


Open Data and Journalism

- Data journalism ethos: to expose the data and methods used to produce news items
- Data wrangling can introduce errors, data journalists should care about the validity of data; provenance of data should include its primary source, but also all the transformational steps performed by anyone.

http://datadrivenjournalism.net/featured_projects/how_spending_stories_spots_errors_in_public_spending

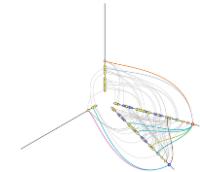




Tracing Information in the Social Web

- In the Social Web, users are given the ability to select contents from across the Web, integrate it together, edit it, rate it, publish it and share it with others.
- There is **a consume-select-curate-share** workflow similar to the data wrangling performed by data journalists, but typically require little technical expertise
- “Good curation demands good provenance. Provenance is no longer merely the nicety of artists, academics, and wine makers. It is an ethic we expect.”

<http://buzzmachine.com/2010/06/27/the-importance-of-provenance/>



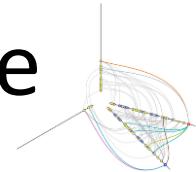
Reproducibility of Science

- Science is becoming computation and data intensive, but the fundamental tenet of the scientific method remains unchanged: experimental results need to be reproducible.
- Provenance is the equivalent of a logbook
 - capturing all the steps involved in the derivation of a result,
 - could be used to replay the execution that led to that result so as to validate it.



The
FOURTH
PARADIGM
DATA-INTENSIVE SCIENTIFIC DISCOVERY

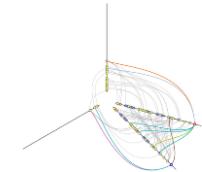
Accountability, Transparency, Compliance in Business Applications



- Steve New refers to the provenance of a company's products, and explains how businesses have changed their practice to make their **supply chain transparent**, because they worry about quality, safety, ethics, and environmental impact.
- Governments increasingly request transparency and provenance information in the area of anti-corruption compliance.
- Weitzner notes: provenance is a substrate that can be used to perform policy checks and to make systems accountable.

<http://hbr.org/2010/10/the-transparent-supply-chain/ar/1>

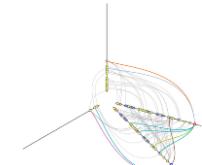




Provenance Definition

- Oxford English Dictionary:
 - the fact of coming from some particular **source** or quarter; **origin**, derivation
 - the **history** or pedigree of a work of art, manuscript, rare book, etc.;
 - concretely, **a record of the passage** of an item through its various owners.

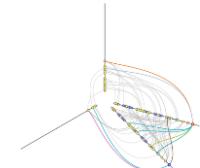




Provenance Definition (2)

- Provenance is a record that describes the people, institutions, entities, and activities, involved in producing, influencing, or delivering a piece of data or a thing in the world
- Provenance is crucial in deciding:
 - whether information is to be trusted,
 - how it should be integrated with other sources, and
 - how to give credit to its originators when reusing it.
- Provenance can help users to make trust judgments.



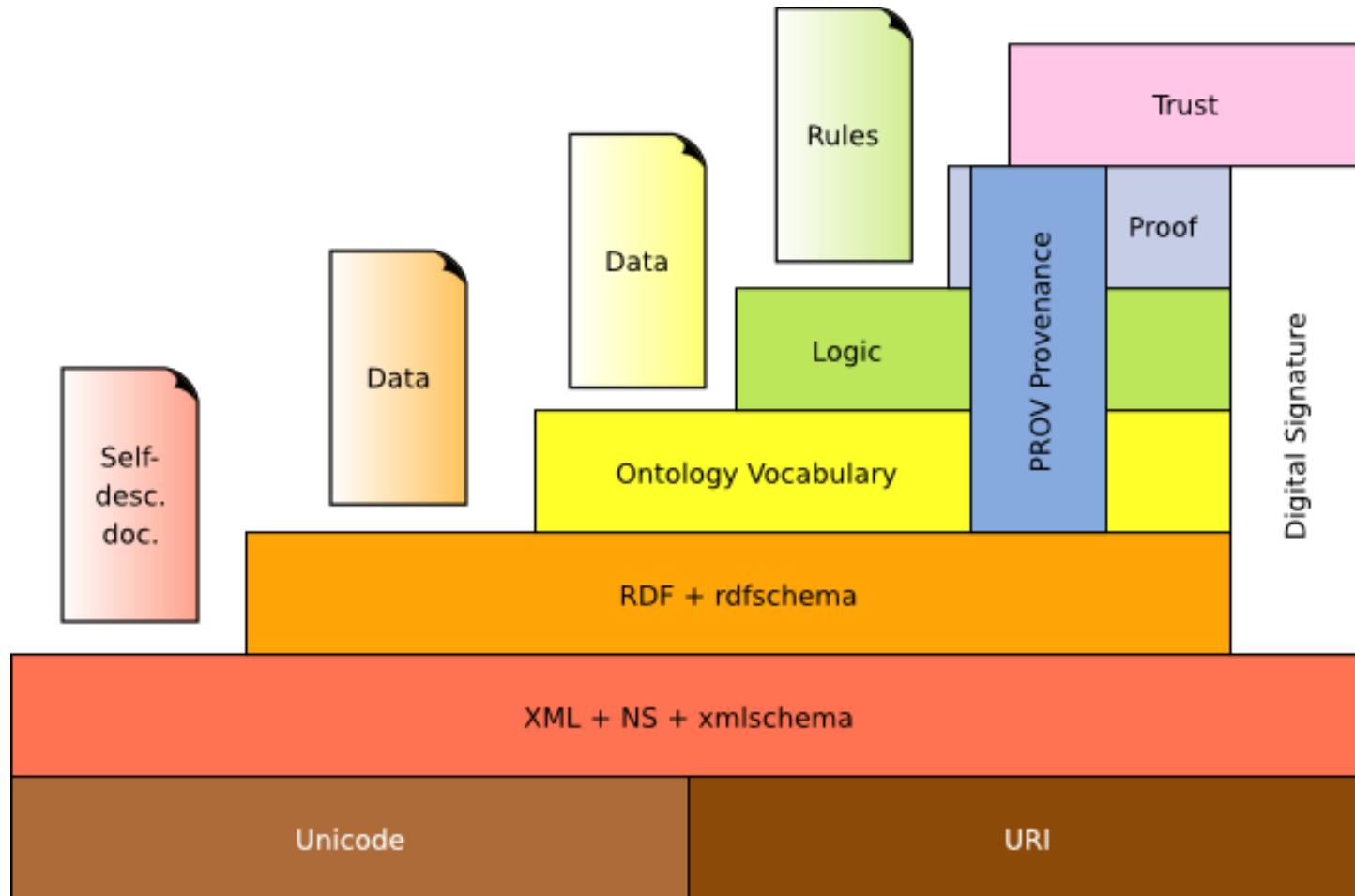


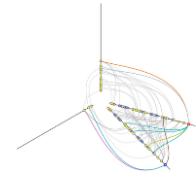
Provenance on the Web

Tim Berners-Lee's "Oh Yeah" button:

- A browser button by which the user can express their uncertainty about a document being displayed "so how do I know I can trust this information?".
- Upon activation of the button, the software then retrieves metadata about the document, listing assumptions on which trust can be based.

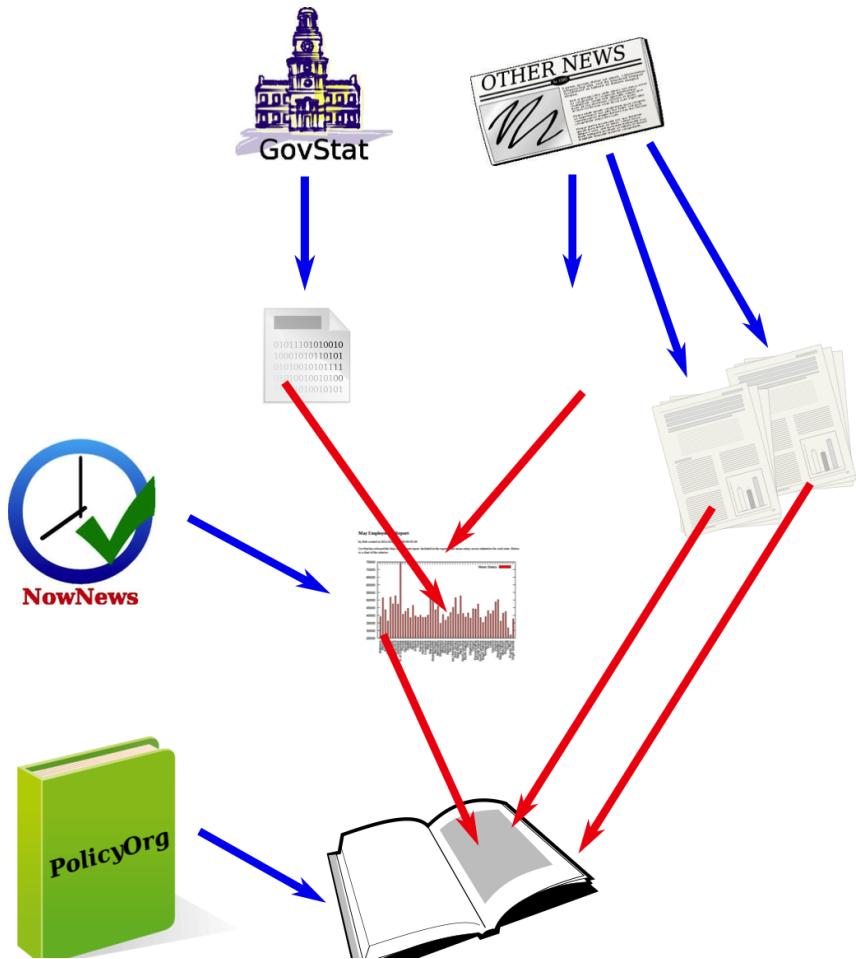
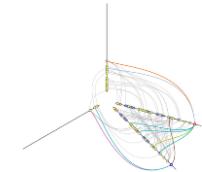
Provenance in the Semantic Web Stack



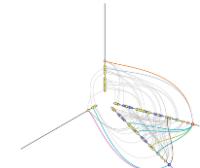


EXAMPLE: DATA JOURNALISM

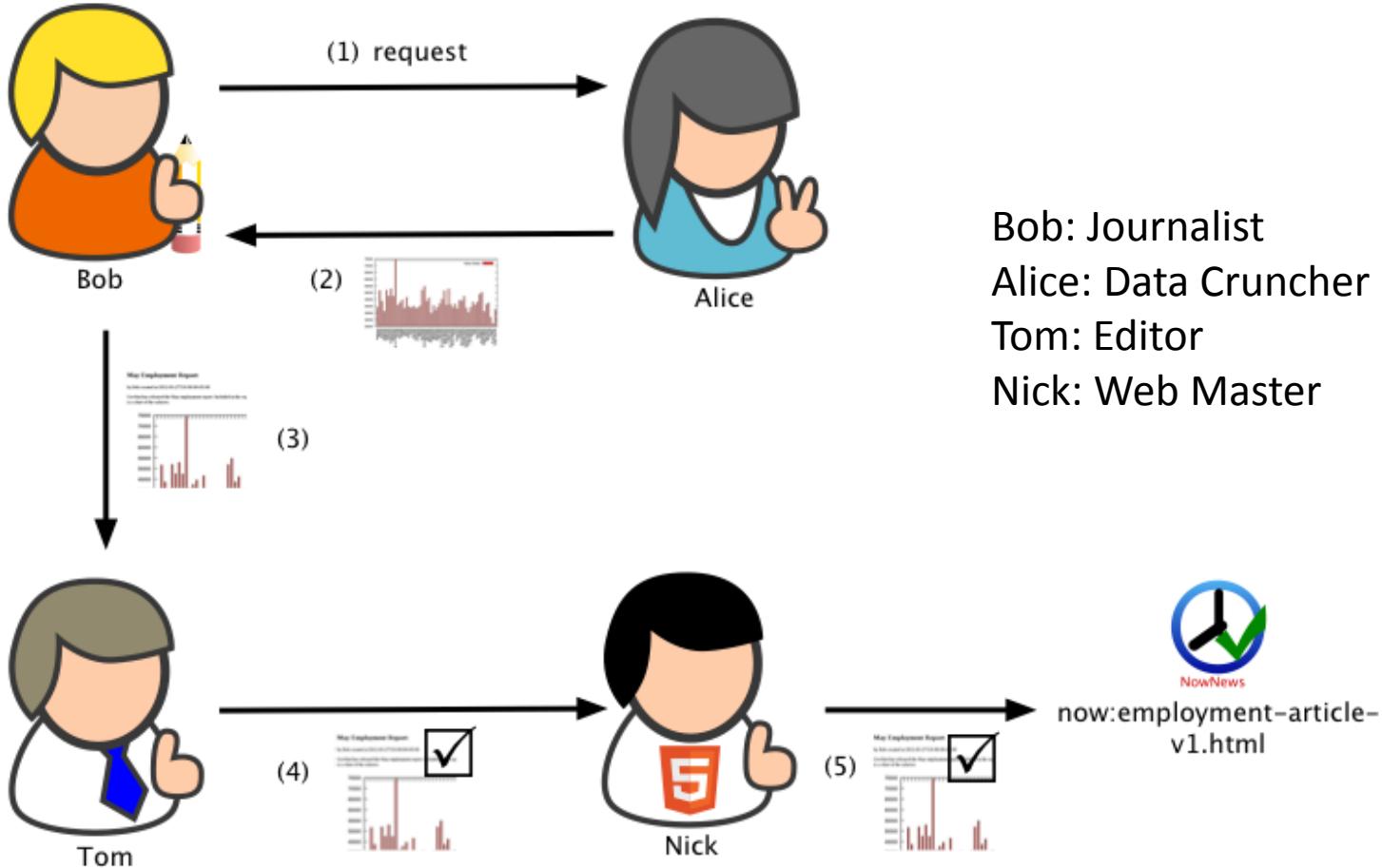
NowNews Publishing

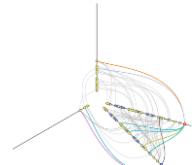


- NowNews publishes an article based on the latest employment data published by GovStat
- PolicyOrg compiles a report including NowNews article



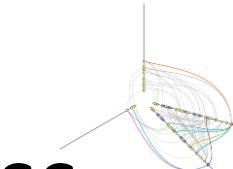
Within NowNews





Provenance Use Cases

- Quality Assessment
 - The latest data – timeliness
 - Finding trusted articles
 - Finding flawed figures
- Compliance
 - Following policy
 - Licensing
- Cataloging
 - Building an Index
 - Acknowledgements
- Replay
 - Reproducibility
 - Publication Embargo

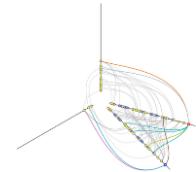


Use Case: The latest data - timeliness

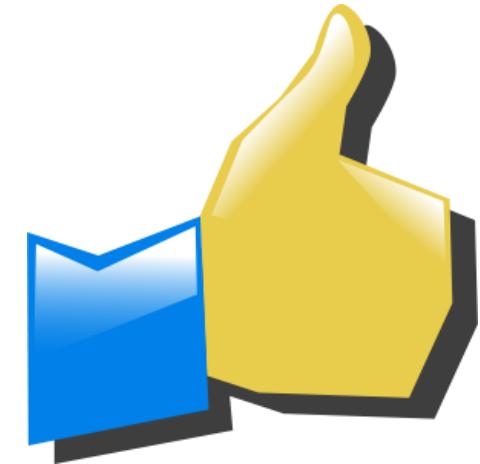
- PolicyOrg is about to issue their report publicly, before releasing the report they want to confirm that the report is based on the most up-to-date data.
- One of the figures that they have reused in the report stems from Bob's article on employment that appeared in NowNews.
- PolicyOrg needs to run a check that ascertains which data that figure was based upon.

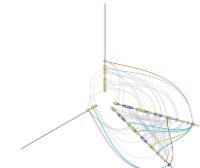


Use Case: Finding trusted articles

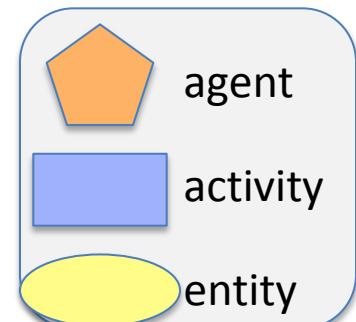
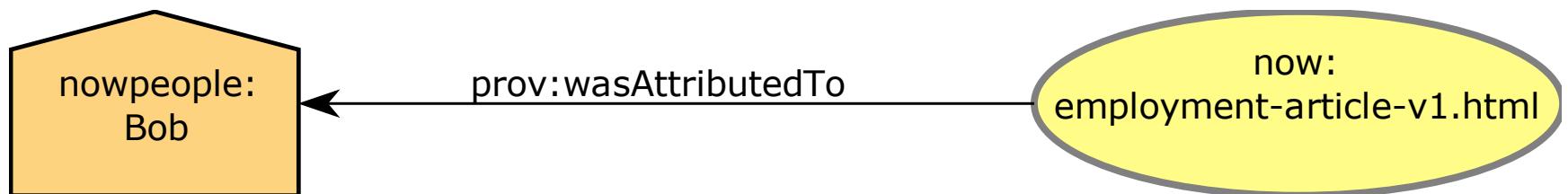


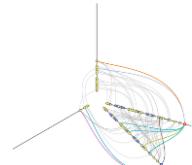
- When putting together a story or a report, content creators want to find information that is based on trusted sources.
- PolicyOrg may want to search for articles based on trusted sources information.
- PolicyOrg views data supplied by the government as reliable.
- However, when searching for content it is not always clear whether a content source is derived from data coming from such a reliable source.



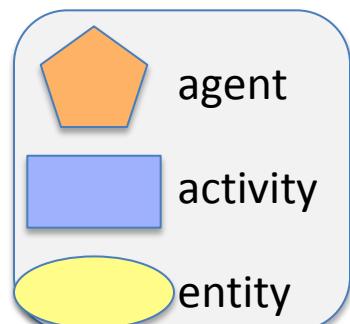
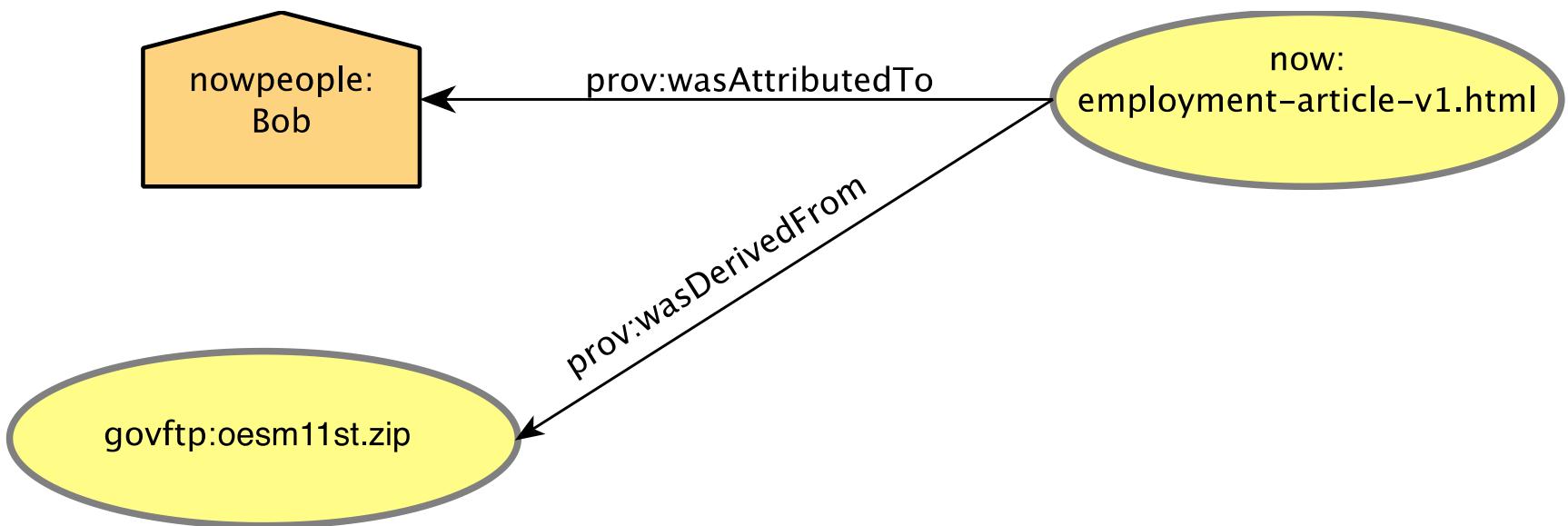


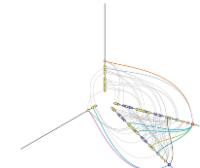
Attribution



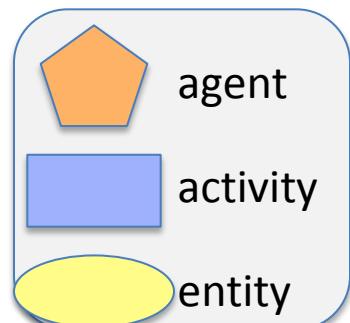
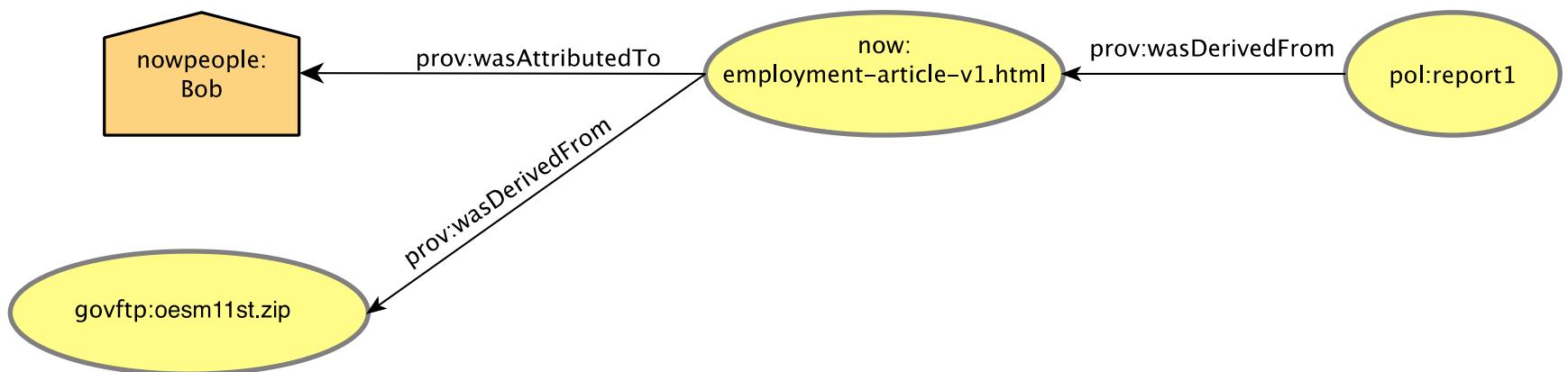


Derivation

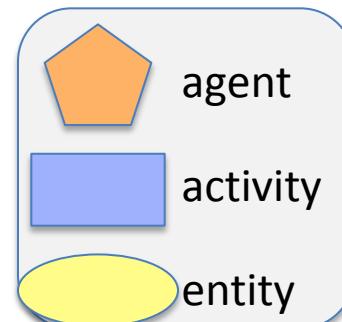
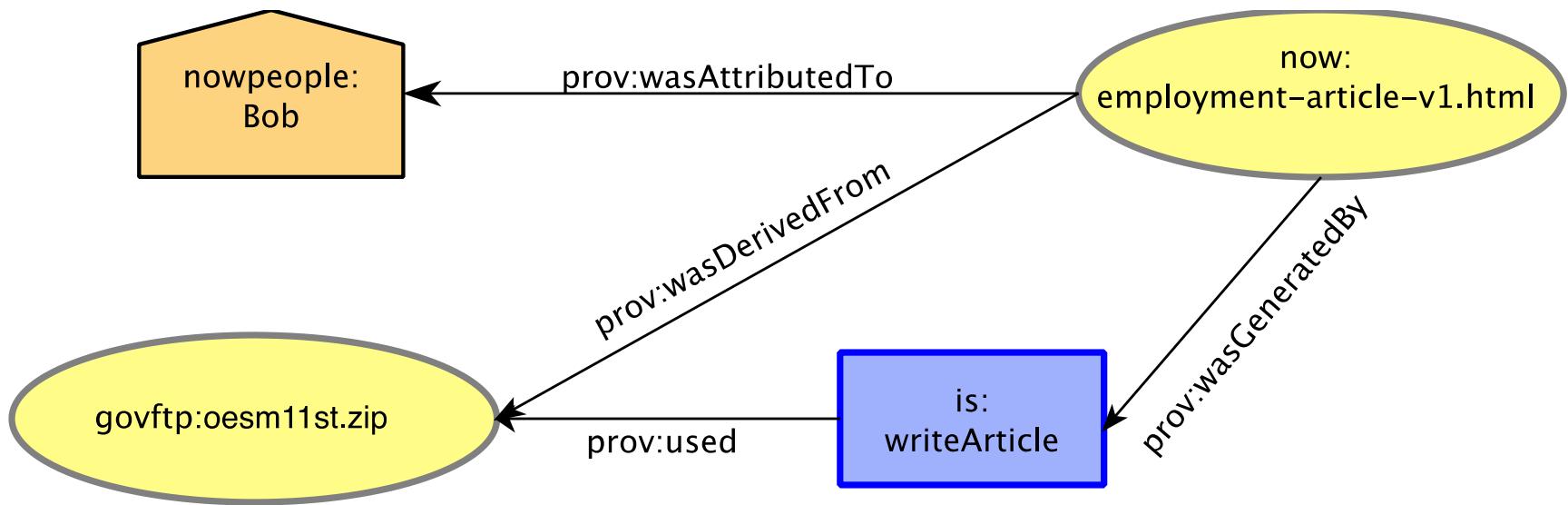




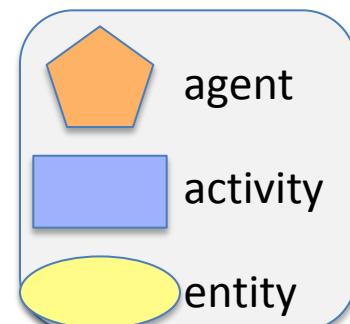
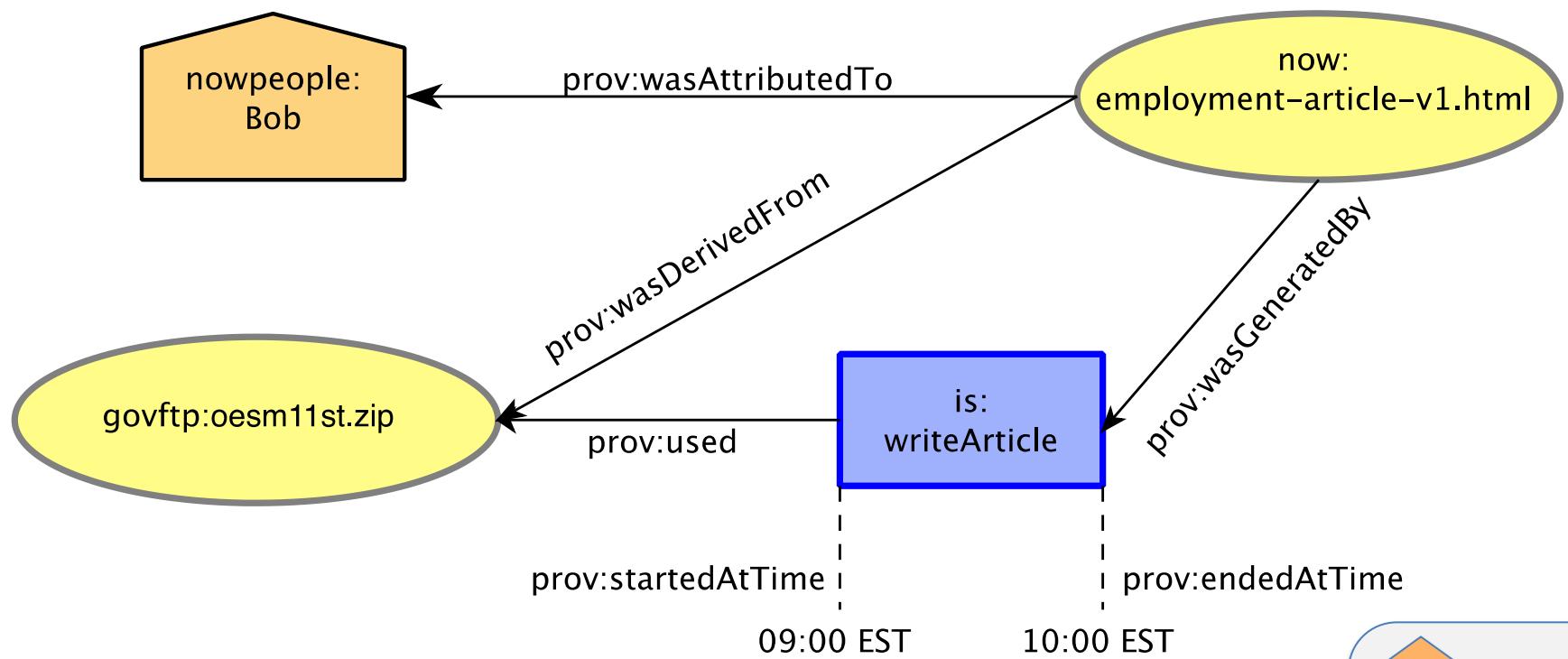
Derivation Chain

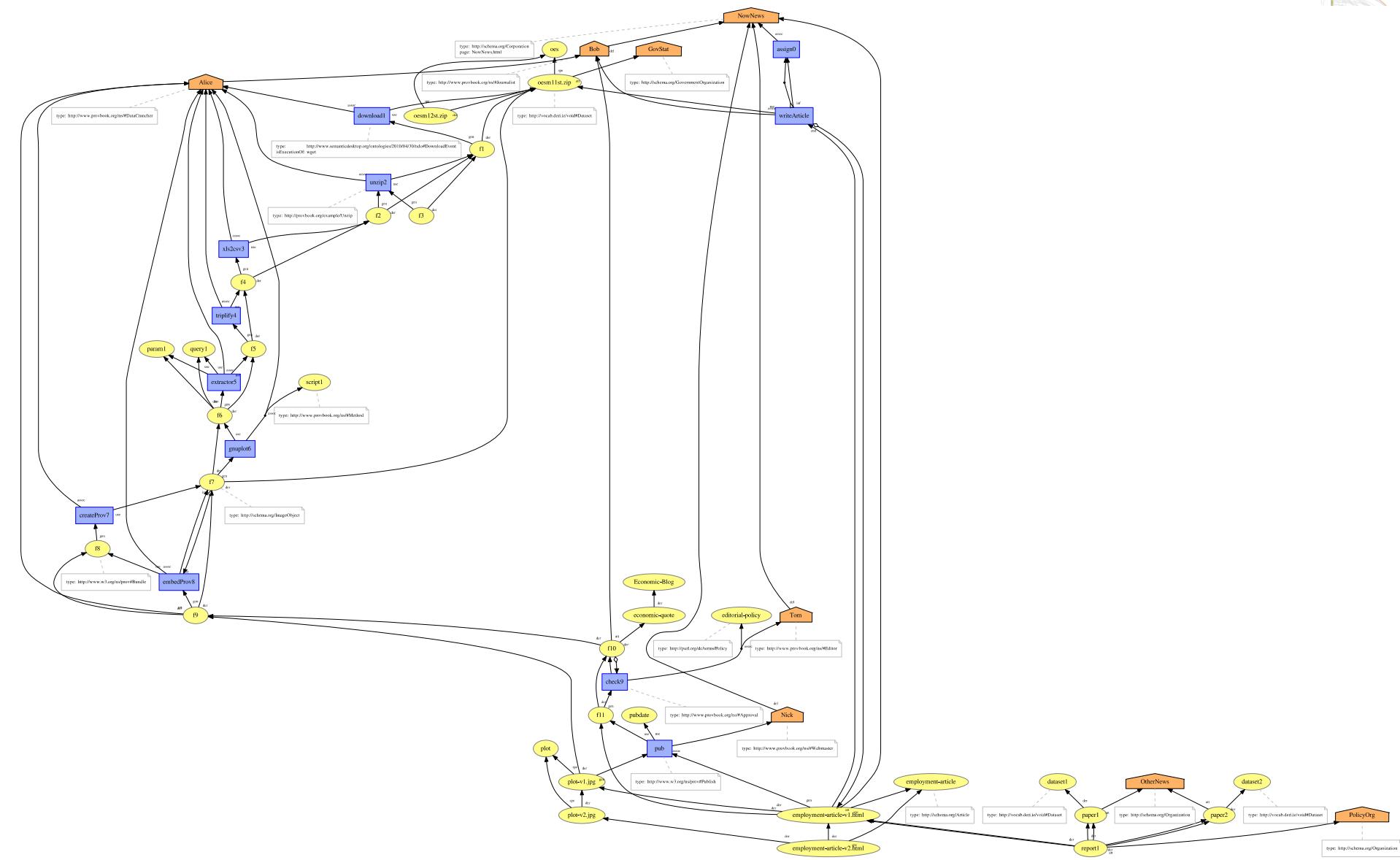


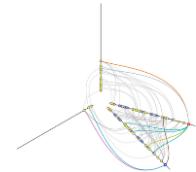
Activity: Writing Article



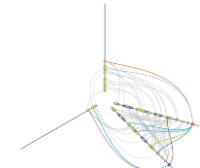
Activity Start and End Times







W3C PROVENANCE WORKING GROUP

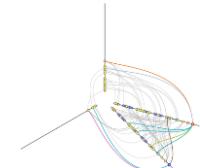


Provenance Interchange Working Group Charter

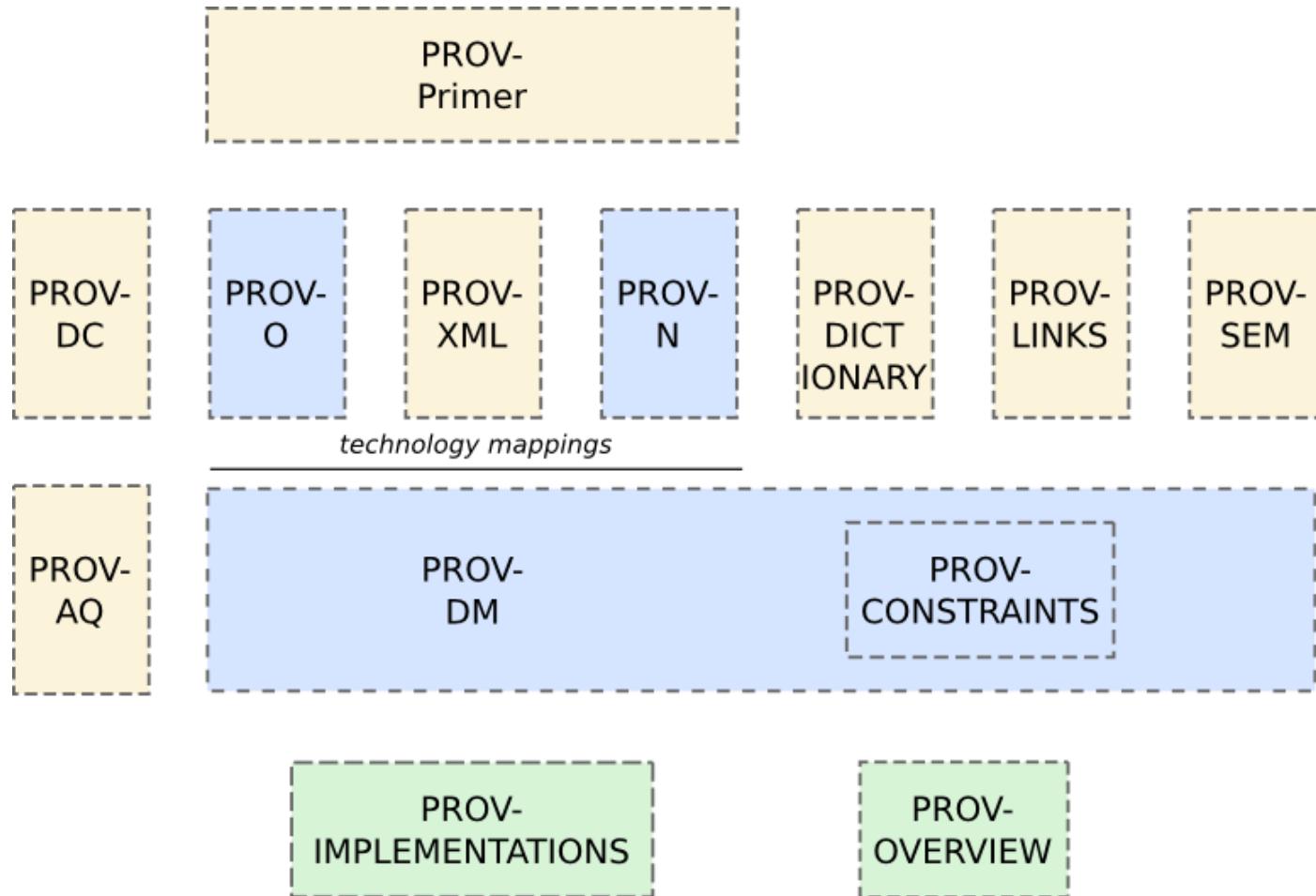
The **mission** of the [Provenance Working Group](#), part of the [Semantic Web Activity](#), is to support the widespread publication and use of provenance information of Web documents, data, and resources. The Working Group will publish W3C Recommendations that define a language for *exchanging* provenance information among applications.

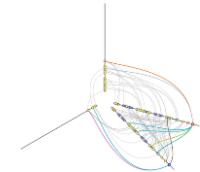
[Join the Provenance Working Group.](#)

End date	1 October 2012
Confidentiality	Proceedings are public
Initial Chairs	Luc Moreau , University of Southampton Paul Groth , VU University Amsterdam
Initial Team Contacts (FTE %: 20)	Sandro Hawke
Usual Meeting Schedule	Teleconferences: Weekly Face-to-face: Once Annually



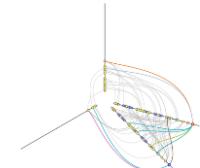
PROV Family of Specifications



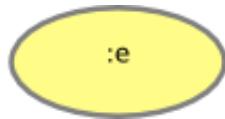


PROV





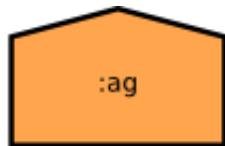
Three Core Classes



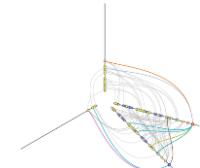
An entity is a physical, digital, conceptual, or other kind of thing with some fixed aspects; entities may be real or imaginary.



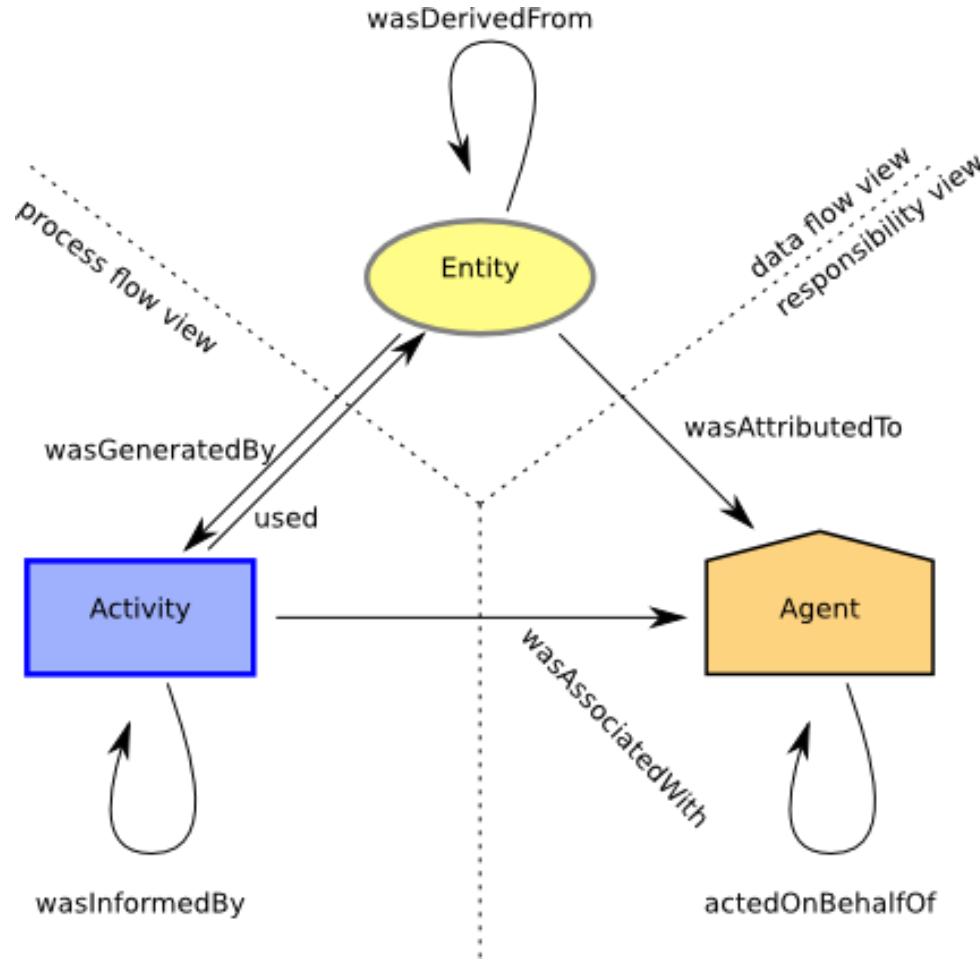
An activity is something that occurs over a period of time and acts upon or with entities; it may include consuming, processing, transforming, modifying, relocating, using, or generating entities.

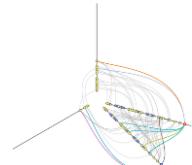


An agent is something that bears some form of responsibility for an activity taking place, for the existence of an entity, or for another agent's activity.

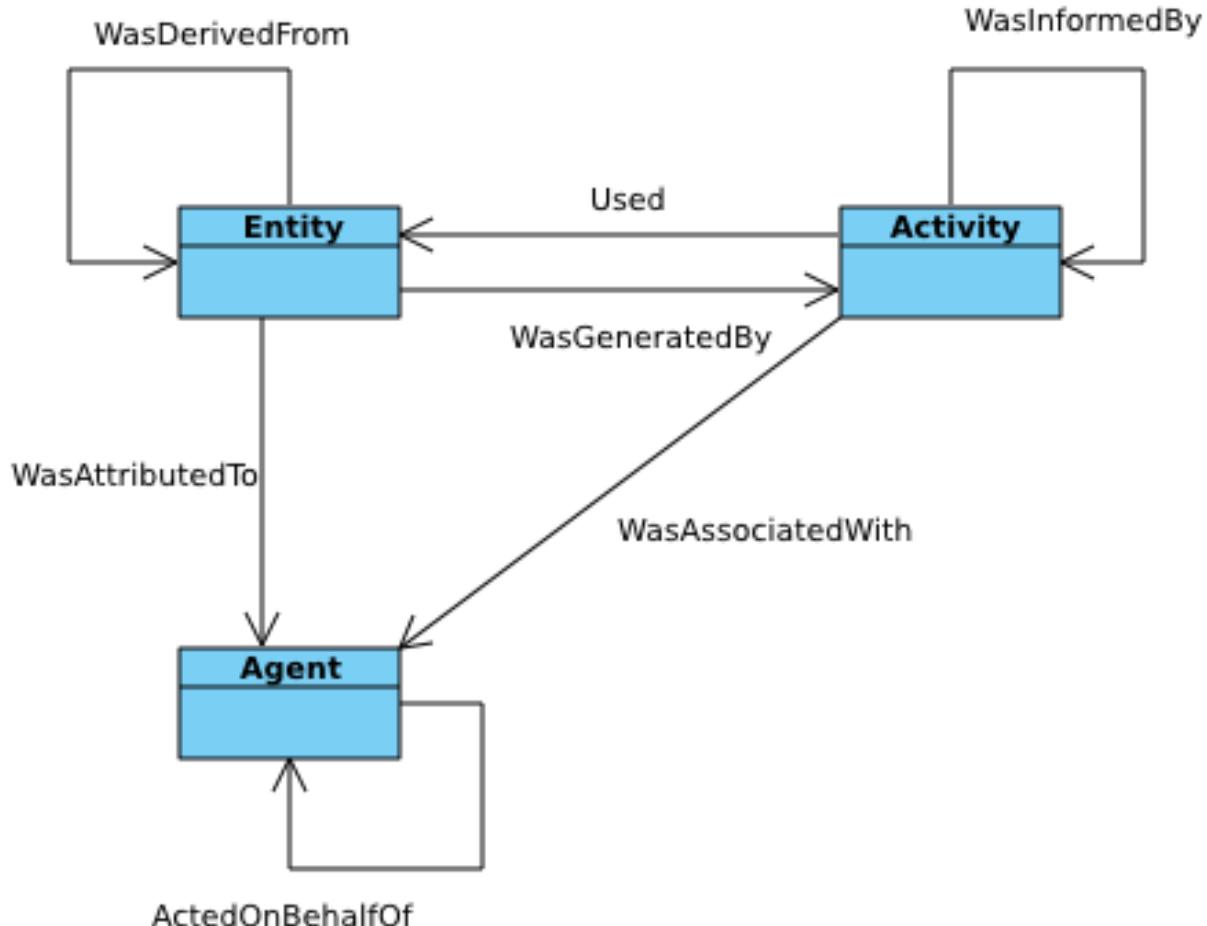


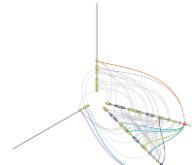
Three Views of Provenance



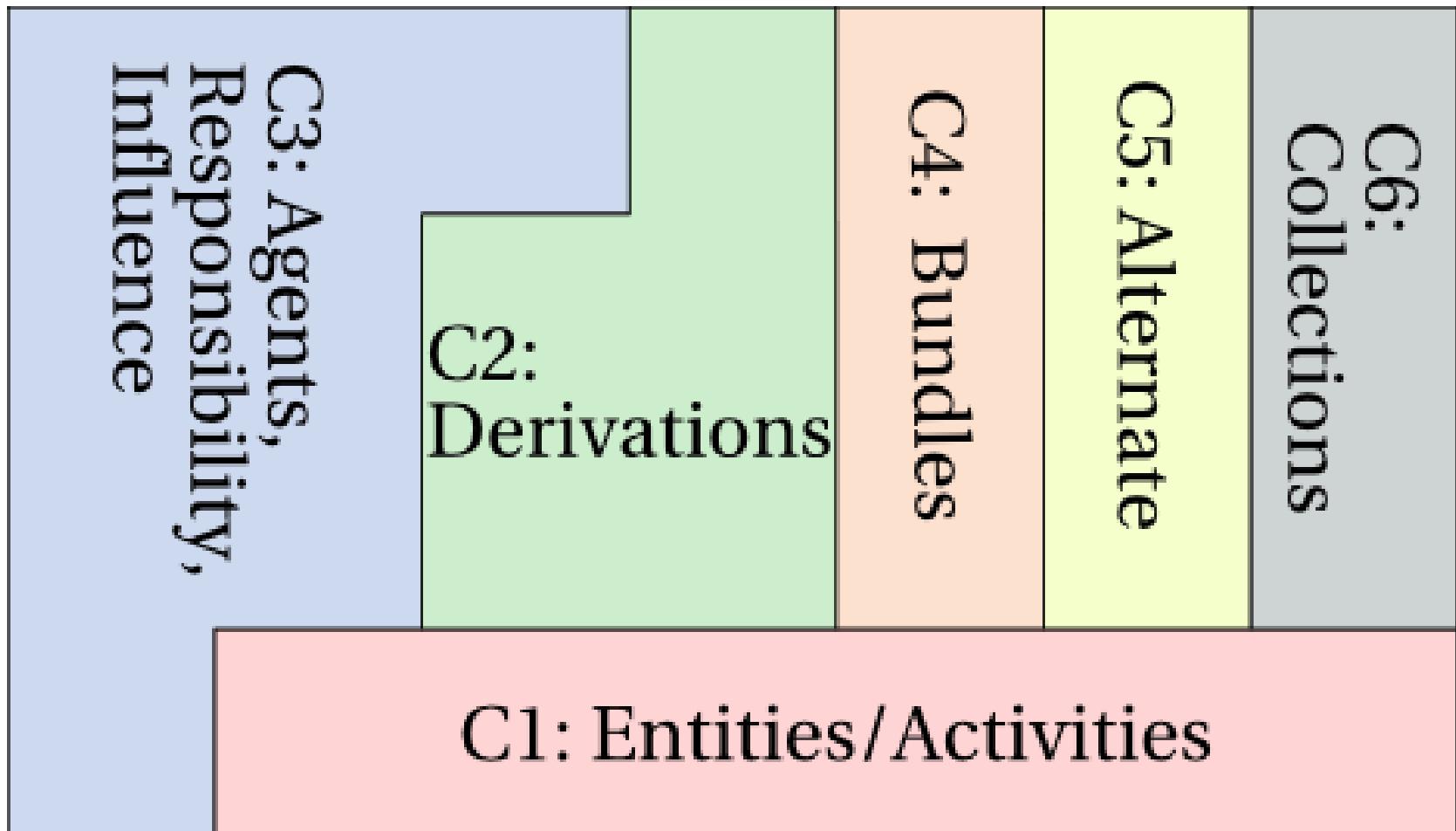


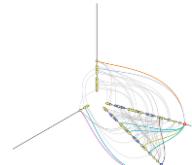
UML View of PROV Core



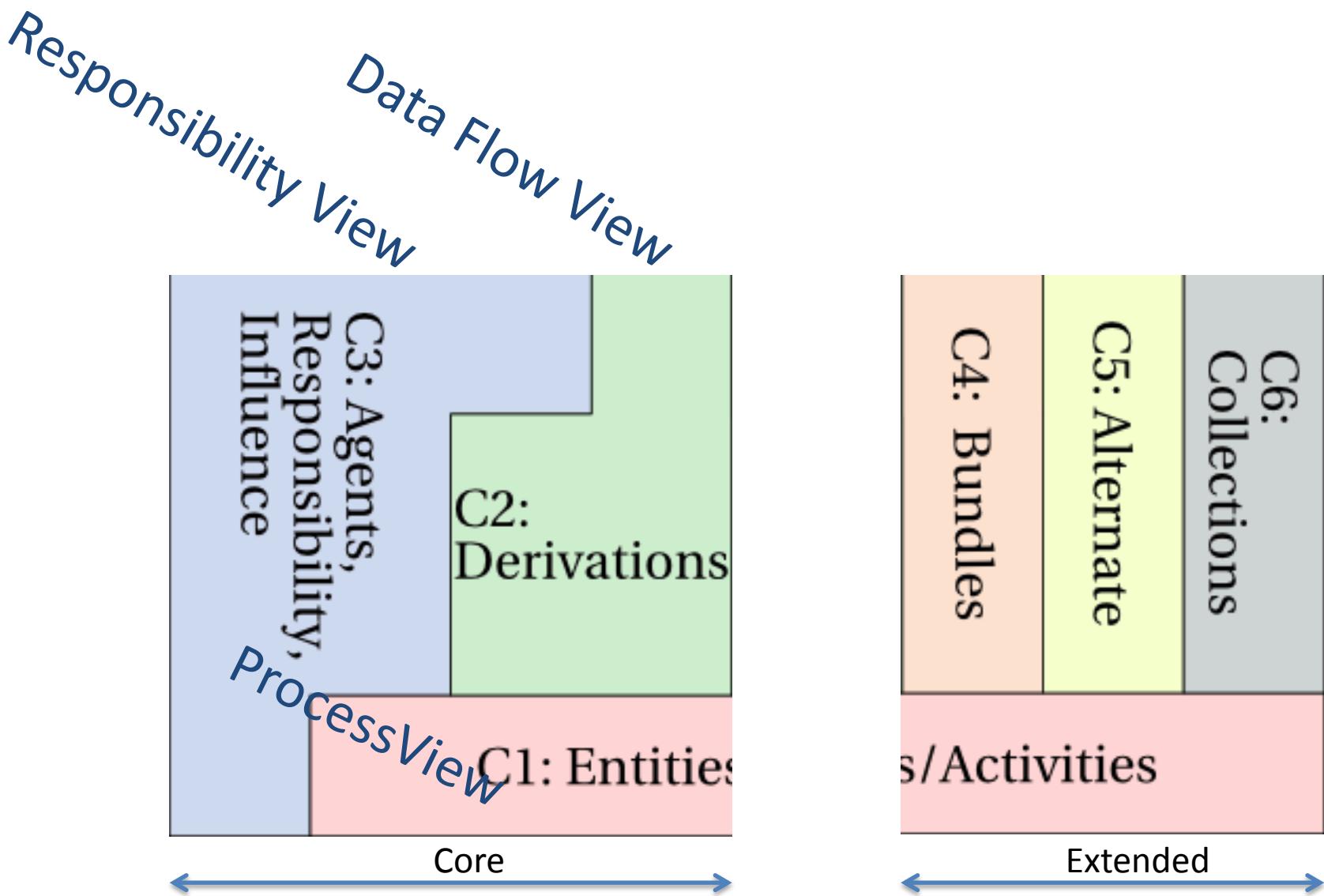


Component Structure for PROV

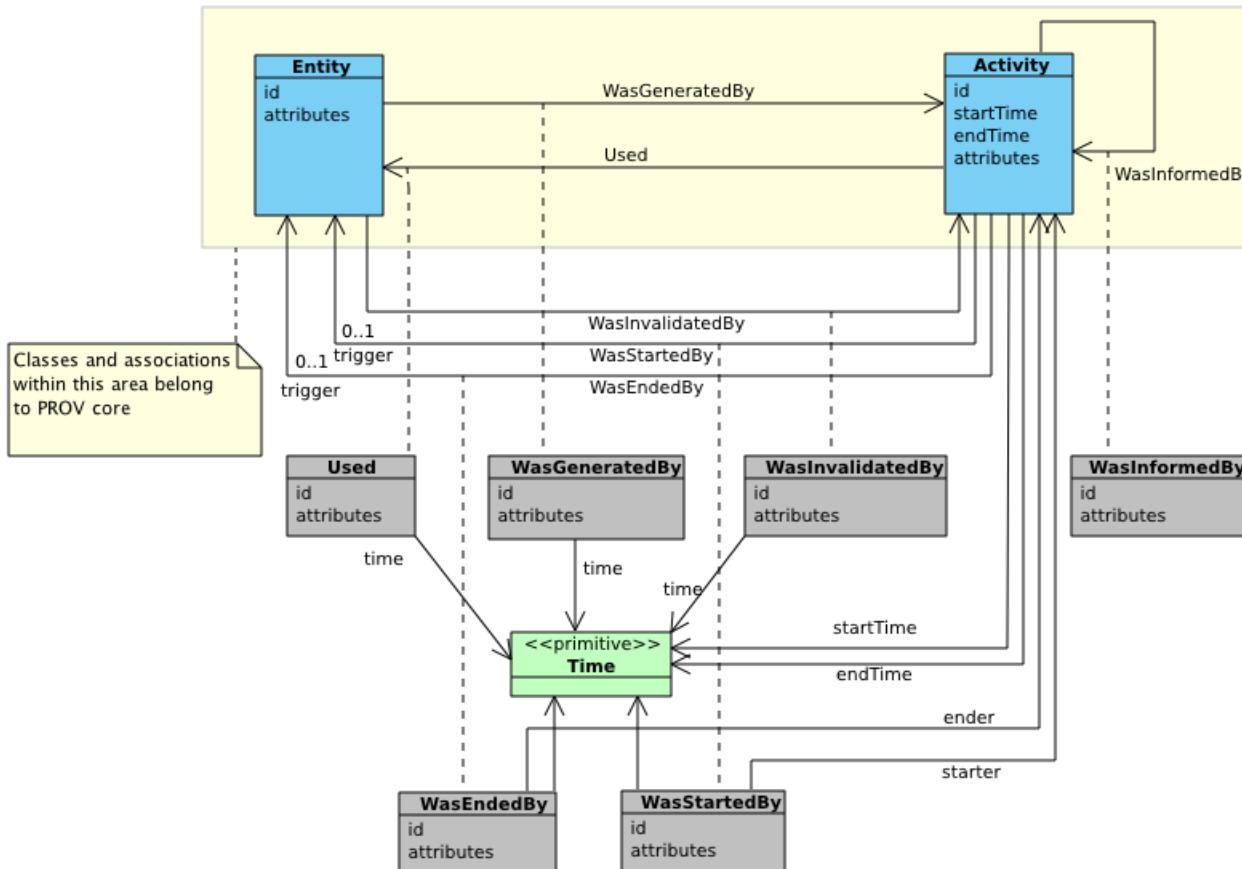


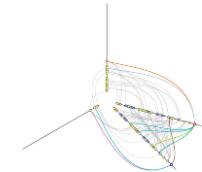


Core vs Extended

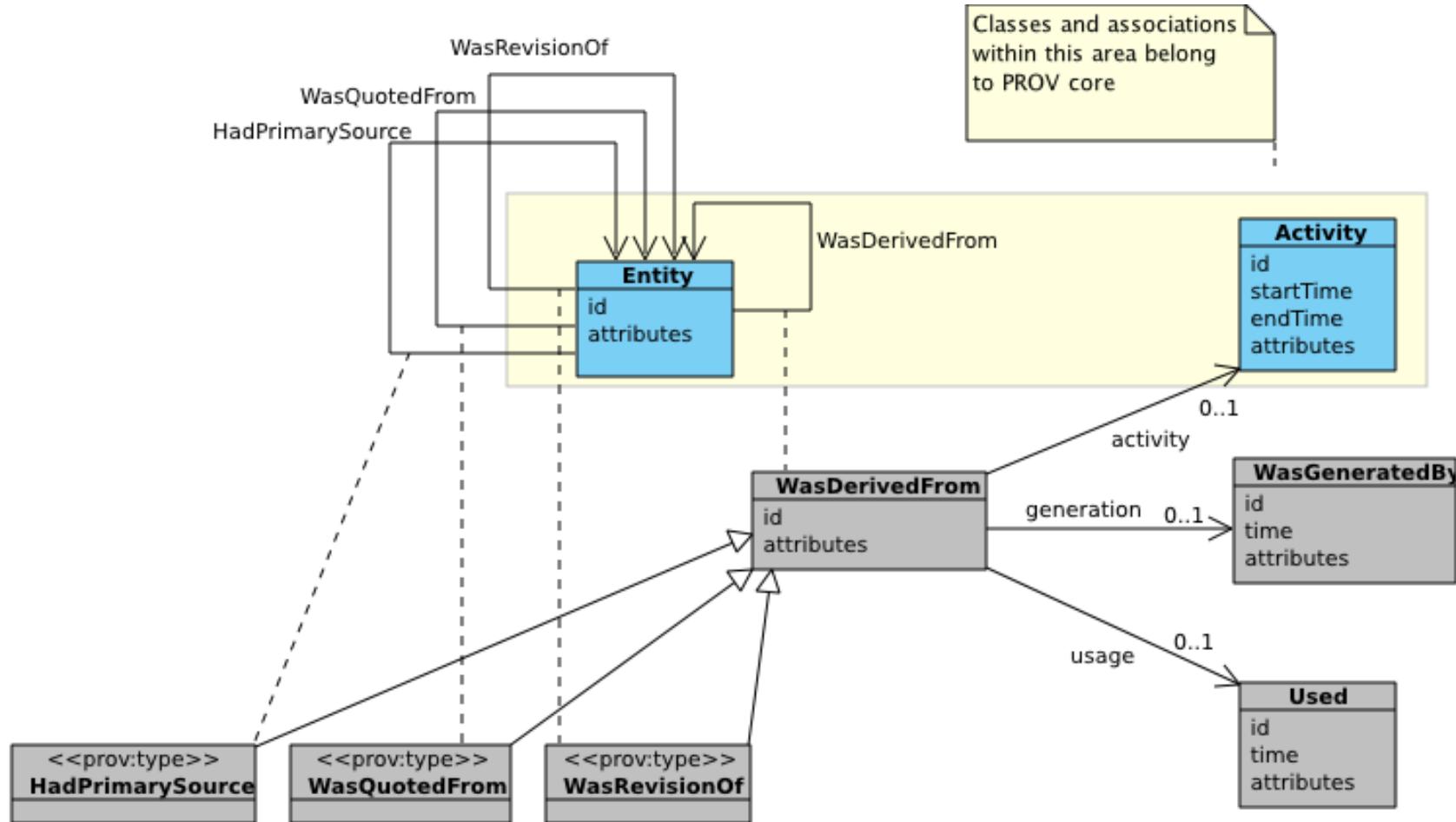


Component 1: Entities and Activities

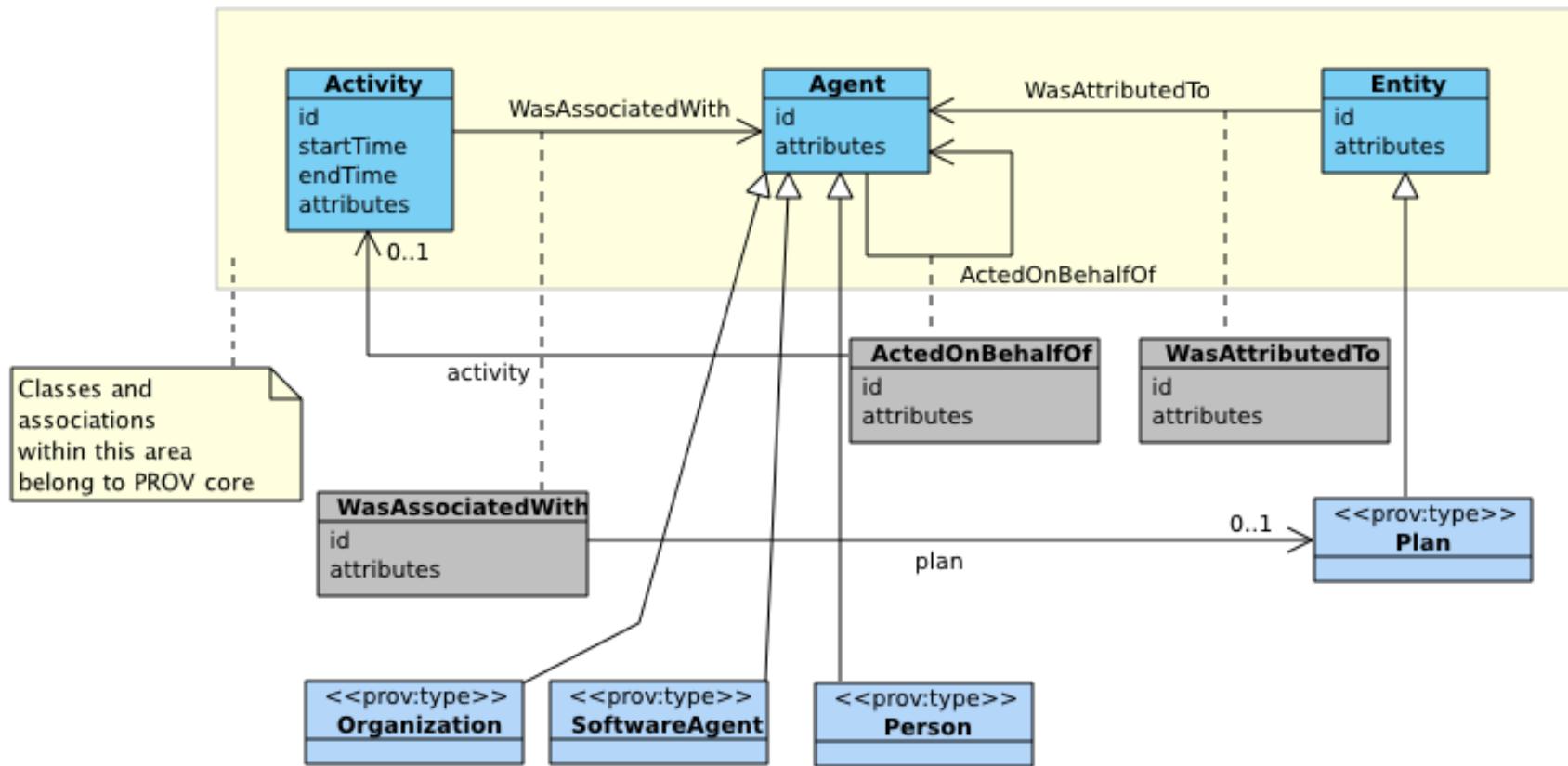
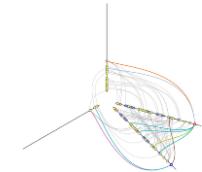


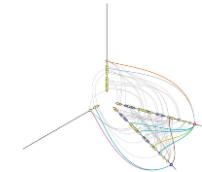


Component 2: Derivation

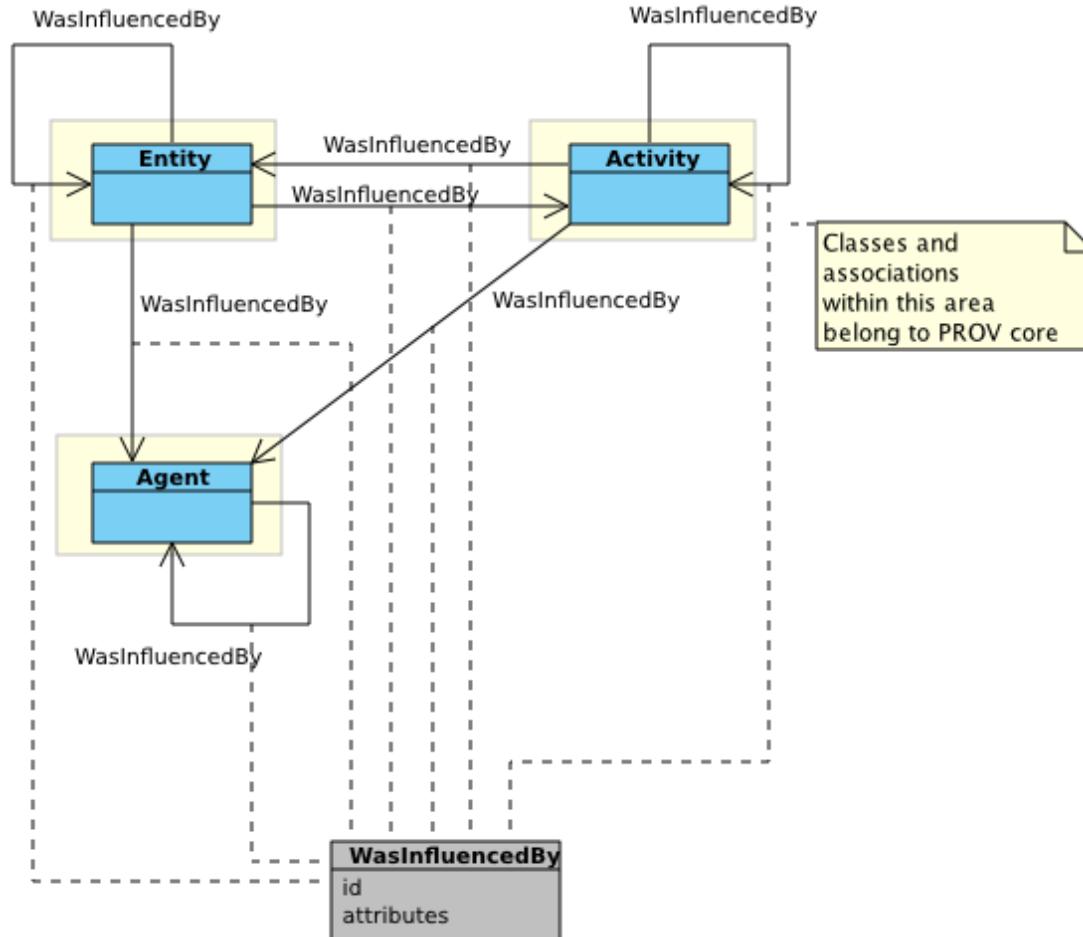


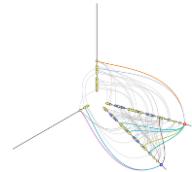
Component 3: Agents, Responsibility...



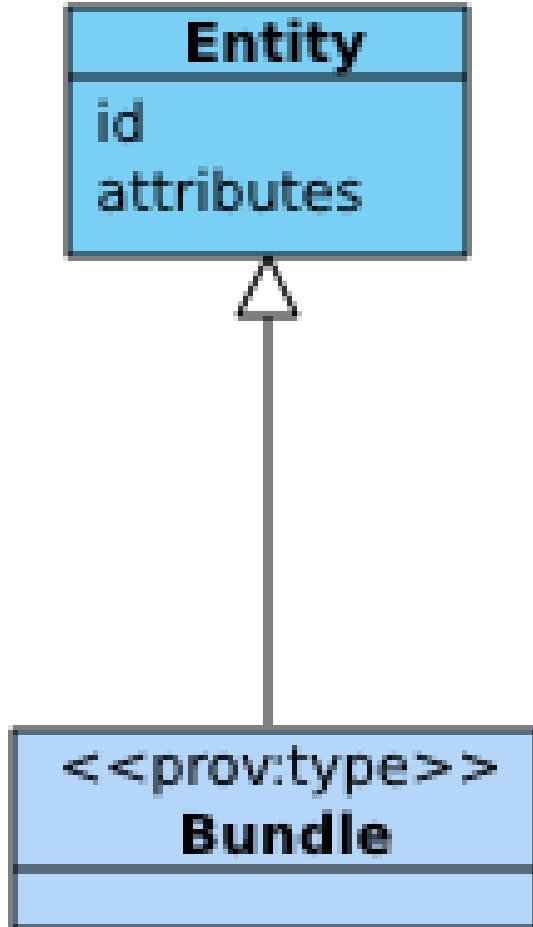


Component 3: ... and Influence

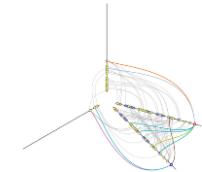




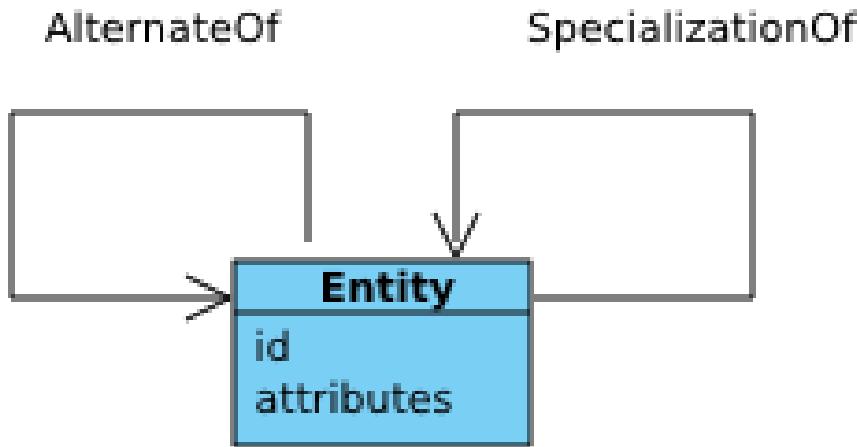
Component 4: Bundles

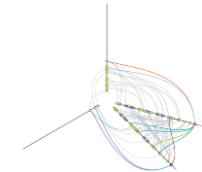


- A Bundle is a named set of provenance assertions
- A Bundle is also an entity
- Its provenance can be expressed with PROV

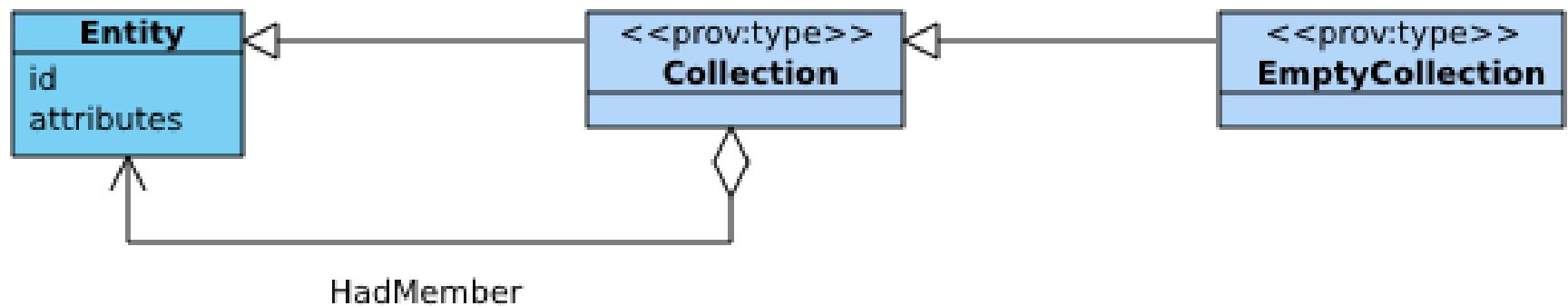


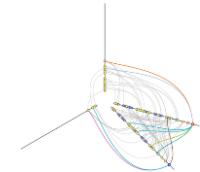
Component 5: Alternate Views



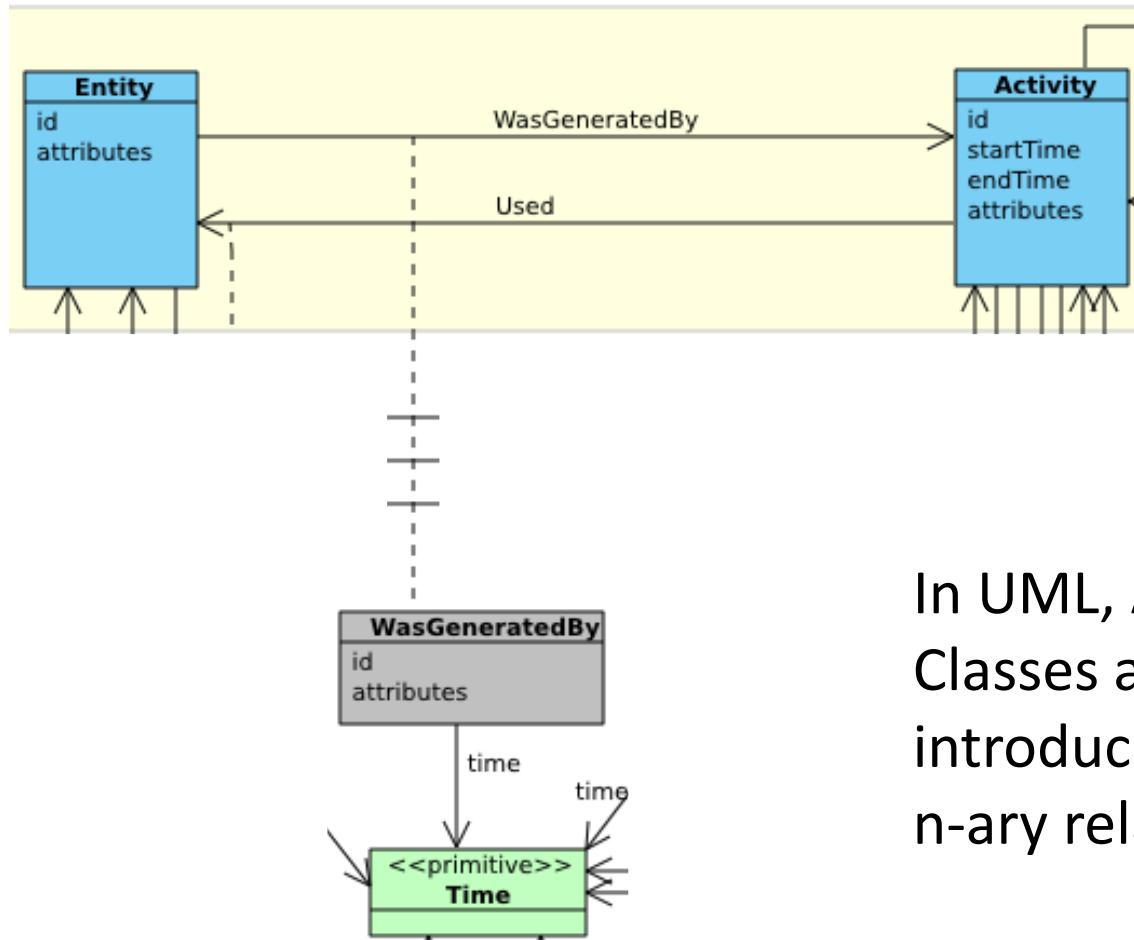


Component 6: Collections

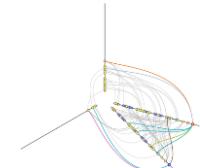




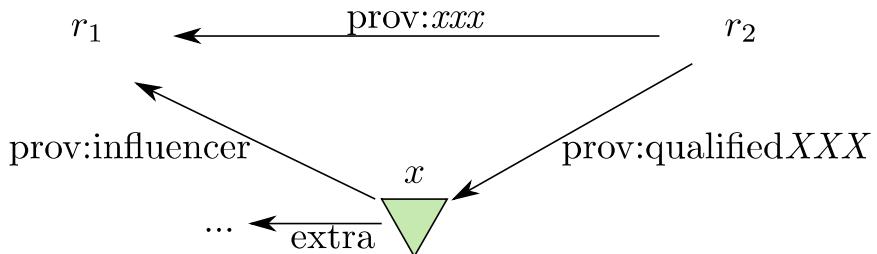
Beyond Binary Relations



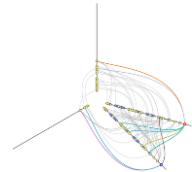
In UML, Association Classes are introduced to express n-ary relations



Directed Qualified Pattern

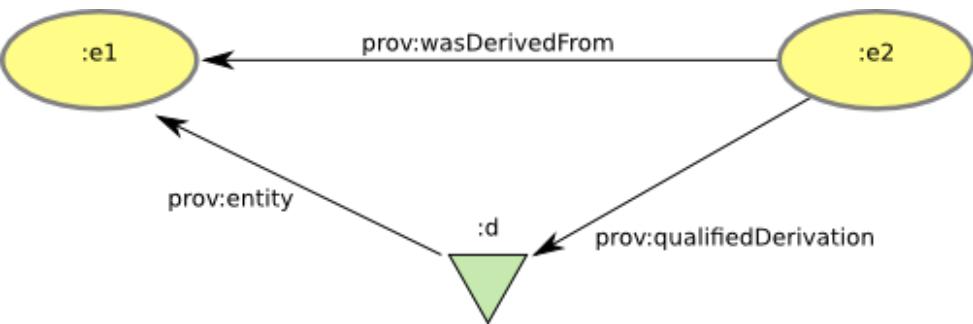


- A convention to express n-ary relations in RDF
- Flows in the same direction as the unqualified binary relation
- Introduces an explicit resource from which extra information can be hooked



Derivation

A derivation is a transformation of an entity into another, an update of an entity resulting in a new one, or the construction of a new entity based on a pre-existing entity.



Unqualified

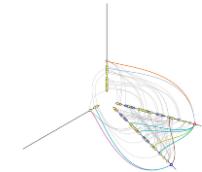
:e2 a prov:Entity.
:e1 a prov:Entity.
:e2 prov:wasDerivedFrom :e1.

Qualified

:d a prov:Derivation.
:e2 prov:qualifiedDerivation :d.
:d prov:entity :e1.

Class	Parent	
prov:Derivation	prov:Influence	
Property	Domain	Range
prov:wasDerivedFrom	prov:Entity	prov:Entity
prov:qualifiedDerivation	prov:Entity	prov:Derivation
prov:entity	prov:Influence	prov:Entity

Derivation



document

prefix ex <http://example/>

entity(ex:e1)

entity(ex:e2)

wasDerivedFrom(ex:e2, ex:e1)

endDocument

```
{  
  "entity": {  
    "ex:e2": {},  
    "ex:e1": {}  
  },  
  "prefix": {  
    "xsd": "http://www.w3.org/2001/XMLSchema",  
    "prov": "http://www.w3.org/ns/prov#",  
    "ex": "http://example/"  
  },  
  "wasDerivedFrom": {  
    "_:wDF15": {  
      "prov:generatedEntity": "ex:e2",  
      "prov:usedEntity": "ex:e1"  
    }  
  }  
}
```

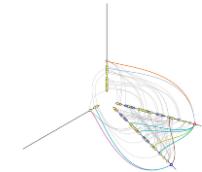
```
@prefix prov: <http://www.w3.org/ns/prov#> .  
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .  
@prefix ex: <http://example/> .
```

```
ex:e1 a prov:Entity .
```

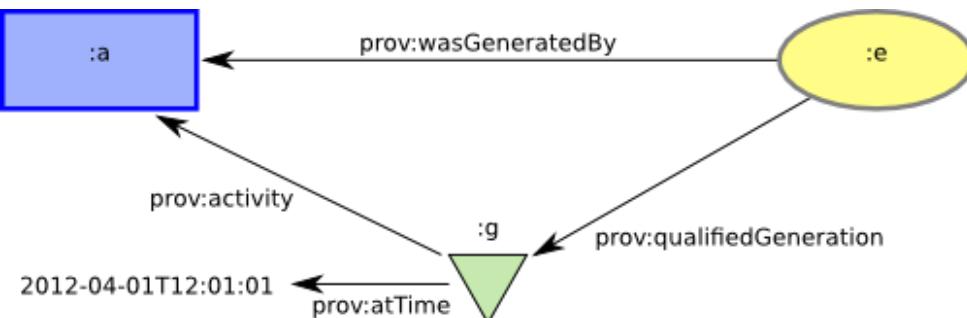
```
ex:e2 a prov:Entity ;  
  prov:wasDerivedFrom ex:e1 .
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<prov:document xmlns:prov="http://www.w3.org/ns/prov#"  
  xmlns:ex="http://example/">  
  <prov:entity prov:id="ex:e1"/>  
  <prov:entity prov:id="ex:e2"/>  
  <prov:wasDerivedFrom>  
    <prov:generatedEntity prov:ref="ex:e2"/>  
    <prov:usedEntity prov:ref="ex:e1"/>  
  </prov:wasDerivedFrom>  
</prov:document>
```

Generation



Generation is the completion of production of a new entity by an activity. This entity did not exist before generation and becomes available for usage after this generation.



Unqualified

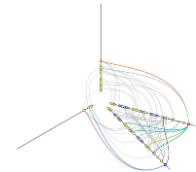
```
:e a prov:Entity.  
:a a prov:Activity.  
:e prov:wasGeneratedBy :a.
```

Qualified

```
:g a prov:Generation.  
:e prov:qualifiedGeneration :g.  
:g prov:activity :a.  
:g prov:atTime "2012-04-01T12:01:01"^^xsd:dateTime.
```

Class	Parent	
prov:Generation	prov:ActivityInfluence, prov:InstantaneousEvent	
Property	Domain	Range
prov:wasGeneratedBy	prov:Entity	prov:Activity
prov:qualifiedGeneration	prov:Entity	prov:Generation
prov:activity	prov:ActivityInfluence	prov:Activity
prov:atTime	prov:InstantaneousEvent	xsd:dateTime

Generation



```
document
prefix ex <http://example/>
entity(ex:e2)
activity(ex:a1)
wasGeneratedBy(ex:e2, ex:a1)
endDocument
```

```
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix ex: <http://example/> .

ex:e2 a prov:Entity .

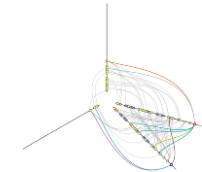
ex:a1 a prov:Activity .

ex:e2 prov:wasGeneratedBy ex:a1 .
```

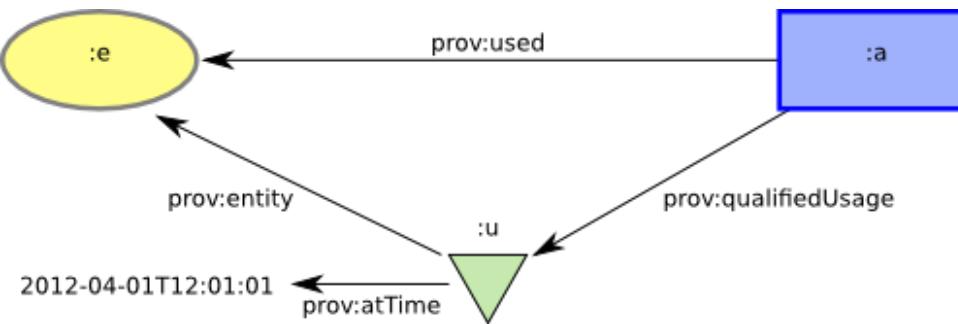
```
{
  "wasGeneratedBy": {
    "_:wGB13": {
      "prov:activity": "ex:a1",
      "prov:entity": "ex:e2"
    }
  },
  "entity": {
    "ex:e2": {}
  },
  "prefix": {
    "xsd": "http://www.w3.org/2001/XMLSchema",
    "prov": "http://www.w3.org/ns/prov#",
    "ex": "http://example/"
  },
  "activity": {
    "ex:a1": {}
  }
}
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<prov:document xmlns:prov="http://www.w3.org/ns/prov#"
                 xmlns:ex="http://example/">
  <prov:entity prov:id="ex:e2"/>
  <prov:activity prov:id="ex:a1"/>
  <prov:wasGeneratedBy>
    <prov:entity prov:ref="ex:e2"/>
    <prov:activity prov:ref="ex:a1"/>
  </prov:wasGeneratedBy>
</prov:document>
```

Usage



Usage is the beginning of utilizing an entity by an activity. Before usage, the activity had not begun to utilize this entity and could not have been affected by the entity.



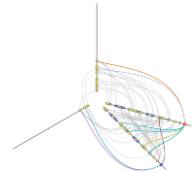
Unqualified

```
:e a prov:Entity.  
:a a prov:Activity.  
:a prov:used :e.
```

Qualified

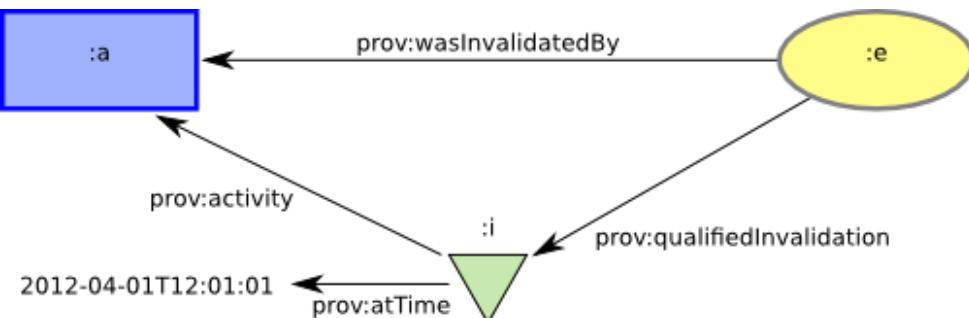
```
:u a prov:Usage.  
:a prov:qualifiedUsage :u.  
:u prov:entity :e.  
:u prov:atTime "2012-04-  
01T12:01:01"  
^^xsd:dateTime.
```

Class	Parent	
prov:Usage	prov:EntityInfluence, prov:InstantaneousEvent	
Property	Domain	Range
prov:used	prov:Activity	prov:Entity
prov:qualifiedUsage	prov:Activity	prov:Usage
prov:entity	prov:EntityInfluence	prov:Entity
prov:atTime	prov:InstantaneousEvent	xsd:dateTime



Invalidation

Invalidation is the start of the destruction, cessation, or expiry of an existing entity by an activity. The entity is no longer available for use (or further invalidation) after invalidation.



Unqualified

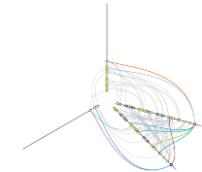
```
:e a prov:Entity.  
:a a prov:Activity.  
:e prov:wasInvalidatedBy :a.
```

Qualified

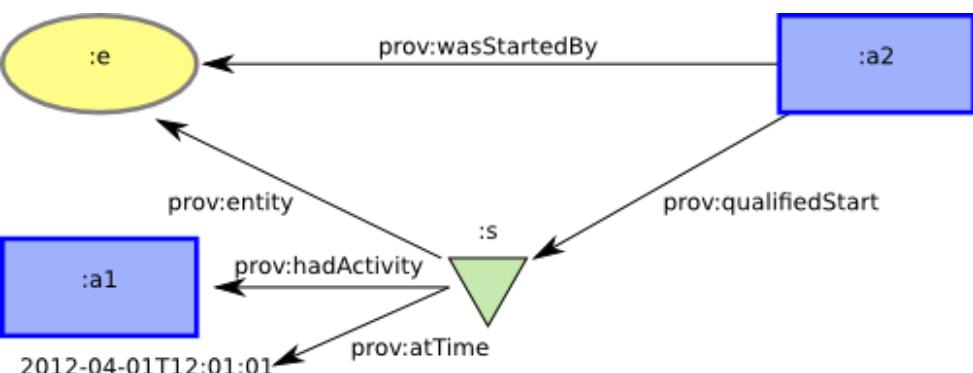
```
:i a prov:Invalidation.  
:e prov:qualifiedInvalidation :i.  
:i prov:activity :a.  
:i prov:atTime "2012-04-  
01T12:01:01"^^xsd:dateTime.
```

Class	Parent	
prov:Invalidation	prov:ActivityInfluence, prov:InstantaneousEvent	
Property	Domain	Range
prov:wasInvalidatedBy	prov:Entity	prov:Activity
prov:qualifiedInvalidation	prov:Entity	prov:Invalidation
prov:activity	prov:ActivityInfluence	prov:Activity
prov:atTime	prov:InstantaneousEvent	xsd:dateTime

Start



Start is when an activity is deemed to have been started by an entity, known as trigger. The activity did not exist before its start.



Unqualified

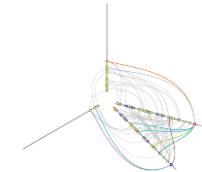
:a2 a prov:Activity.
:e a prov:Entity.
:a2 prov:wasStartedBy :e.

Qualified

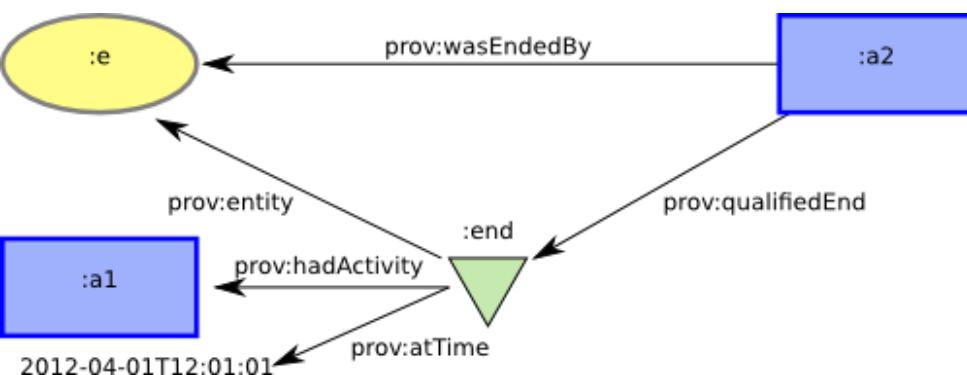
:s a prov:Start.
:a2 prov:qualifiedStart :s.
:s prov:entity :e.
:s prov:hadActivity :a1.
:s prov:atTime "2012-04-01T12:01:01"^^xsd:dateTime.

Class	Parent	
prov:Start	prov:EntityInfluence, prov:InstantaneousEvent	
Property	Domain	Range
prov:wasStartedBy	prov:Activity	prov:Entity
prov:qualifiedStart	prov:Activity	prov:Start
prov:entity	prov:EntityInfluence	prov:Entity
prov:hadActivity	prov:Influence	prov:Activity
prov:atTime	prov:InstantaneousEvent	xsd:dateTime

End



End is when an activity is deemed to have been ended by an entity, known as trigger. The activity no longer exists after its end.



Unqualified

```

:a2 a prov:Activity .
:e a prov:Entity .
:a2 prov:wasEndedBy :e .
  
```

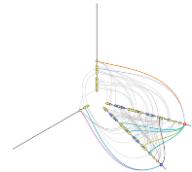
Qualified

```

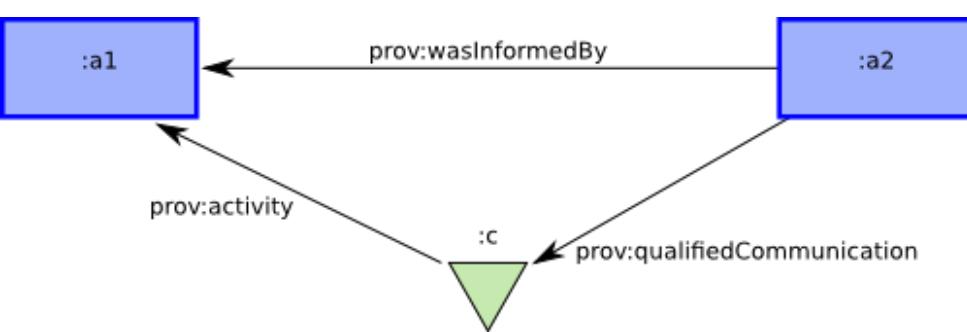
:end a End .
:a2 prov:qualifiedEnd :end .
:end prov:entity :e .
:end prov:hadActivity :a1 .
:end prov:atTime "2012-04-01T12:01:01"^^xsd:dateTime .
  
```

Class	Parent	
prov:End	prov:EntityInfluence , prov:InstantaneousEvent	
Property	Domain	Range
prov:wasEndedBy	prov:Activity	prov:Entity
prov:qualifiedEnd	prov:Activity	prov:End
prov:entity	prov:EntityInfluence	prov:Entity
prov:hadActivity	prov:Influence	prov:Activity
prov:atTime	prov:InstantaneousEvent	xsd:dateTime

Communication



Communication is the exchange of some unspecified entity by two activities, one activity using some entity generated by the other.



Unqualified

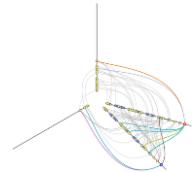
```
:a2 a prov:Activity.  
:a1 a prov:Activity.  
:a2 prov:wasInformedBy :a1 .
```

Qualified

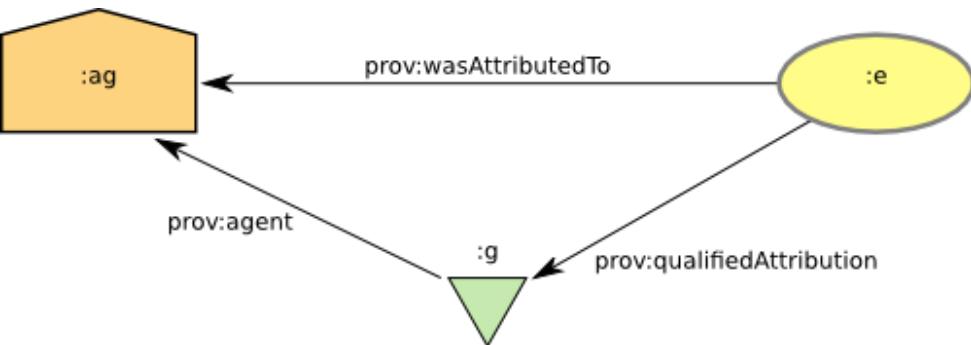
```
:c a prov:Communication.  
:a2  
prov:qualifiedCommunication  
:c .  
:c prov:activity :a1
```

Class	Parent	
prov:Communication		prov:ActivityInfluence
Property	Domain	Range
prov:wasInformedBy	prov:Activity	prov:Activity
prov:qualifiedCommunication	prov:Activity	prov:Communication
prov:activity	prov:ActivityInfluence	prov:Activity

Attribution



Attribution is the ascribing of an entity to an agent.



Unqualified

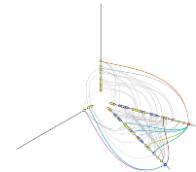
```
:e a prov:Entity.  
:ag a prov:Agent.  
:e prov:wasAttributedTo  
:ag.
```

Qualified

```
:attr a prov:Attribution.  
:e  
prov:qualifiedAttribution  
:attr.  
:attr prov:agent :ag.
```

Class	Parent	
prov:Attribution		prov:AgentInfluence
Property	Domain	Range
prov:wasAttributedTo	prov:Entity	prov:Agent
prov:qualifiedAttribution	prov:Entity	prov:Attribution
prov:agent	prov:AgentInfluence	prov:Agent

Attribution



```
document
prefix ex <http://example/>
entity(ex:e2)
agent(ex:ag1)
wasAttributedTo(ex:e2, ex:ag1)
endDocument
```

```
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix ex: <http://example/> .

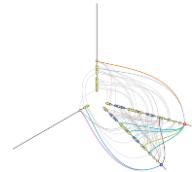
ex:e2 a prov:Entity .

ex:ag1 a prov:Agent .

ex:e2 prov:wasAttributedTo ex:ag1 .
```

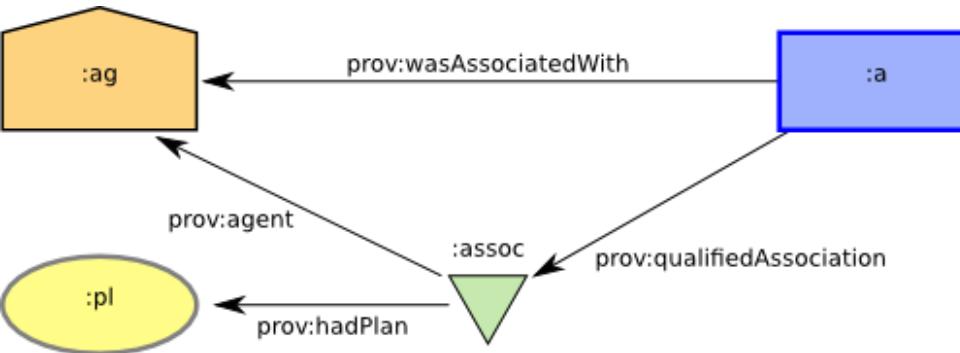
```
{
  "wasAttributedTo": {
    "_:wAT5": {
      "prov:agent": "ex:ag1",
      "prov:entity": "ex:e2"
    }
  },
  "entity": {
    "ex:e2": {}
  },
  "prefix": {
    "xsd": "http://www.w3.org/2001/XMLSchema",
    "prov": "http://www.w3.org/ns/prov#",
    "ex": "http://example/"
  },
  "agent": {
    "ex:ag1": {}
  }
}
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<prov:document xmlns:prov="http://www.w3.org/ns/prov#"
                 xmlns:ex="http://example/">
  <prov:entity prov:id="ex:e2"/>
  <prov:agent prov:id="ex:ag1"/>
  <prov:wasAttributedTo>
    <prov:entity prov:ref="ex:e2"/>
    <prov:agent prov:ref="ex:ag1"/>
  </prov:wasAttributedTo>
</prov:document>
```



Association

An association is an assignment of responsibility to an agent for an activity, indicating that the agent had a role in the activity. It further allows for a plan to be specified.,



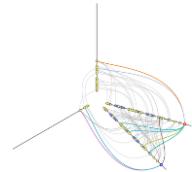
Unqualified

```
:a a prov:Activity.  
:ag a prov:Agent.  
:a prov:wasAssociatedWith  
:ag.
```

Qualified

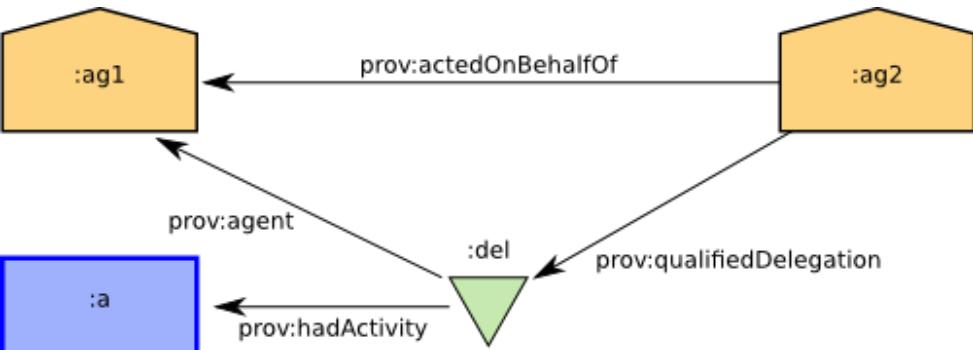
```
:assoc a prov:Association.  
:pl a prov:Plan.  
:a prov:qualifiedAssociation  
:assoc.  
:assoc prov:agent :ag.  
:assoc prov:hadPlan :pl.
```

Class	Parent	
prov:Association	prov:AgentInfluence	
prov:Plan	prov:Entity	
Property	Domain	Range
prov:wasAssociatedWith	prov:Activity	prov:Agent
prov:qualifiedAssociation	prov:Activity	prov:Association
prov:agent	prov:AgentInfluence	prov:Agent
prov:hadPlan	prov:Association	prov:Plan



Delegation

Delegation is the assignment of authority and responsibility to an agent to carry out a specific activity as a delegate or representative.



Unqualified

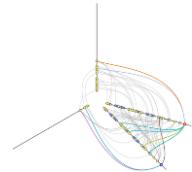
```
:ag2 a prov:Agent.  
:ag1 a prov:Agent.  
:ag2 prov:actedOnBehalfOf  
:ag1.
```

Qualified

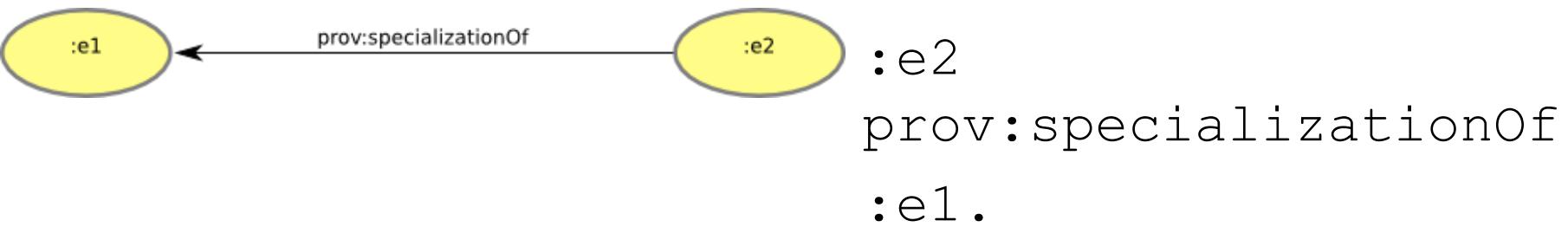
```
:del a prov:Delegation.  
:a a prov:Activity.  
:ag2  
    prov:qualifiedDelegation  
    :del.  
:del prov:agent :ag1.  
:del prov:hadActivity :a.
```

Class	Parent	
prov:Delegation	prov:AgentInfluence	
Property	Domain	Range
prov:actedOnBehalfOf	prov:Agent	prov:Agent
prov:qualifiedDelegation	prov:Agent	prov:Delegation
prov:agent	prov:AgentInfluence	prov:Agent
prov:hadActivity	prov:Delegation	prov:Activity

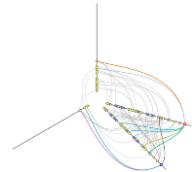
Specialization



An entity that is a specialization of another shares all aspects of the latter, and additionally presents more specific aspects of the same thing as the latter.

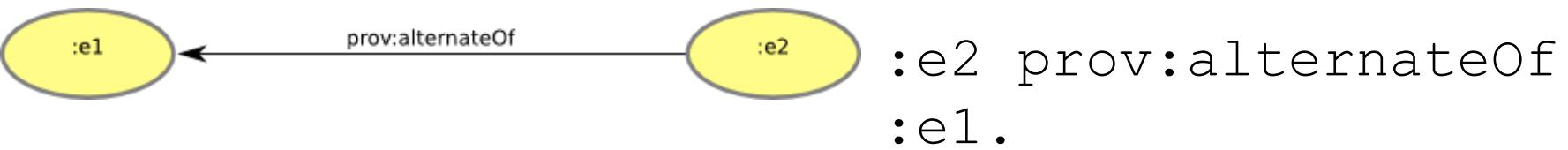


Property	Domain	Range
<code>prov:specializationOf</code>	<code>prov:Entity</code>	<code>prov:Entity</code>



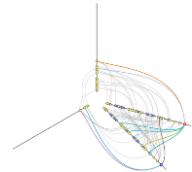
Alternate

An entity that is a specialization of another shares all aspects of the latter, and additionally presents more specific aspects of the same thing as the latter.



Property	Domain	Range
prov:alternateOf	prov:Entity	prov:Entity

Membership

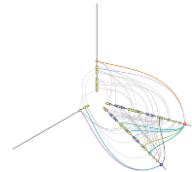


A collection is an entity that provides a structure to some constituents that must themselves be entities. These constituents are said to be member of the collections.



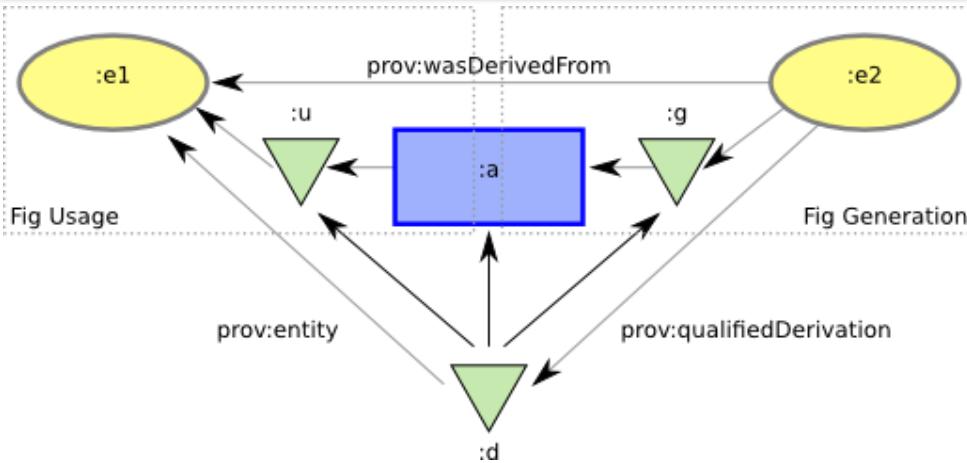
:c a prov:Collection.
:e a prov:Entity.
:c prov:hadMember :e.

Class	Parent	
prov:Collection	prov:Entity	
Property	Domain	Range
prov:hadMember	prov:Collection	prov:Entity



Refined Derivation

A derivation is a transformation of an entity into another, an update of an entity resulting in a new one, or the construction of a new entity based on a pre-existing entity.



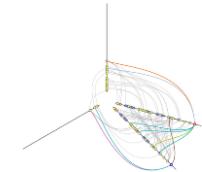
Unqualified

`:e2 a prov:Entity.`
`:e1 a prov:Entity.`
`:e2 prov:wasDerivedFrom :e1.`

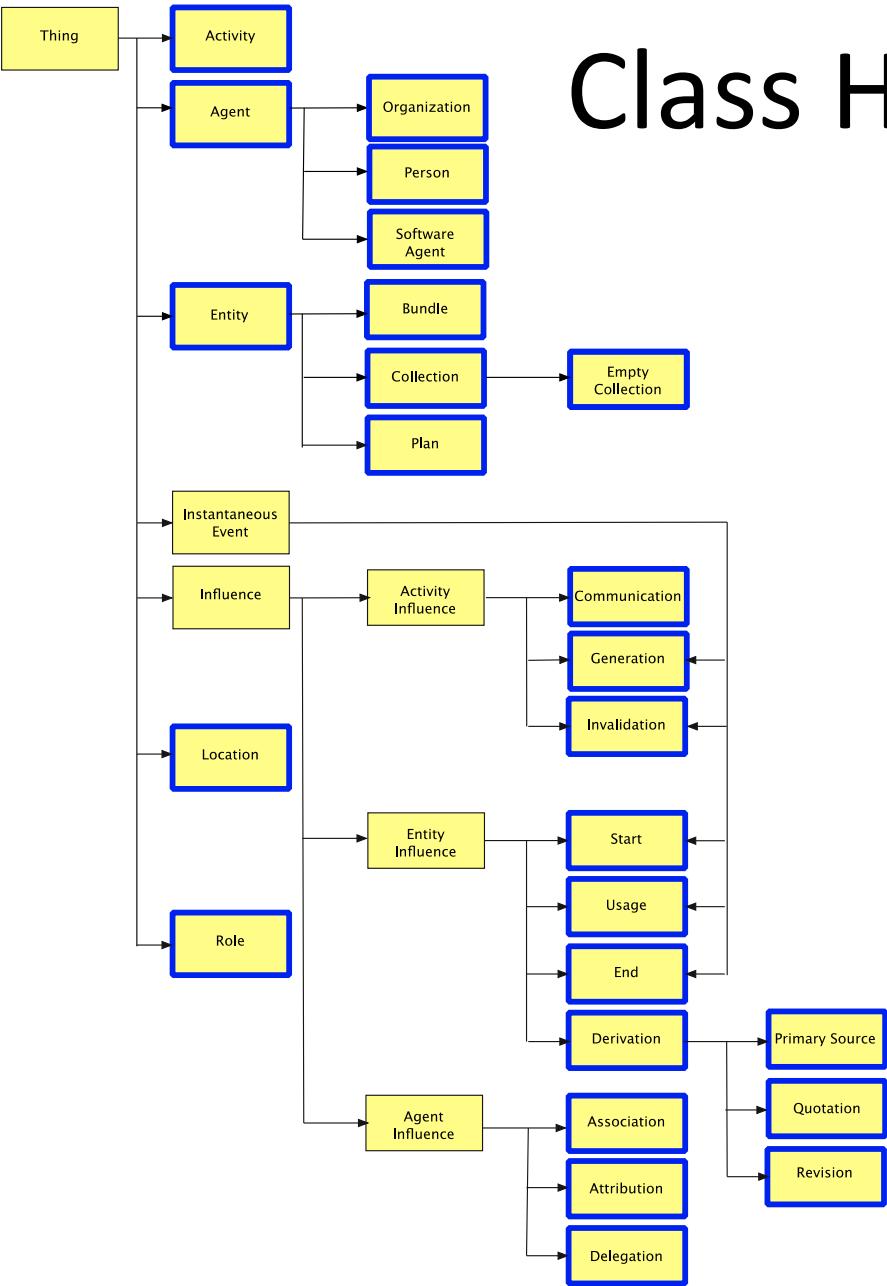
Qualified

`:d a prov:Derivation.`
`:e2 prov:qualifiedDerivation :d.`
`:d prov:entity :e1.`
`:d prov:hadActivity :a.`
`:d prov:hadGeneration :g.`
`:e2 prov:hadQualifiedGeneration :g.`
`:d prov:hadUsage :u.`
`:a prov:hadQualifiedUsage :u.`

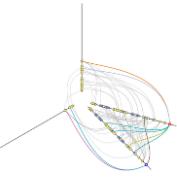
Class	Parent	
Property		
<code>prov:Derivation</code>	<code>prov:EntityInfluence</code>	
Property	Domain	Range
<code>prov:hadActivity</code>	<code>prov:Influence</code>	<code>prov:Activity</code>
<code>prov:hadGeneration</code>	<code>prov:Derivation</code>	<code>prov:Generation</code>
<code>prov:hadUsage</code>	<code>prov:Derivation</code>	<code>prov:Usage</code>

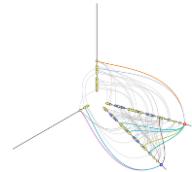


Class Hierarchy

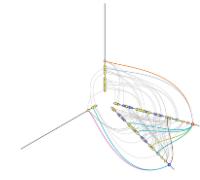


- A few classes are introduced to provide structure (e.g. **InstantaneousEvent**)
- A few subclasses are introduced for interoperability (e.g. **Person**)
- Developers are invited to extend classes with a blue border (to ensure interoperability)

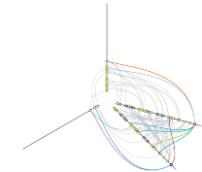




```
provbook:a-little-provenance-goes-a-long-way a prov:Entity;
prov:value "A little provenance goes a long way";
prov:wasAttributedTo provbook:Paul, provbook:Luc;
prov:wasDerivedFrom <http://www.cs.rpi.edu/~hendler/LittleSemanticsWeb.html>.
```



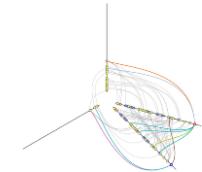
RECIPES



Provenance Recipes

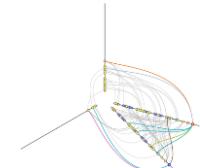
- Modelling
- Organizing
- Collecting
- Anti-Patterns





Modeling

- 1.1 Iterative Modeling
- 1.2 Identify, Identify, Identify!
- 1.3 From data flow to activities
- **1.4 Plan for revisions**
- 1.5 Modeling replacement and other destructive activities
- 1.6 Modeling message passing
- 1.7 Modeling parameters
- 1.8 Introduce the execution environment
- 1.9 Modeling sub-activities



1.4 Plan For Revision

Q

How does one express revisions to a resource or document using PROV?

S

Collect all the revisions of a resource under a single general resource using prov:specializationOf.

- For a resource create an identifier to denote it in general irrespective of its actual version.
 - For example, the BBC news website.
- For each revision of the resource, create a fresh identifier for that resource.
 - For example, the BBC news website on a given day.
- Relate the reversion to the previous one using prov:wasRevisionOf.
- Relate each version to the resource in general using prov:specializationOf.

1.4 Plan For Revision

document

prefix ex <http://example/>

entity(ex:e)

entity(ex:e0)

entity(ex:e1)

entity(ex:e2)

entity(ex:e3)

specializationOf(ex:e0,ex:e)

specializationOf(ex:e1,ex:e)

specializationOf(ex:e2,ex:e)

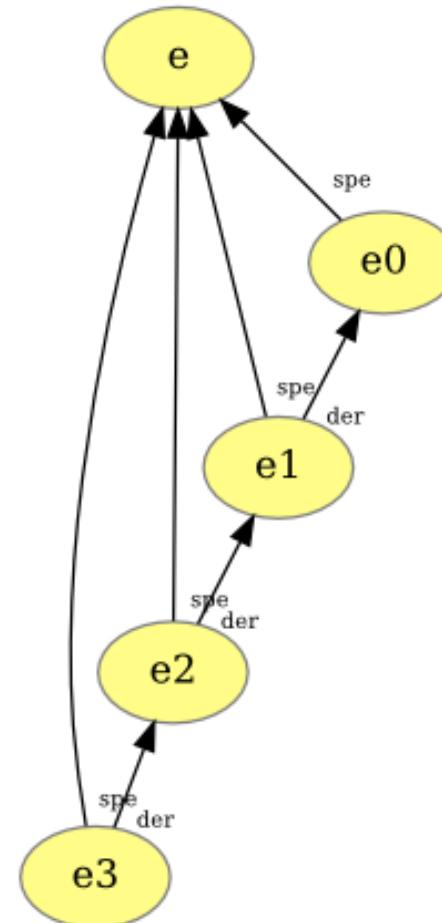
specializationOf(ex:e3,ex:e)

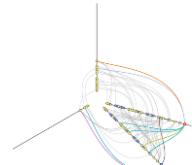
wasDerivedFrom(ex:e1,ex:e0)

wasDerivedFrom(ex:e2,ex:e1)

wasDerivedFrom(ex:e3,ex:e2)

endDocument

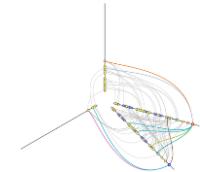




Organizing

- 2.1 Stitch provenance together
- 2.2 Use Content-Negotiation when Exposing Provenance
- 2.3 Bundle up and Provide Attribution to Your Provenance
- 2.4 Embedding Provenance in HTML
- 2.5 Embedding Provenance in other Media
- 2.6 When all else fails, add provenance to http Headers
- 2.7 Embed Provenance in Bundles: Self-Referential Bundles
- 2.8 When displaying provenance, adopt conventional layout

2.2 Use Content-Negotiation When Exposing Provenance



Q

What serialization should I provide provenance in?

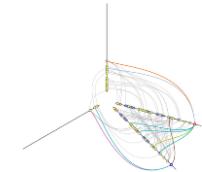
S

PROV was designed to supply provenance in multiple representations catering for different types of development platforms whether they be enterprise XML-based applications or RDF-based Semantic Web applications. Thus, providers should ensure their provenance is useful for a variety of needs by supplying the same provenance data in multiple representations using content-negotiation.

For example, for the provenance located at <http://www.provbook.org/provenance>, the following lists the curl commands to retrieve 3 different representations of the provenance (namely, turtle, svg, and xml):

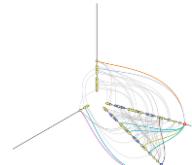
- curl -sH "Accept: text/turtle" -L <http://www.provbook.org/provenance>
- curl -sH "Accept: image/svg+xml" -L <http://www.provbook.org/provenance>
- curl -sH "Accept: application/provenance+xml" -L <http://www.provbook.org/provenance>

The provenance information corresponding to a particular format can either be generated up-front or produced on the fly. Most web servers and environments support content negotiation.



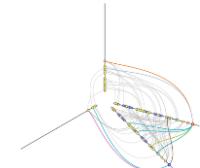
Collecting

- 3.1 Use structured logs to collect provenance
- 3.2 Collect in a local form, expose as PROV



Anti-patterns

- 4.1 Activity but no Derivation
- 4.2 Association but no Attribution
- 4.3 Don't identify prov:Agent, identify responsibility properties



4.1 Activity but no Derivation

Q

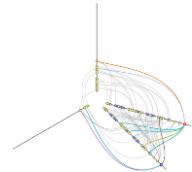
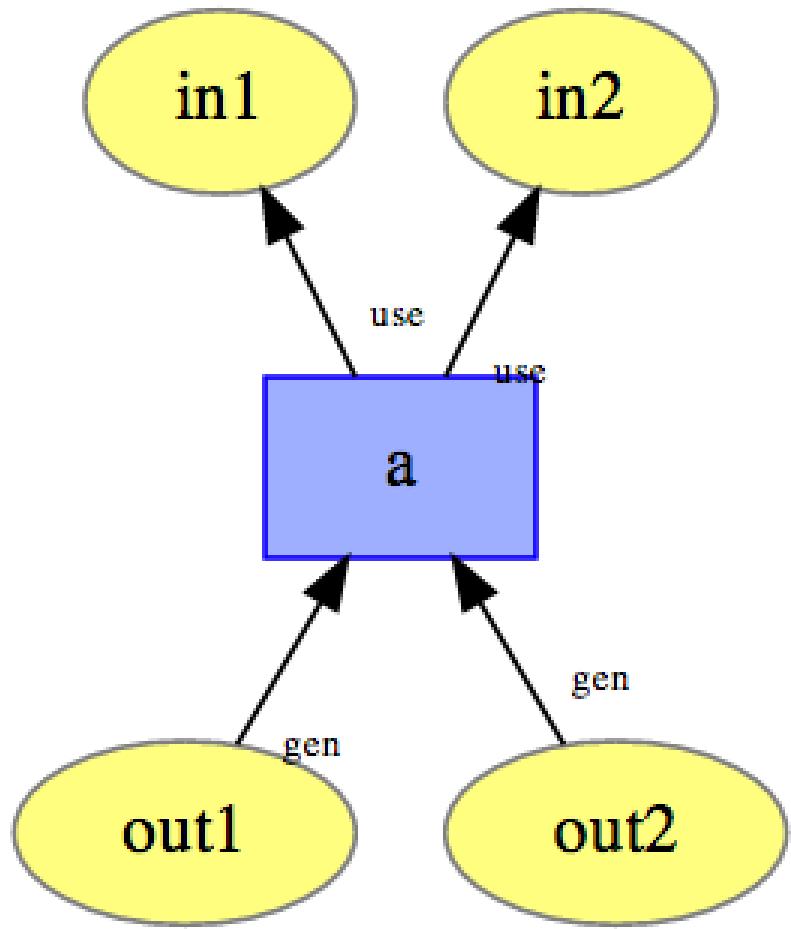
Is there a connection between an entity generated by an activity and the entities used by that activity?

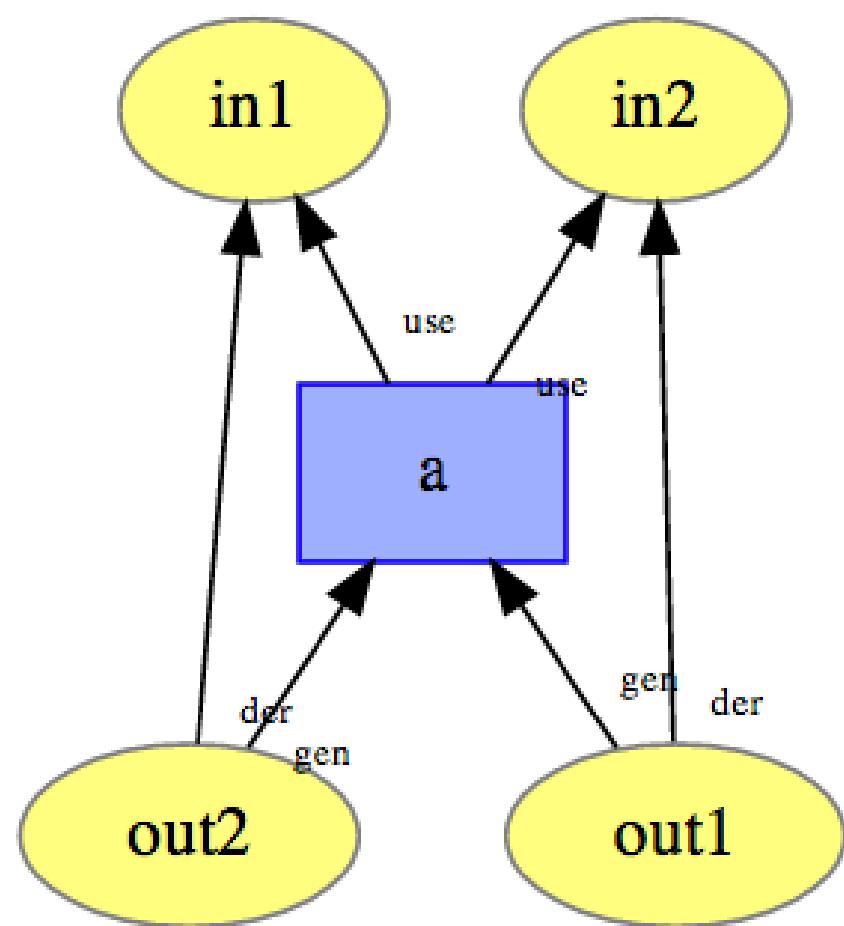
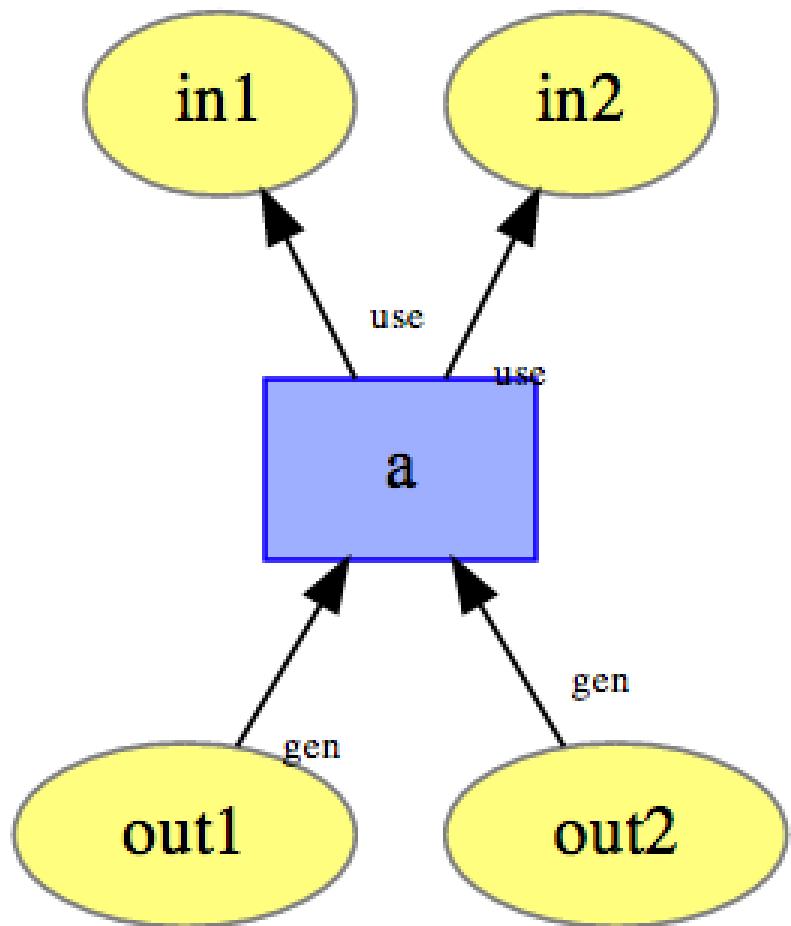
S

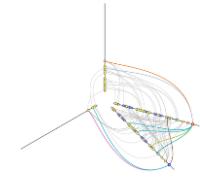
There are two solutions to this problem.

- One is to introduce explicit derivation edges between entities that are dependent on one another.
- The other is to define an application-specific extension to prov:Activity (e.g. Function) that defines each output as being derived from all inputs.

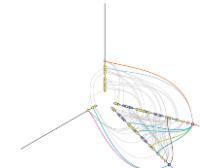
The first solution is somewhat preferable as it makes the derivations explicit and is more interoperable.







TOOLS



provenance.ecs.soton.ac.uk

provenance.ecs.soton.ac.uk

w3c iGoogle ProvExtract google maps patina inaugural Functional

Southampton Provenance Suite

Home Publications Contact

Validator

A RESTful web service that validates PROV descriptions against the PROV Constraints specification. Supports uploading PROV by URL, file upload or inline statements.

[Validator](#)

Translator

Translates between different representations of PROV. Supports PROV-N, PROV-XML, PROV-O and PROV-JSON.

[Translator](#)

Store

A provenance repository that allows storing, browsing, and managing provenance documents via a Web interface or a REST API.

[Store](#)

Applications

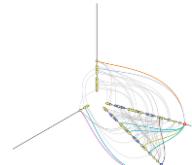
- [CollabMap](#) - a platform for crowdsourcing the task of identifying building evacuation routes
- [AgentSwitch](#) - a personalized energy tariff-recommender system
- [PoN](#) - an experimental web application for collecting and organizing research data and notes

Tools

- [ProvToolbox](#) - a Java toolbox for handling PROV
- [Prov Python](#) - a Python implementation of the PROV data model
- [ProvExtract](#) - for dealing with PROV embedded in web pages
- [ProvVis](#) - experimental visualizations of PROV

PROV

- [Overview of PROV](#)
- [PROV Model Primer](#)
- [PROV-O](#)
- [PROV-DM](#)
- [PROV-N](#)
- [PROV Constraints](#)
- [Provenance Working Group at W3C](#)
- [PROV-JSON](#)



ProvValidator

[ProvValidator](#) [ProvTranslator](#) [API](#) [About](#) [Contact](#)

Select a file:

No file chosen

Enter a URL:

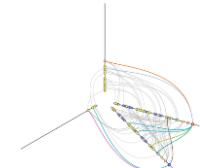


Enter PROV statements:

ttl rdf

provn xml

trig json



ProvTranslator

[ProvValidator](#) **ProvTranslator** [API](#) [About](#) [Contact](#)

Select a file:

Choose File No file chosen

Enter a URL:

Enter PROV statements:

- ttl rdf
- provn provx
- trig json

json

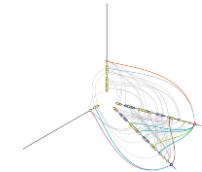
provx

provn

turtle

trig

svg



ProvStore

Welcome to Prov Store

Prov Store is a free service for storing, viewing and collaborating on [provenance documents](#) together with others or in private. You can browse public documents without registration or register to store your own.

[Sign Up Now](#)[Find out more](#)

Create

Store documents and collaborate on them with others using group-based or individual access rights.

Query

Search for specific documents or subsets of graphs using our query and search system.

Share

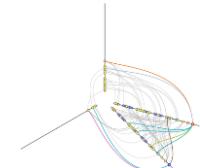
Host your documents on our servers so others can trace provenance of your entities at a centralized location.

Analyze

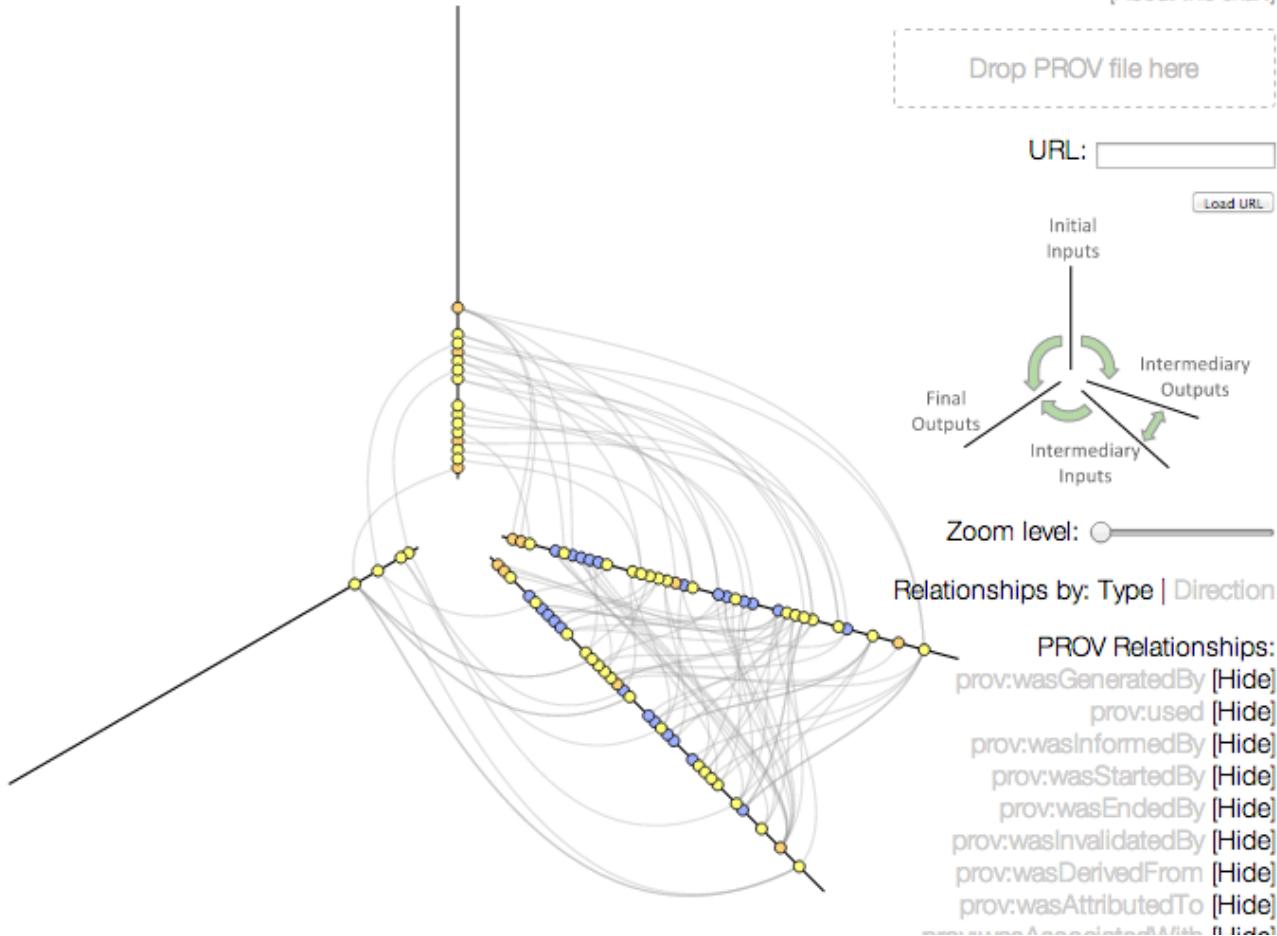
We gather statistics about your data to give you valuable insights.

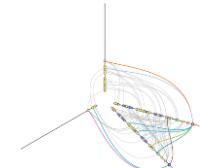
Connect

Query, insert and update graphs using our REST API and connect to your apps using OAuth support.



ProvVis





ProvToolbox

GitHub

This repository ▾

Search or type a command

?

Explore

Features

Enterprise

Blog



lucmoreau / ProvToolbox



Java toolbox to create and convert W3C PROV data model representations

1,440 commits

8 branches

14 releases

7 contributors



branch: master ▾

ProvToolbox / +

using @Lob instead of string



lucmoreau authored a month ago

latest commit 4bb3de2062 ↗



eclipse prefs

a year ago



[maven-release-plugin] prepare for next development iteration

3 months ago



Embedded jvnet.Equals/ToString interfaces and implementation in prov...

2 months ago



added known namespaces in prov-json importer

2 months ago



Embedded jvnet.Equals/ToString interfaces and implementation in prov...

2 months ago



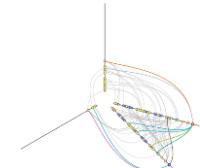
[maven-release-plugin] prepare for next development iteration

3 months ago



Embedded jvnet.Equals/ToString interfaces and implementation in prov...

2 months ago



ProvPy

 python™

» Package Index > prov > 0.5.3

PACKAGE INDEX »

Browse packages
Package submission
List trove classifiers
List packages
RSS (latest 40 updates)
RSS (newest 40 packages)
Python 3 Packages
PyPI Tutorial
PyPI Security
PyPI Support
PyPI Bug Reports
PyPI Discussion
PyPI Developer Info

ABOUT »

NEWS »

DOCUMENTATION »

DOWNLOAD »

COMMUNITY »

FOUNDATION »

CORE DEVELOPMENT »

Not Logged In

[Login](#)
[Register](#)
[Lost Login?](#)
[Use OpenID](#)    

Status

[Nothing to report](#)

prov 0.5.3

A Python implementation of PROV data model, providing simple provenance tracking and persistence using Django ORM.

[Download prov-0.5.3.tar.gz](#)

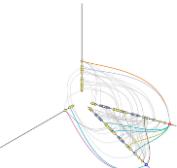
WARNING: Under active development

This package provides an implementation of the [PROV Data Model](#) in Python. It contains a number of sub-modules:

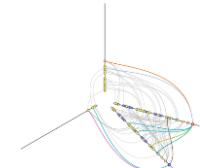
- prov.model: In-memory classes for PROV assertions. *ProvBundle.JSONEncoder* and *ProvBundle.JSONDecoder* provide JSON serialisation and deserialisation for a *ProvBundle* in the [PROV-JSON representation](#). In addition, the *prov.model.graph* module exports PROV documents into graphical formats (e.g. PDF, PNG, SVG).
- prov.persistence: A Django app for storing and loading *ProvBundle* instances to/from databases using Django ORM
- prov.tracking: a logging-like module to facilitate tracking provenance in Python programs

See [prov/model/test/examples.py](#) for example usages.

Deployment: The package was used to build [ProvStore](#), a repository for provenance documents.



Even More...



Git2Prov

Git2PROV

About

Contact



Enter a Git Repo:

Choose a serialization:

PROV-JSON

PROV-N

PROV-O

SVG

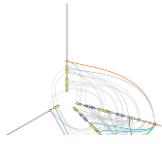
Download

Powered by:



iMinds
CONNECT.INNOVATE.CREATE





PROV-O-Viz Visualizing the Train of Thought

Copyright (c) 2013, Rinke Hoekstra, VU University Amsterdam

About PROV-O-Viz

With PROV-O-Viz you can visualize *any* provenance graph expressed using the [W3C PROV-O vocabulary](#) as an intuitive [Sankey diagram](#). Really neat.

PROV-O-Viz was developed by the [Data2Semantics](#) project, funded under the [COMMIT/ programme](#).

How to use it

To see what it does, you can point PROV-O-Viz to your own SPARQL endpoint, containing Named Graphs with PROV-O activities and entities in it.

Alternatively, you can simply copy and paste some of PROV-O in Turtle format in the Paste-And-Go form.

Finally, you can POST PROV-O in Turtle format to the PROV-O-Viz service, and we'll return a self-contained HTML snippet for you to include in your website.

- The service lives at <http://provoviz.org/service>, and
- expects a POST with a `graph_uri` parameter (a URI that identifies your provenance graph), and

Go on, give us some PROV!

Paste-And-Go

SPARQL Endpoint

Paste-And-Go form

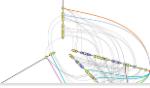
Paste a Turtle serialization of a PROV-O graph in the text area below.

PROV-O input

Go

Example

Use the most excellent Git2PROV we developed together with the University of Ghent to generate PROV-O from your favorite Git repository. The output of Git2PROV can simply be copied to the text area.



PROV-O-Matic

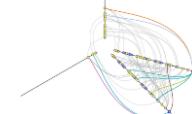
Python Provenance Tracer

** Author: ** Rinke Hoekstra, VU University Amsterdam, rinke.hoekstra@vu.nl

Provenance is key in improving the transparency of scientific data publishing.

PROV-O-Matic provides three things:

- a **decorator** for functions and methods that builds an RDF PROV-O representation of the inputs and outputs of the respective function. The provenance trace is persistent within a Python session. And,
- it integrates provenance tracing in IPython Notebook, a tool frequently used by scientists for analysing data, and reporting on it. All functions defined in the notebook are automatically decorated, and all executions of steps in the notebook are recorded as well (including changing variable values). And
- it connects to **PROV-O-Viz** for interactive visualization of the provenance graph, and integrates it into IPython notebook.



Taverna PROV support

This is a plugin for the [Taverna](#) Workbench and Taverna Command Line which allows export of the provenance of a workflow run according to the [W3C PROV-O standard](#).

Source code and license

This plugin is distributed under the [GNU Lesser General Public License 2.1](#) (LGPL). The source code for this plugin is available at <https://github.com/taverna/taverna-prov>

Installation for Taverna workbench

Installation of an release mechanism.



WINGS

You need:

- [Taverna Workben](#)
- Java 1.7 (now co

[Home](#)[About](#)[Research](#)

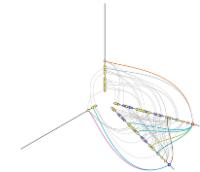
WINGS is a semantic workflow system that assists scientists with the design of computational experiments. A unique feature of WINGS is that its workflow representations incorporate semantic constraints about datasets and workflow components, and are used to create and validate workflows and to generate metadata for new data products. WINGS submits workflows to execution frameworks such as **Pegasus** and **OODT** to run workflows at large scale in distributed resources. [\[more\]](#)

komadu

Provenance Collection and Visualization Tool

Komadu is a redesign of Karma (OPM based provenance implementation) which supports the W3C PROV specification. It comes with a new Client API which aligns with the W3C PROV standards. This Client API is more generalized and supports capturing any kind of provenance.

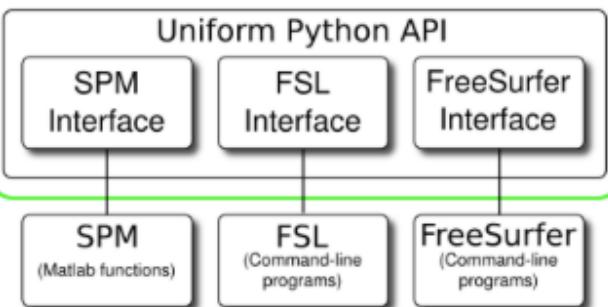
Following is a quick start guide for Komadu. More information can be found in [docs/KomaduUserGuide.pdf](#).



Nipype: Neuroimaging in Python Pipelines and Interfaces

[Home](#) · [Quickstart](#) · [Documentation](#) · [Citation](#) · [NiPy](#)

Interfaces



Idiosyncratic, Heterogeneous APIs

Workflow Engine



Current neuroimaging software offer users an incredible opportunity to analyze data using a variety of different algorithms. However, this has resulted in a heterogeneous collection of specialized applications without transparent interoperability or a uniform operating interface.

Nipype, an open-source, community-developed initiative under the umbrella of [NiPy](#), is a Python project that provides a uniform interface to existing neuroimaging software and facilitates interaction between these packages within a single workflow. Nipype provides an environment that encourages interactive exploration of algorithms from different packages (e.g., [SPM](#), [FSL](#), [FreeSurfer](#), [Camino](#), [MRtrix](#), [MNE](#), [AFNI](#), [Slicer](#)), eases the design of workflows within and between packages, and reduces the learning

Google™ Custom Search

Versions

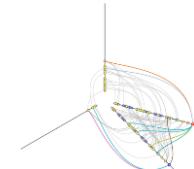
Release	Devel
0.9.2	1.0-dev
Download	Github

[Ohloh](#) [I USE IT](#) [g+1](#)

Links

Docs: [Stable](#) · [Nightly](#)
Code: [Github](#) · [Bugs-Requests](#)
Forum: [User](#) · [Developer](#)
Info: [License](#) · [Funding](#)

[build](#) [coverage](#) 70%

[DOWNLOAD](#)[DOCUMENTATION](#)[COMMUNITY](#)

Build Linked Data Applications ... directly in your Web Browser



Callimachus serves all your
Linked Data needs:

- ➊ Graph Storage
- ➋ Integrated Development Environment
- ➌ Visualizations
- ➍ Web Publishing



Oh, Yeah?

★★★★★ (4)

Productivity

You will need Google Chrome to install most apps, extensions and themes.

[Download Google Chrome](#)

from Ruben Verborgh

33 users

OVERVIEW

DETAILS

REVIEWS

RELATED

g+1

0

Oh, yeah?

Lost that feeling of trust?

Here's some information about this document.

Request date: Fri, 08 Feb 2013 14:07:44 GMT

Last modified on: Wed, 06 Feb 2013 10:49:01 GMT

Linked provenance resources

example ✓

Valid: [This is valid provenance](#) ✓

URL: <http://users.ugent.be/~tdenies/provenance/example.provn>

Target: <http://ruben.verborgh.org:1234/>

Serialization: PROV-N

example2 !

example ✓



Oh?
yeah

This button helps you regain your feeling of trust.

In 1997, Tim Berners-Lee proposed that each browser should have a button marked "Oh, yeah?" that you can press when you lose the feeling of trust when viewing a document. Upon pressing the button, information is shown about why you should (dis)trust it.

Developed by Multimedia Lab – iMinds – Ghent University.



provbench

Contributions Repositories Public activity

Popular repositories

[Wikipedia-PROV](#)

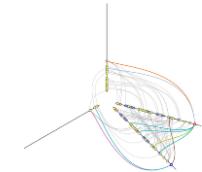
The repository for the Wikipedia-PROV team.

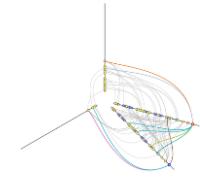
[Chiron-PROV](#)

[CSIRO-PROV](#)

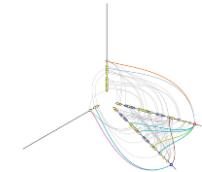
The repository for the CSIRO-PROV team.

[OBIAMA](#)



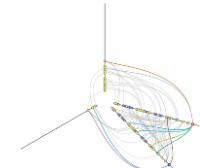


CONCLUSION



Concluding Remarks

- A data model with core and extended terms
- Valid provenance
- Serializations into various Web languages
- A set of design recipes
- Some tools emerging



Further reading

Provenance: An Introduction to PROV

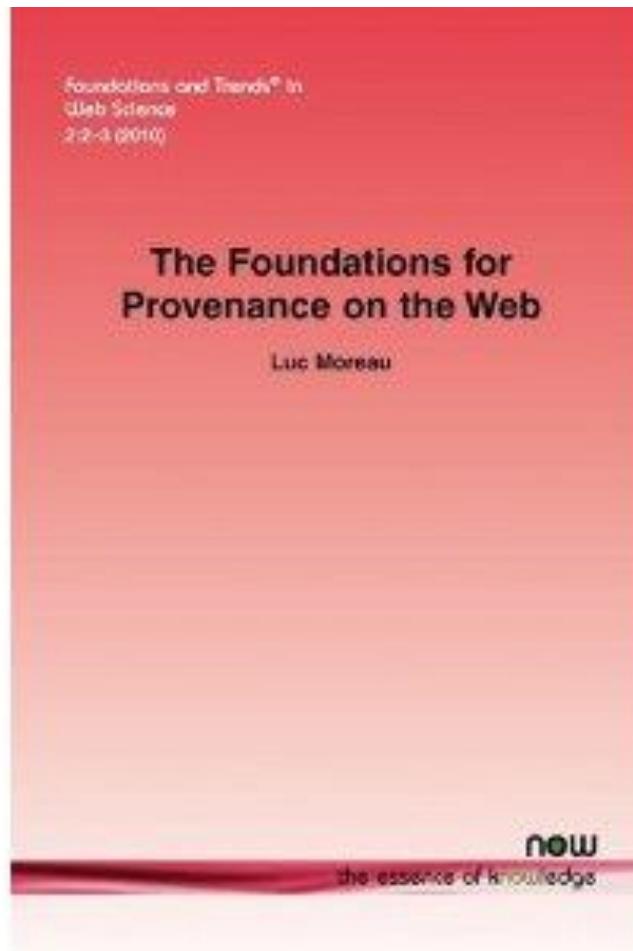
Luc Moreau
University of Southampton
Paul Groth
VU University Amsterdam

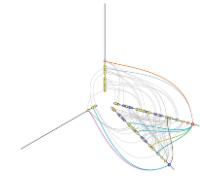
SYNTHESIS LECTURES ON SEMANTIC WEB #1



MORGAN & CLAYPOOL PUBLISHERS

www.provbook.org

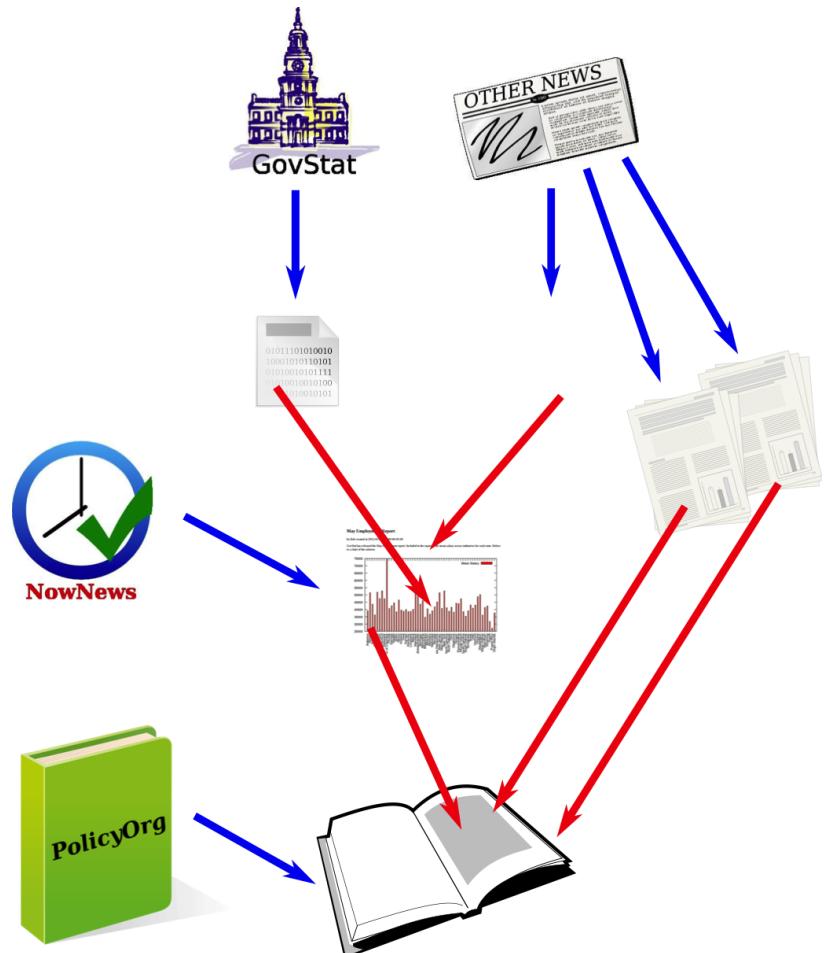


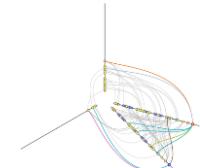


VALIDATION

Use Case (1): Version Problem

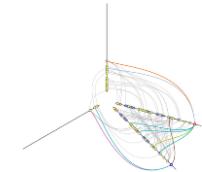
- NowNews provenance indicates that an article includes a quote from another document.
- That document happens to be the compilation produced by PolicyOrg. The compilation itself includes the NowNews article.
- A **circularity** occurs in the provenance, which is indicative of a **problematic** description.





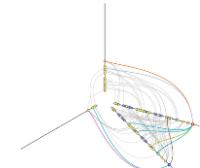
Use Case (2): Date Issue

- A plot was computed from a data set. The plot timestamp is found to precede the data set timestamp.
- Given that the plot was derived from the data set, this is an indicator of a problem.



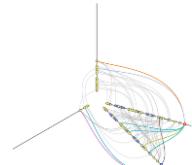
Principles of Validation

- The intent of validation is to ensure that PROV descriptions represent a consistent history of objects and their interactions.
- Once established valid, PROV descriptions are safe to use for the purpose of logical reasoning and other kinds of analysis.



PROV Events

- Time is critical for provenance, since it can help corroborate provenance descriptions.
- PROV makes no assumption on the clocks being used when asserting time: a unique clock is not expected, nor clocks are expected to be synchronized.
- Instead, PROV relies on an event model describing changes in the world.



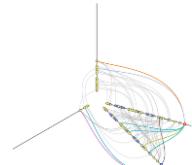
Five Types of Events

Events

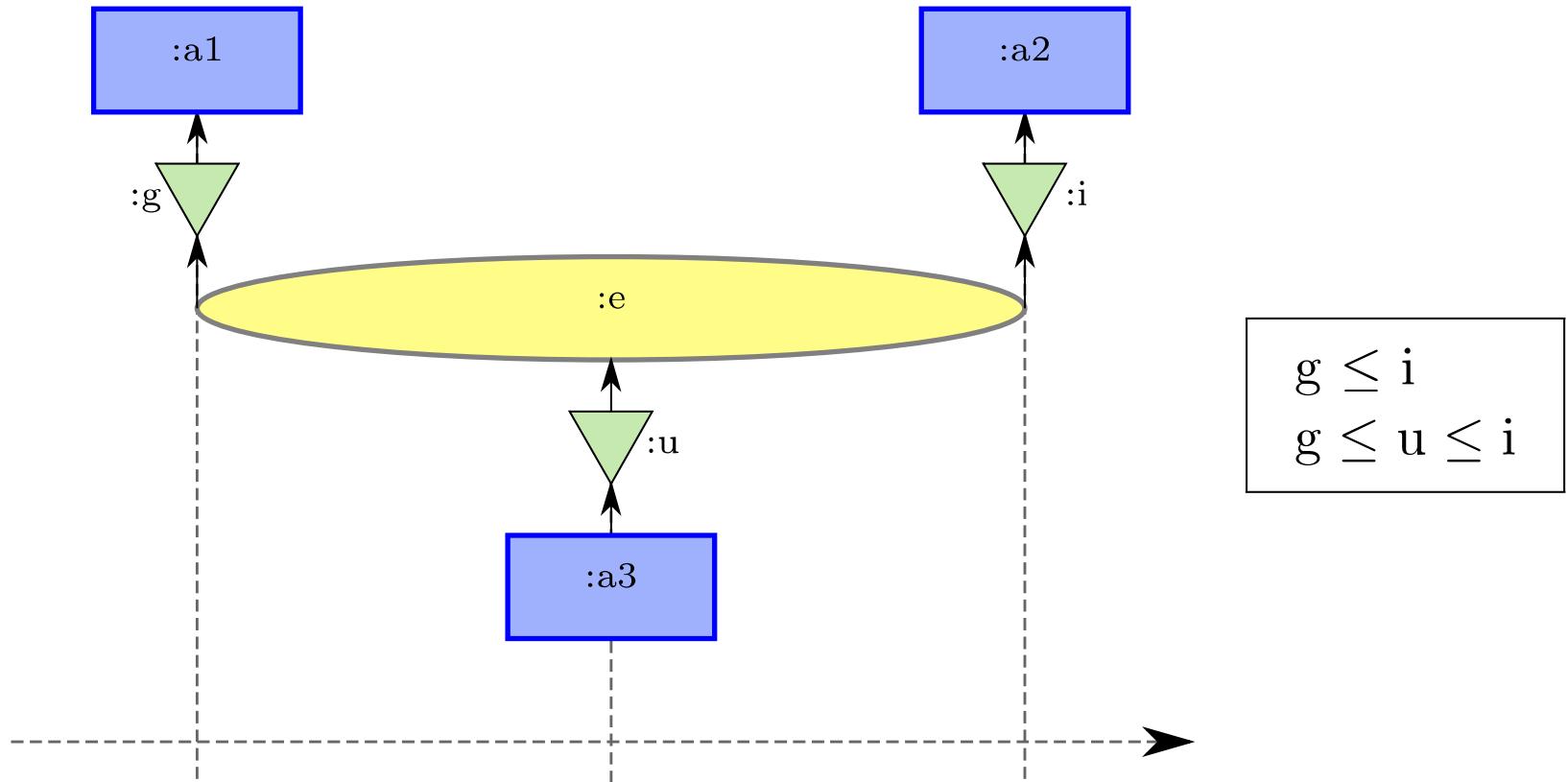
- generation
- invalidation
- usage of entities
- start
- end of activities

Order

- event1 (strictly) precedes event2

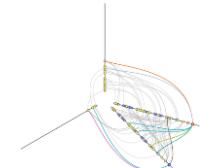


Entities Have a Lifetime

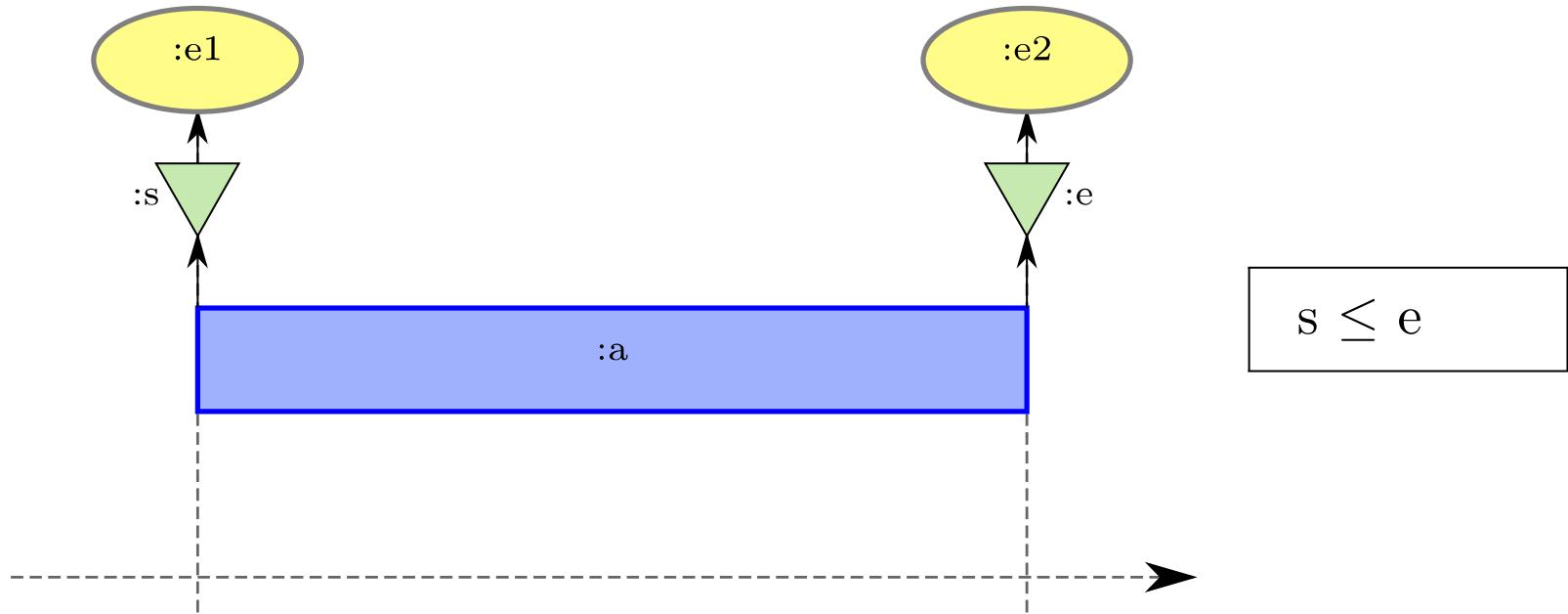


Generation precedes invalidation.

Usage follows generation and precedes invalidation.

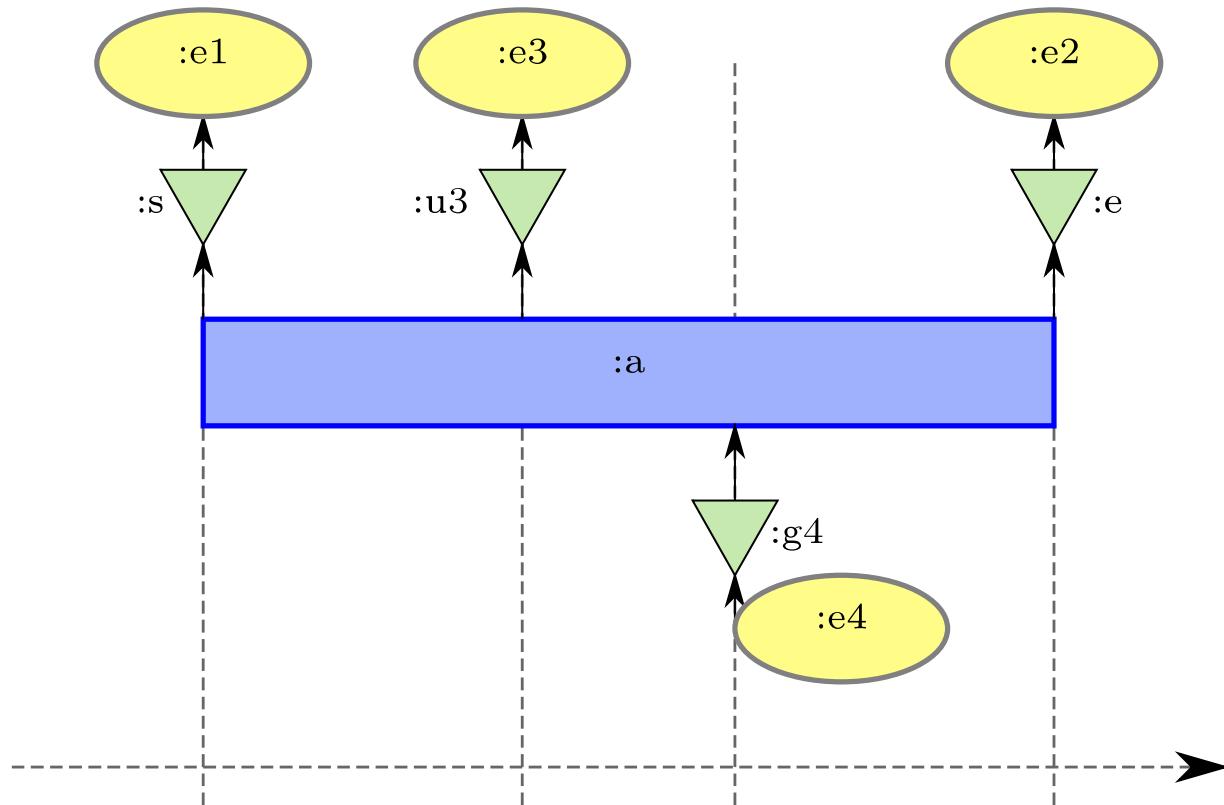


Activities Have a Lifetime



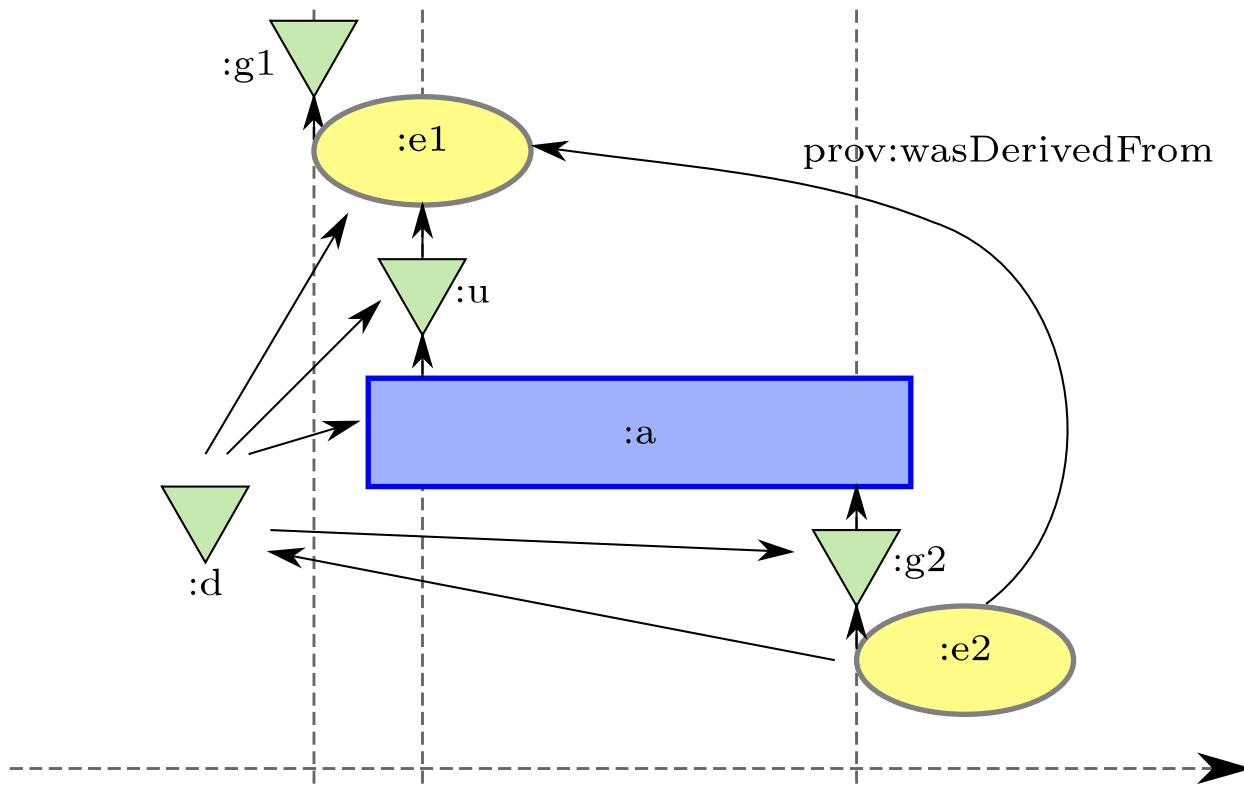
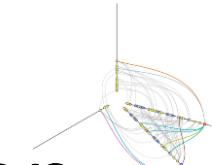
Start precedes end.

Usage within Activity Lifetime


$$\begin{aligned} s &\leq u3 \\ u3 &\leq e \end{aligned}$$
$$\begin{aligned} s &\leq g4 \\ g4 &\leq e \end{aligned}$$

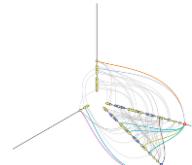
Usage follows start and precedes end.

Constraints associated with Derivation

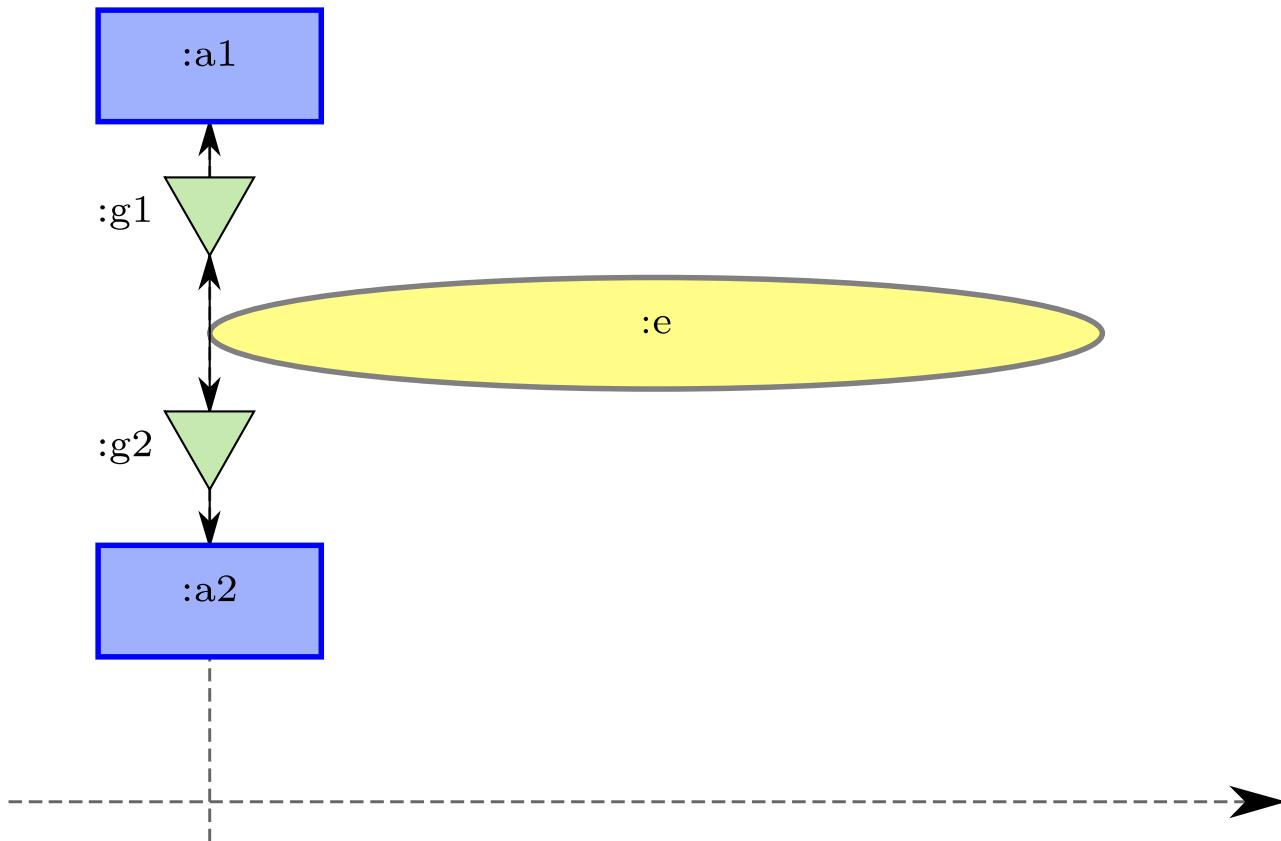


$$\begin{aligned} g1 &< g2 \\ u &\leq g2 \\ g1 &\leq u \end{aligned}$$

Generation of used entity strictly precedes the generation of the derived entity.



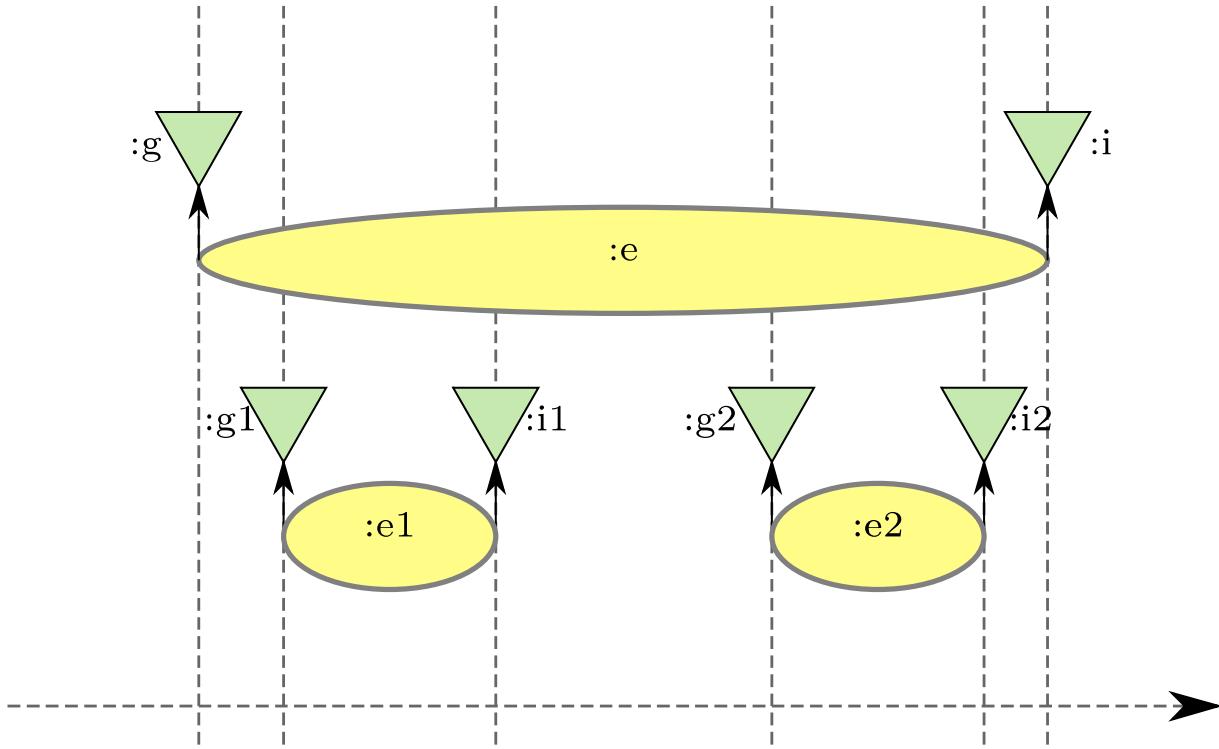
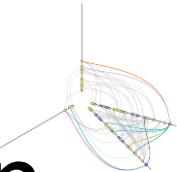
Simultaneous Events



$$\begin{aligned} g1 &\leq g2 \\ g2 &\leq g1 \end{aligned}$$

Two generation events for an entity occur simultaneously.

Nested Intervals and Specialization



$$\begin{aligned} g &\leq g_1 \\ i_1 &\leq i \end{aligned}$$

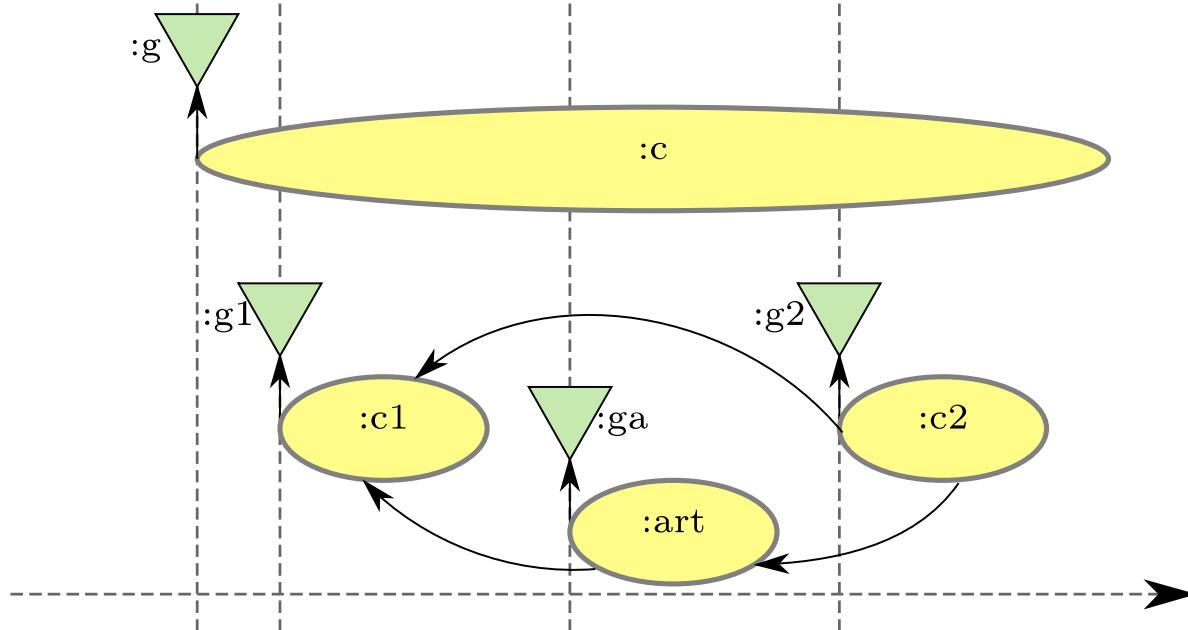
$$\begin{aligned} g &\leq g_2 \\ i_2 &\leq i \end{aligned}$$

$$\begin{aligned} g &\leq i \\ g_1 &\leq i_1 \\ g_2 &\leq i_2 \end{aligned}$$

The lifetime of a specialized entity is included in the lifetime of the general entity.

Back to the Example

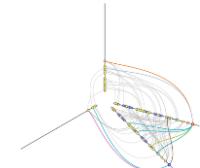
NowNews provenance indicates that an article includes a quote from the compilation produced by PolicyOrg, which includes the NowNews article.



invalid
 $g < ga$
 $ga < g$

valid
 $g1 < ga$
 $ga < g2$
 $g \leq g1$
 $g \leq g2$

Event Chart for Use Case: article :art include quote from :c1 and was itself included in :c2. The generation :g1 of :c1 strictly precedes the generation :ga of :art, which strictly precedes the generation :g2 of :c2



Checking Time Annotations

A plot was computed from a data set. The plot timestamp is found to precede the data set timestamp.

