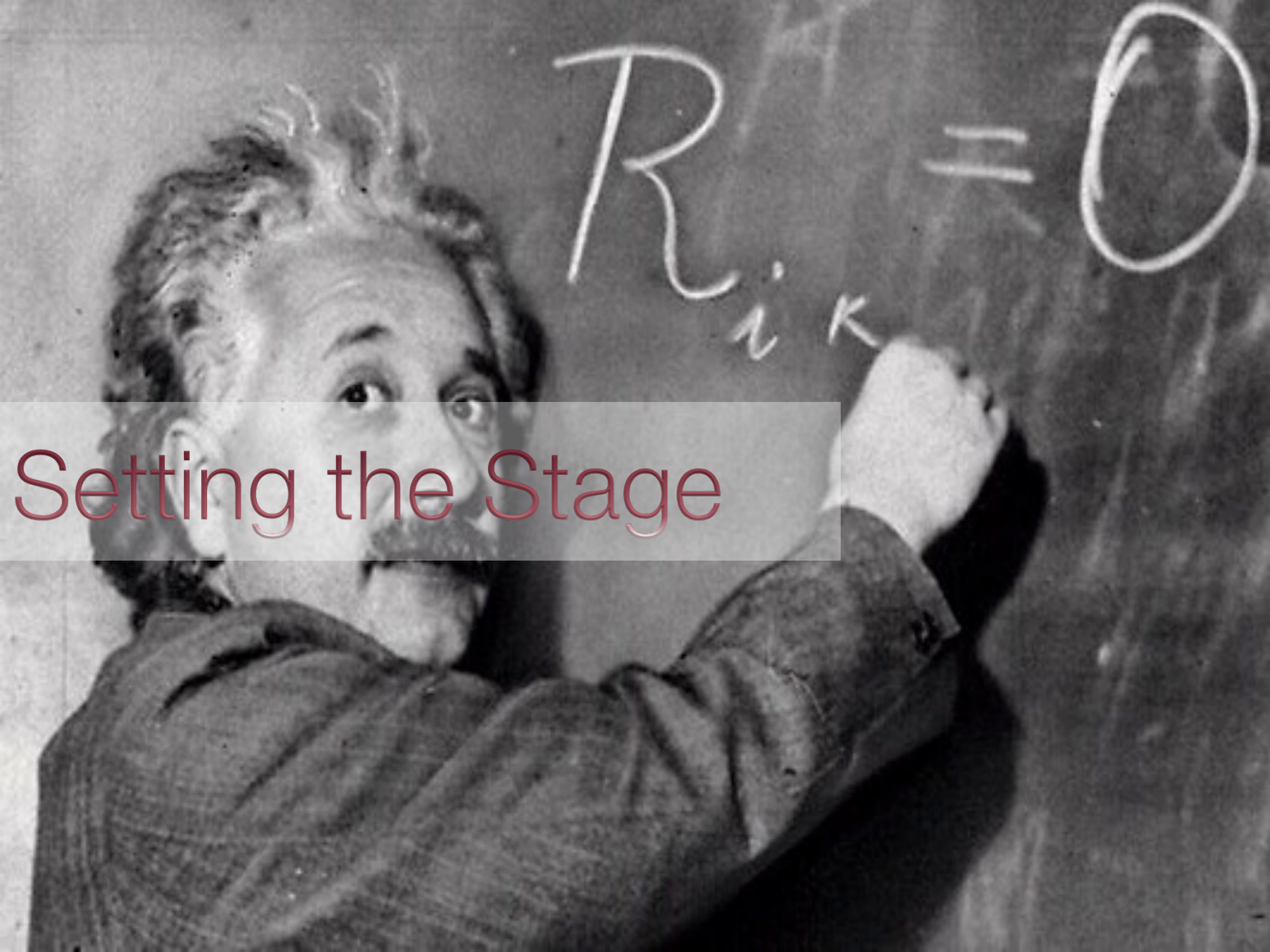
The W3C Prov Ontology

Cambridge Semantic Web Gathering
2012-10-09, Cambridge, MA, USA

Ivan Herman
W3C, Semantic Web Activity Lead



Setting the Stage

The goal is simple…

- We should be able to express special “meta” information on the data
  - who played what role in creating the data (author, reviewer, etc.)
  - view of the full revision chain of the data
  - in case of integrated data which part comes from which original data and under what process
  - what vocabularies/ontologies/rules were used to generate some portions of the data
  - etc.

…the solution is more complicated

- Requires a complete model describing the various constituents (actors, revisions, etc.)
- The model should be usable with RDF to be used on the Semantic Web
- Has to find a balance between
  - simple (“scruffy”) provenance: easily usable and editable
  - complex (“complete”) provenance: allows for a detailed reporting of origins, versions, etc.

Lots of application areas need provenance

- Open Information Systems
  - origin of the data, who was responsible for its creation
- Science applications
  - how the results were obtained
- News
  - origins and references of blogs, news items
- Law
  - licensing attribution of documents, data
  - privacy information
- Etc.

“Provenance” is not a new subject

- There has been lot of work around
  - workflow systems
  - databases
  - knowledge representation
  - information retrieval
- There are communities and vocabularies out there
  - Open Provenance Model (OPM)
  - Dublin Core
  - Provenir ontology
  - Provenance vocabulary
  - SWAN provenance ontology
  - etc.

W3C's Provenance Incubator Group

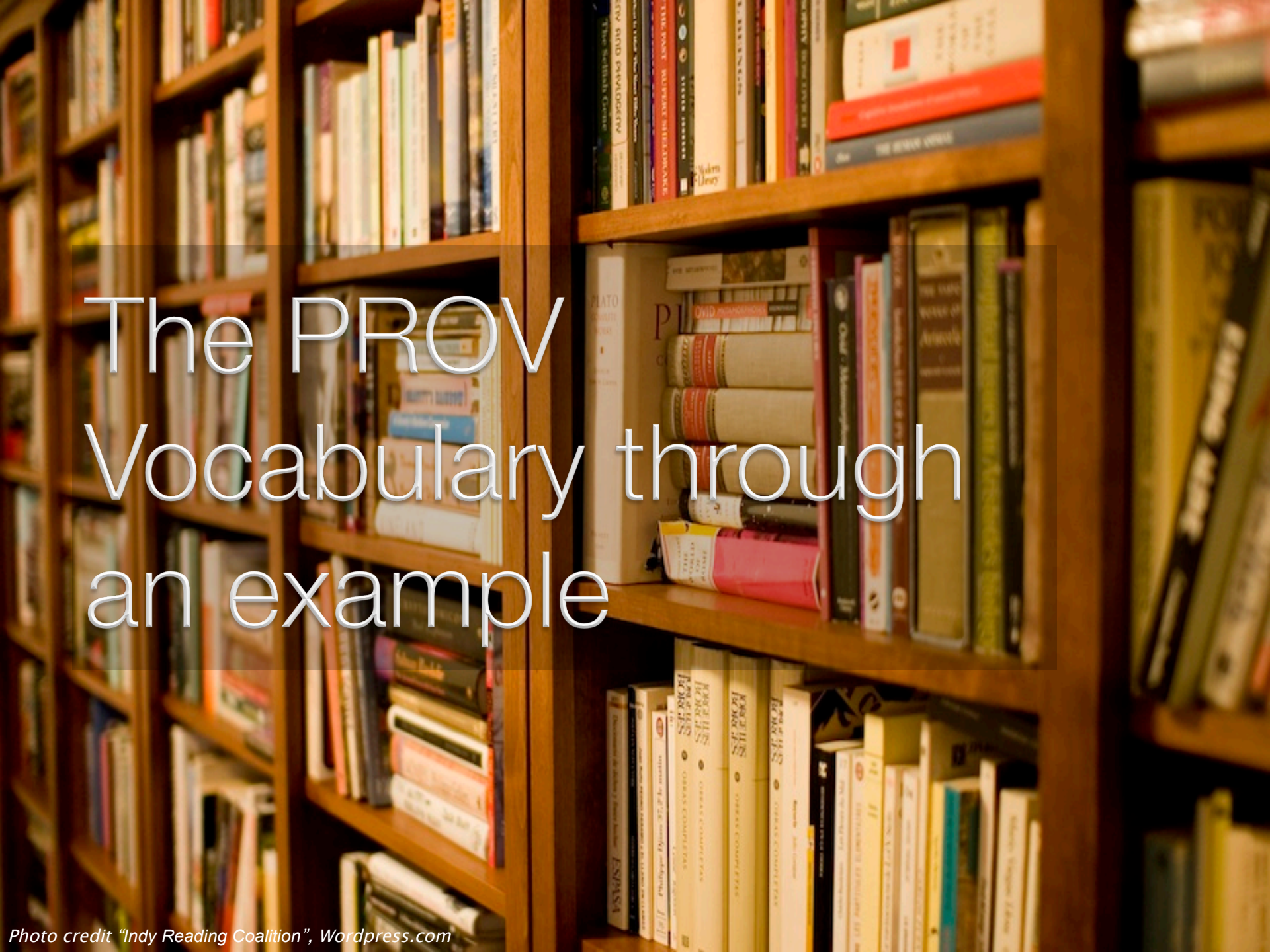
- Worked in 2009-2010 (Chaired by Yolanda Gil)
- Issued a final report
  - "Provenance XG Final Report"
    - http://www.w3.org/2005/Incubator/prov/XGR-prov/
  - provides an overview of the various existing approaches, vocabularies
  - proposes the creation of a dedicated W3C Working Group

Definition of Provenance (by the Provenance XG)

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

W3C Provenance Working Group

- Set up in April 2011 (co-chaired by Paul Groth and Luc Moreau)
- Goal is to define a standard vocabulary for provenance, primarily for the Semantic Web
- This is what I will talk about in what follows…

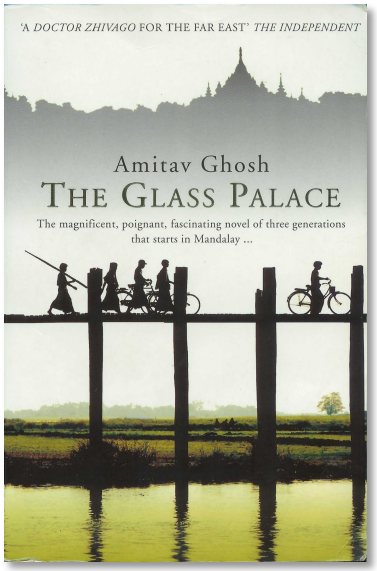


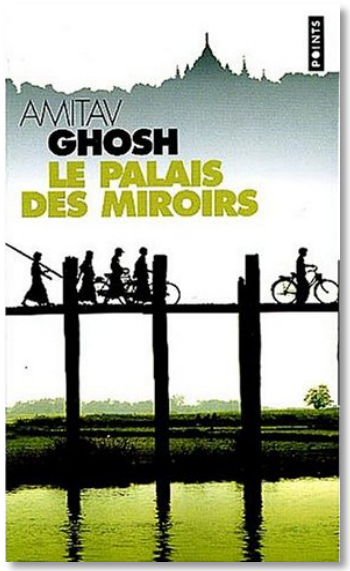
The PROV Vocabulary through an example

Photo credit "Indy Reading Coalition", Wordpress.com

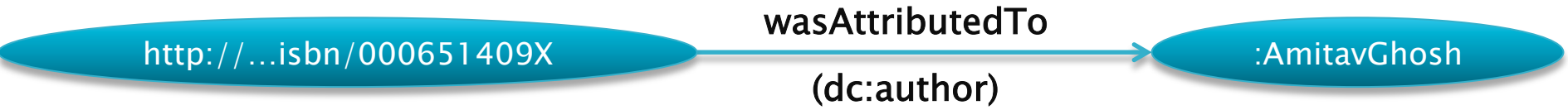
The example

- We have data on two books
  - “The Glass Palace”, written by Amitav Ghosh
  - “Le palais des mirroirs”, 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

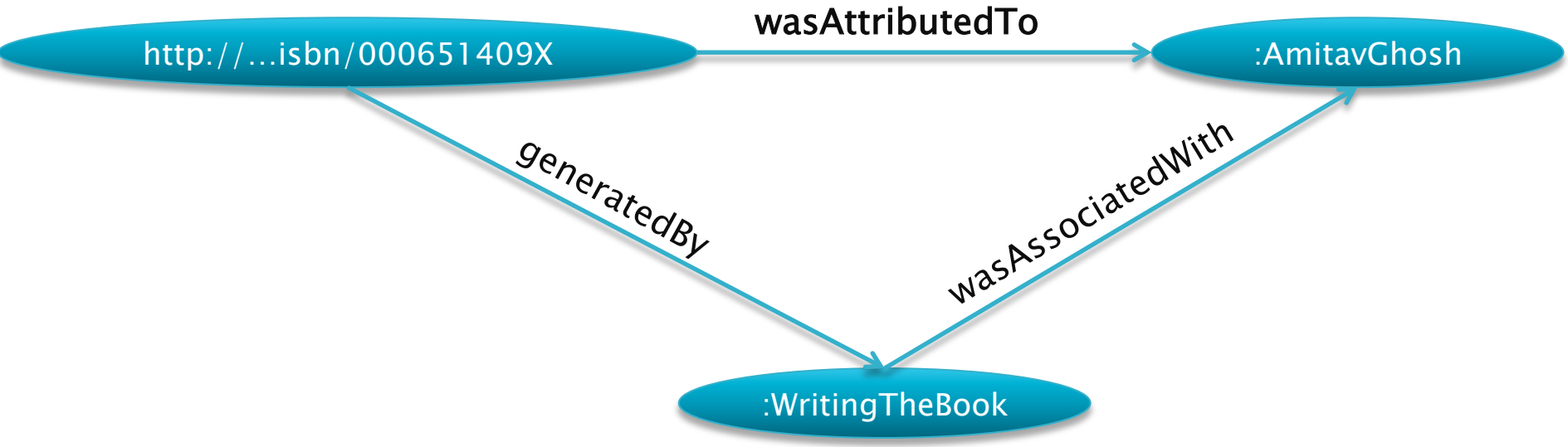




A very simple attribution



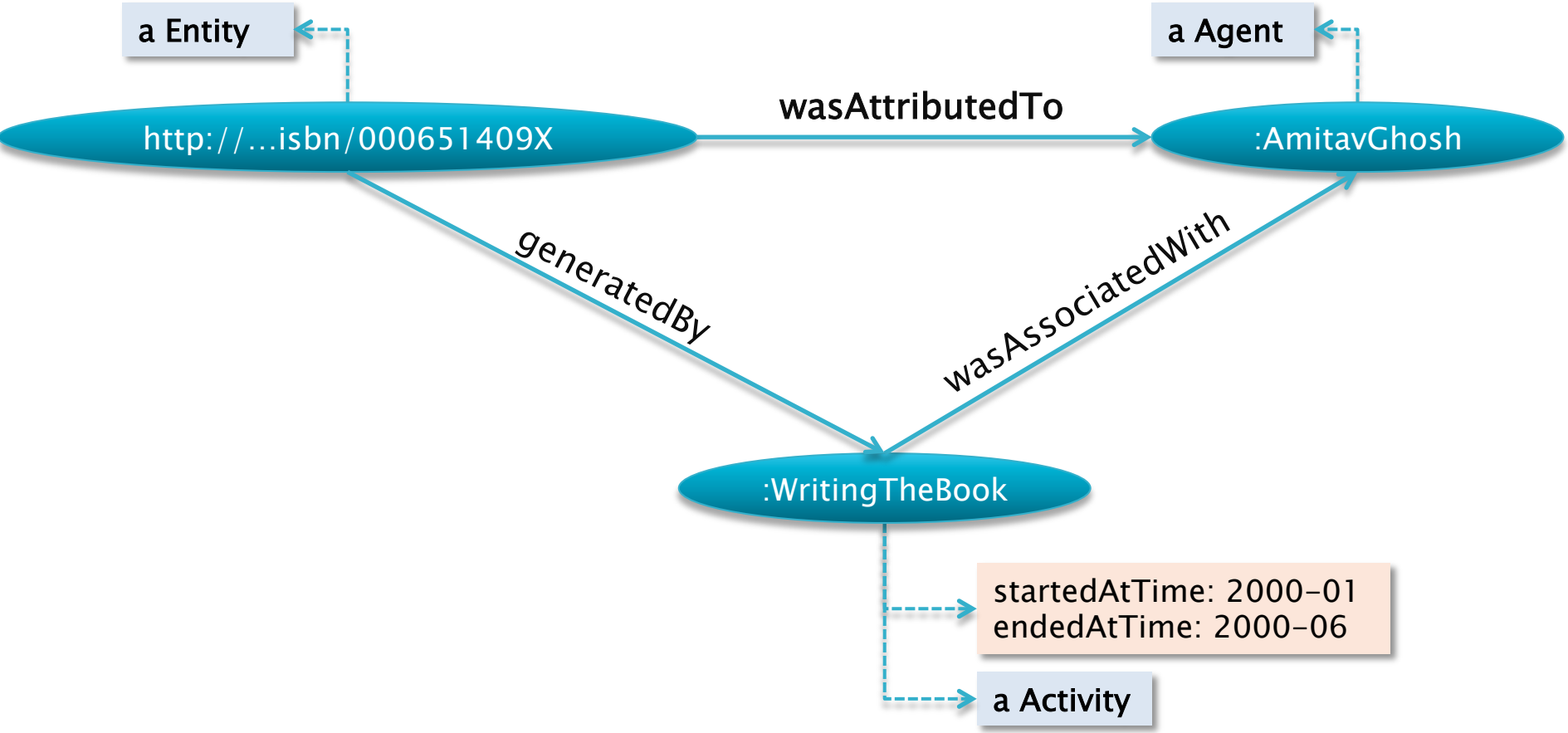
A bit more complicated: make the activity explicit



Why doing this?

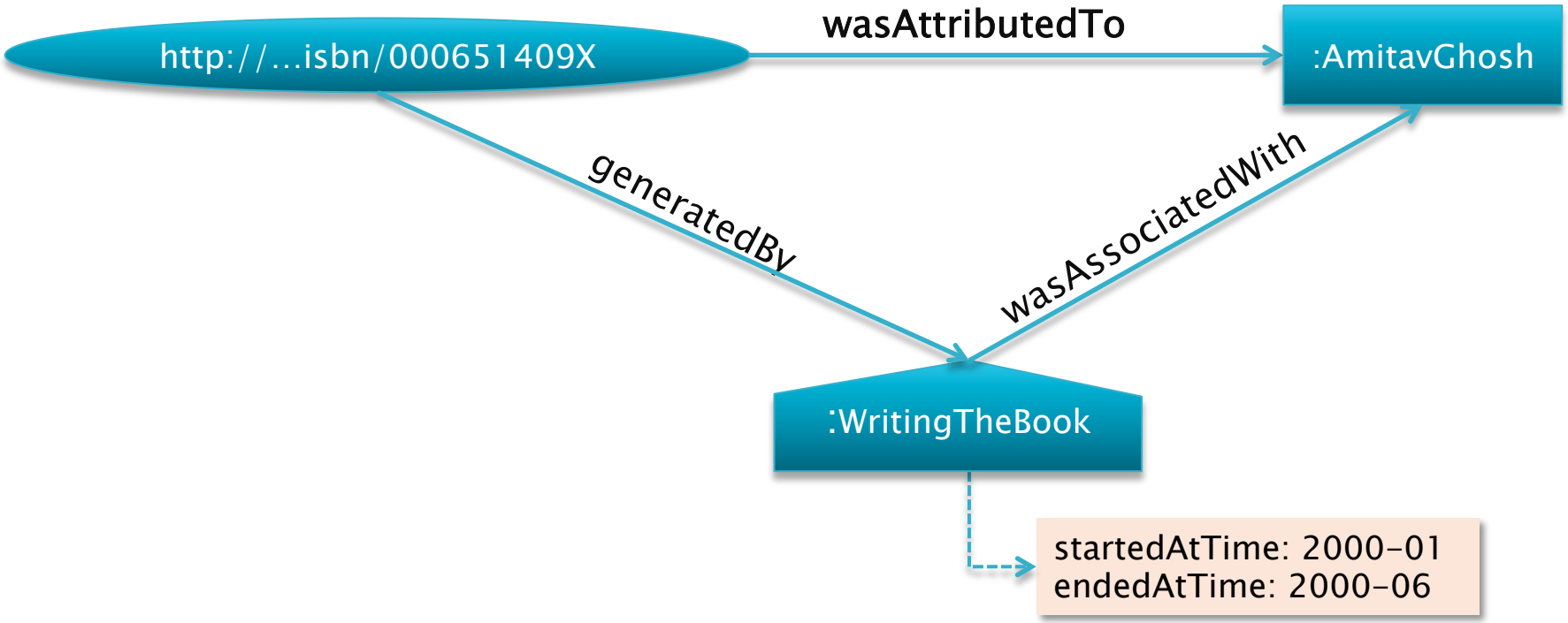
To make some “metadata” explicit

A more complete attribution: make the activity explicit



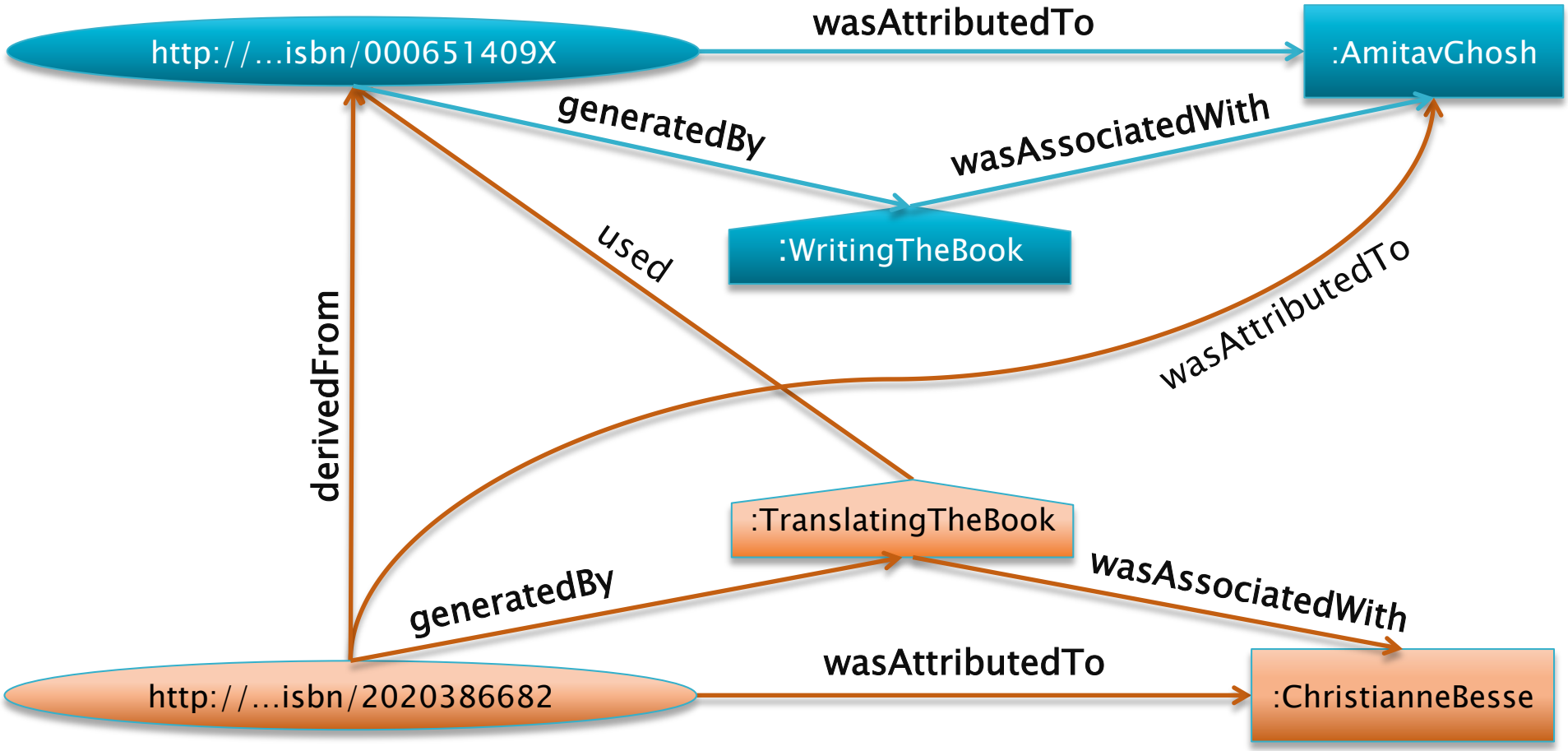
The fundamental notions of the PROV Vocabulary

- This simple example shows the fundamental notions
  - Entity:
    - the “things” whose provenance we want to describe
  - Action:
    - describes how entities are created, changed. The “dynamic” aspect of the world
  - Agent:
    - are responsible for the actions.
  - Usage and generation terms
    - connections describing how entities, agents, and actions interact


http://...isbn/000651409X
wasAttributedTo
:AmitavGhosh
generatedBy
wasAssociatedWith
:WritingTheBook
startedAtTime: 2000-01
endedAtTime: 2000-06

Let us make it a bit more complex

Adding the translation

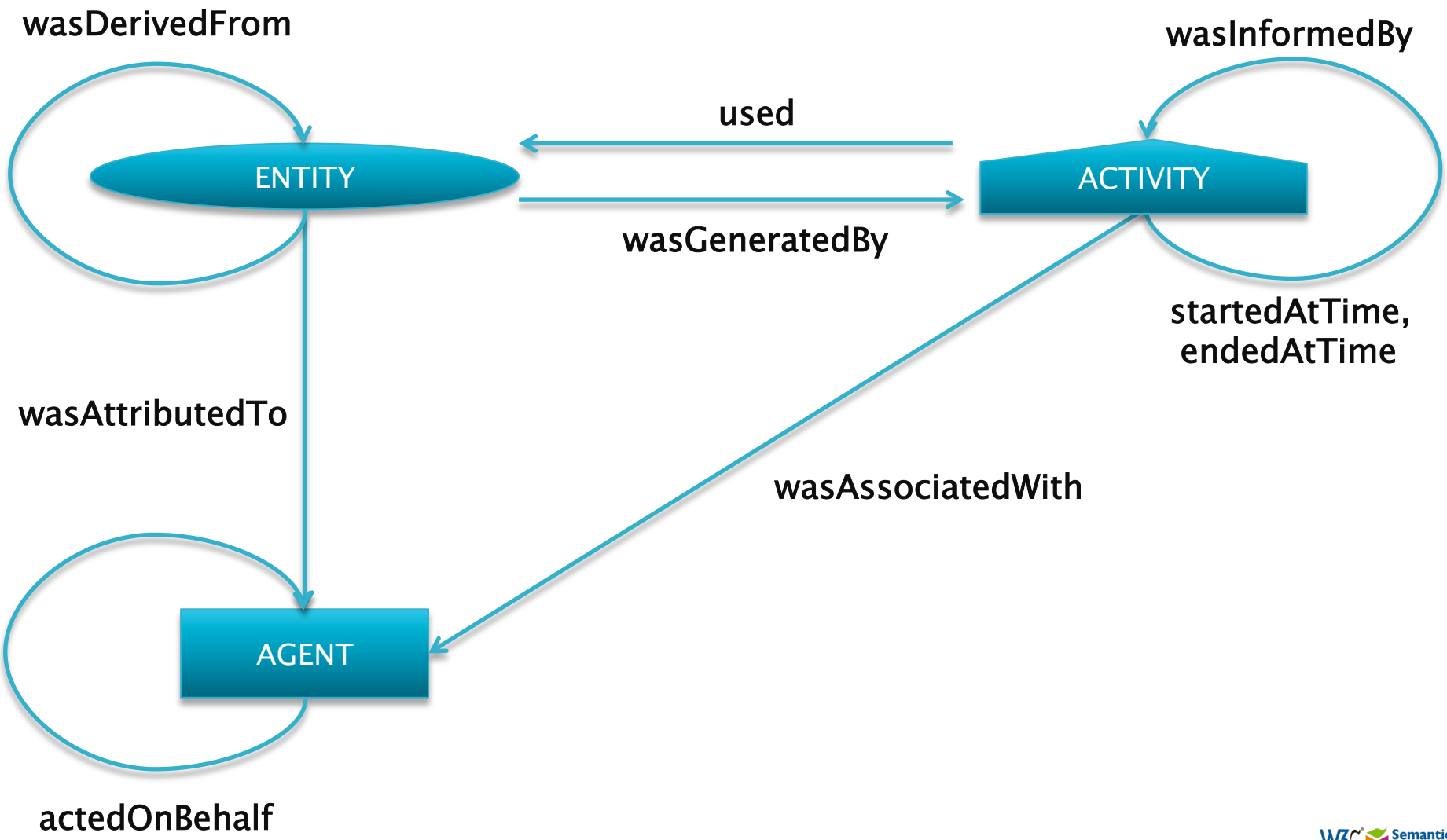


Categories of PROV Terms

Categories of PROV Terms

- *Starting Point classes and properties*: the basics
- *Expanded classes and properties*: additional terms around the starting point terms for richer descriptions
- *Qualified classes and properties*: for provenance geeks ☺

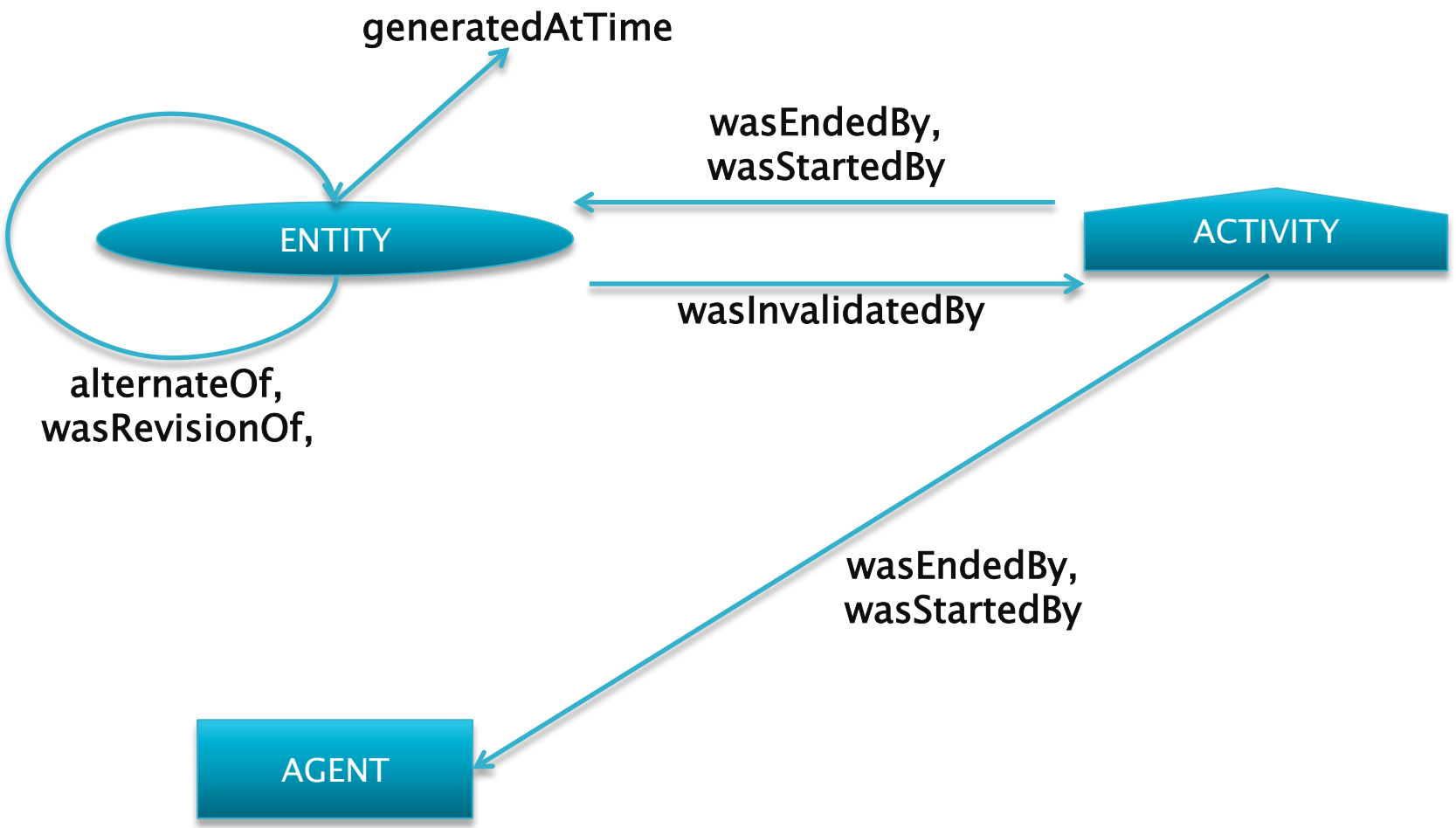
Starting point classes and properties



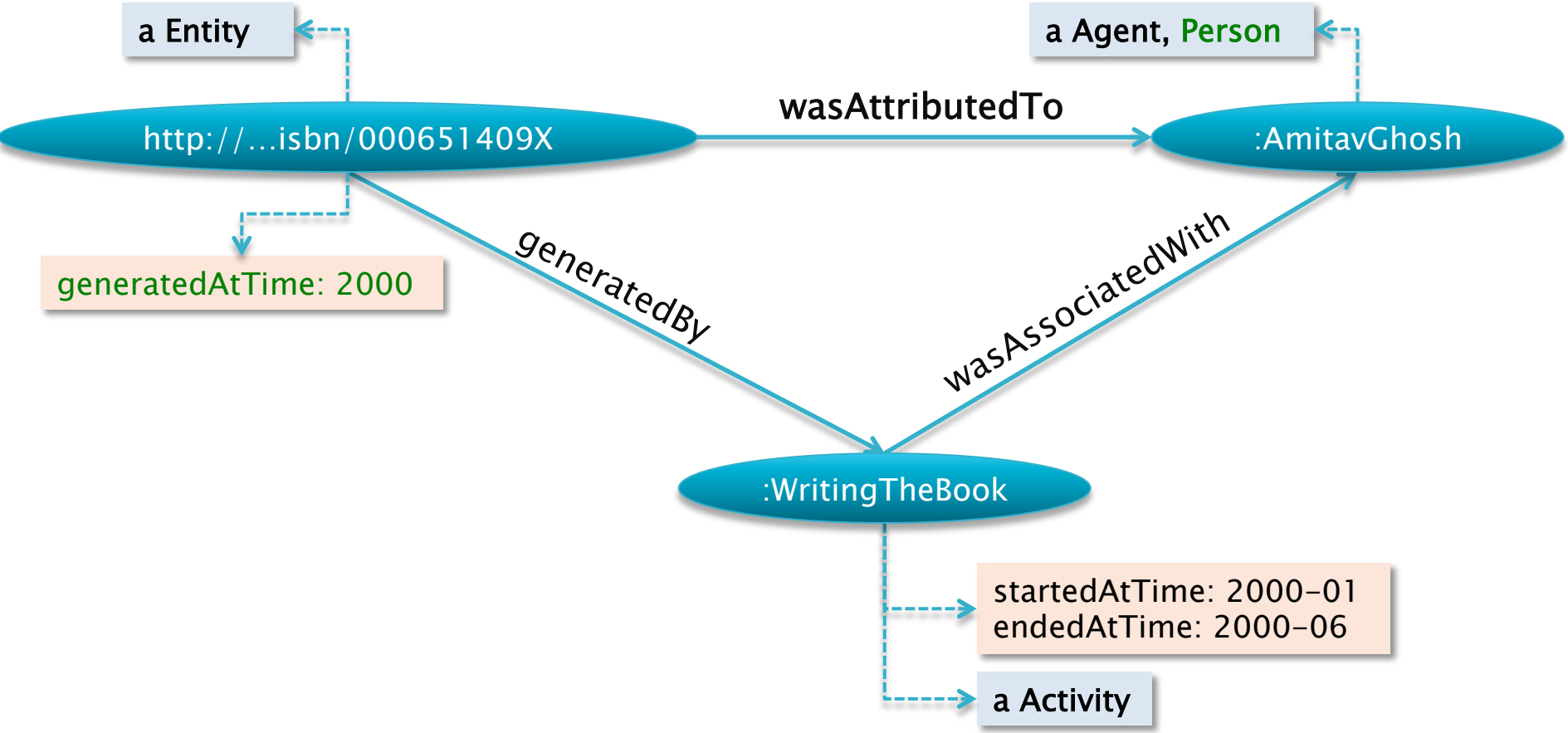
Expanded classes and properties

- Some extra classes, defined as subclasses of agents:
  - Organization, Person, SoftwareAgent
- Some extra properties describing versioning, influencing, invalidation, or creation of entities, etc.
- Nothing structurally different, just adding some specialization
  - applications are of course welcome to add their own specializations

Some examples for extra properties

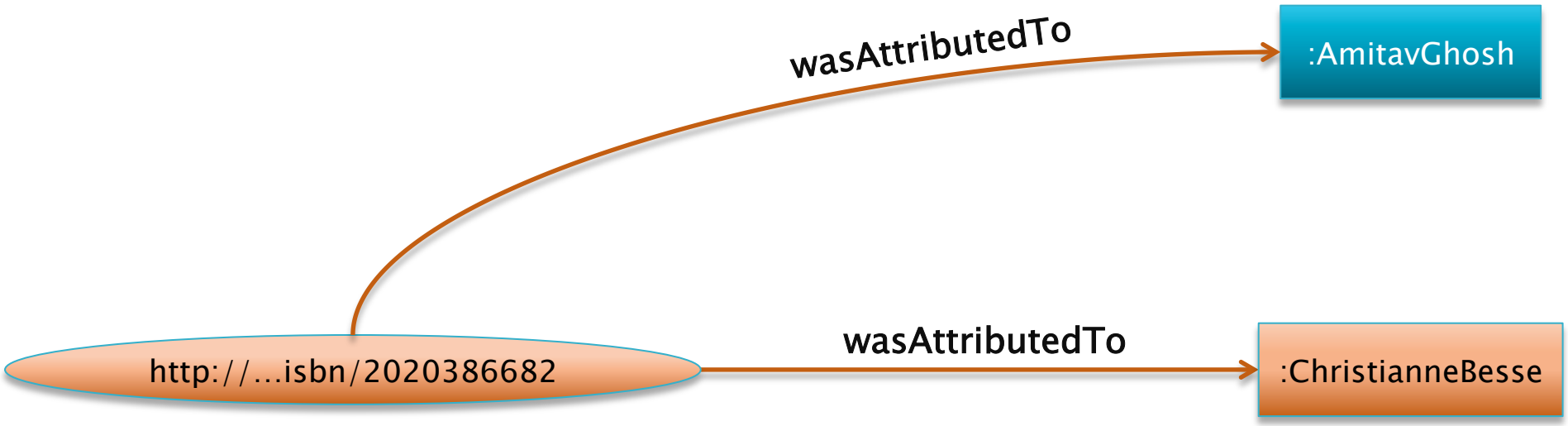


Adding some extra properties



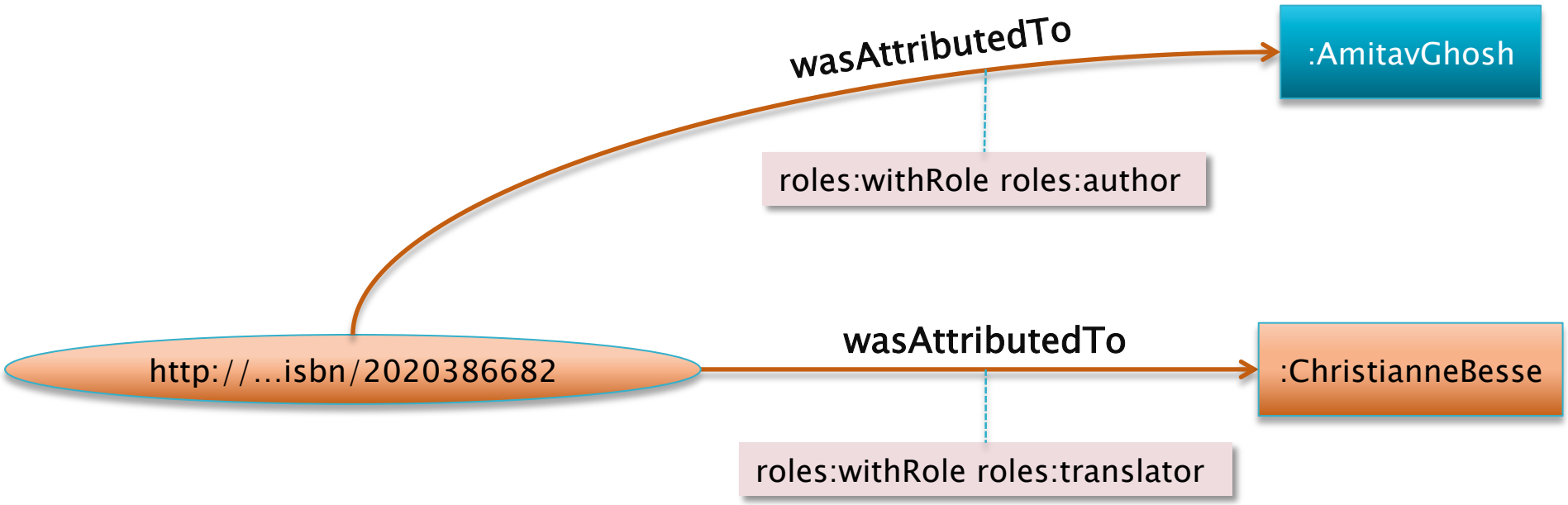
Qualified relationships

Remember this?

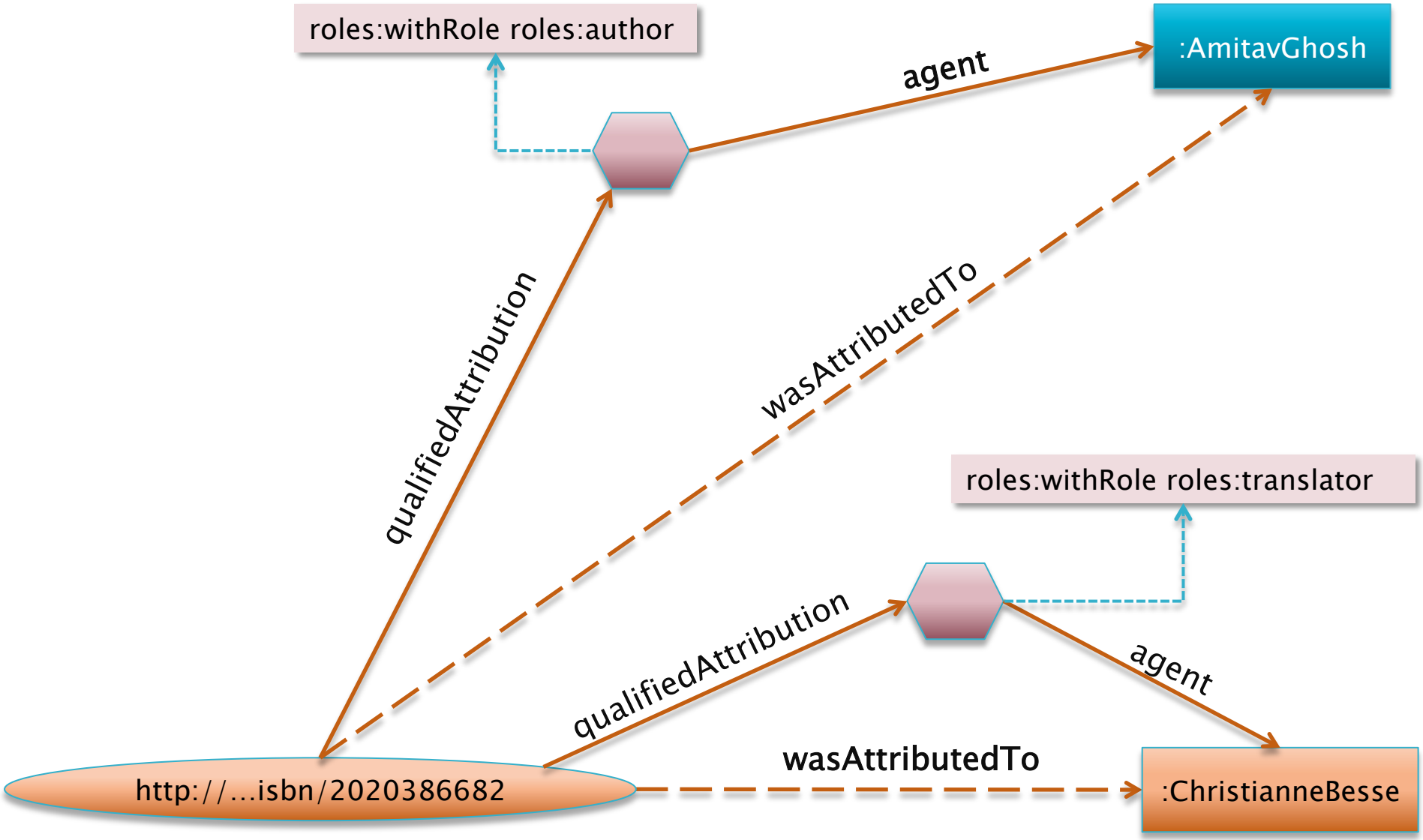


Something is missing here…

- Clearly, Amitav Ghosh played a different role than Christianne Besse
- We want to “qualify” the prov:wasAttributedTo relationships



The chosen approach: define qualification structures



Qualification structures

- Most of the starting or extended properties have their “qualified” counterpart
  - qualifiedAttribution, qualifiedUsage, etc.
- Application may add additional properties to these structures to refine them further

The chosen approach: define qualification structures

```
@prefix prov:   <http://www.w3.org/ns/prov#> .
@prefix roles:  <http://purl.org/spar/pro/>  .

<http://.../isbn/2020386682> a prov:Entity ;
        prov:wasGeneratedBy :TranslatingTheBook ;
        prov:wasAttributedTo :AmitavGhosh, :ChristianneBesse ;
        prov:qualifiedAttribution [
                a prov:Attribution ;
                prov:agent     :AmitavGhosh ;
                roles:withRole roles:author ;
        ]
        prov:qualifiedAttribution [
                a prov:Attribution ;
                prov:agent     :ChristianneBesse ;
                roles:withRole roles:translator ;
        ]
        …
```

courtesy to "analogue kid"

Dublin Core

- Obviously, there are lots of overlap
  - some terms have direct equivalents
  - some need a slightly more complex relationship
- The Working Group will publish a separate note

Some simple Dublin Core relationship examples

- dc:Agent = prov:Agent
- dc:creator ⊑ prov:wasAttributedTo
- dc:isVersionOf ⊑ prov:wasDerivedFrom
- prov:wasRevisionOf ⊑ dc:isVersionOf
- etc.

These relationships will be published in a separate RDFS document

Some cases are more complicated

- For example, Dublin Core's "creator" has more to it than simply an agent. The correspondence is something like:
  - "If an entity is attributed to an agent, and the agent's role matches Dublin Core's definition of a creator, then the agent is the creator of the entity in the Dublin Core sense"
- These (few) cases are described in terms SPARQL CONSTRUCT rules


Constraint checking of provenance statements

Checking provenance statements ("Constraints")

- Provenance statements can become fairly complicated ☹
- In some applications it may become advantageous to *check* the validity of the provenance structures. E.g,
  - typing constraints on the relationships should be upheld
  - if an entity is invalidated by several activities, these events must happen simultaneously
  - the time assigned to the creation of the entity, and the times set on the related activity should be compatible
  - etc.

Definition of the constraints

- An abstract data model for provenance (with its own, abstract notation) is also published

```
entity(<http://.../isbn/000651409X>)
activity(:WritingTheBook)
wasGeneratedBy(<http://.../isbn/000651409X>,:WritingTheBook)
agent(:AmitavGhosh,
          [prov:type='prov:Person',foaf:name='AmitavGhosh'])
wasAttributedTo(<http://.../isbn/000651409X>,:AmitavGhosh,
                    [roles:witRole='roles:author'])
```

Note that the "qualified" versions are unnecessary at that level, relationships are n-ary

Definition of the constraints

- A separate document defines the constraints on the abstract data model
- Constraints themselves are defined as a set of abstract rules
  - they may translated into:
    - (partially) into OWL
    - rules, e.g., using SPARQL
  - general constraint checkers on the abstract model are also doable

Available documents

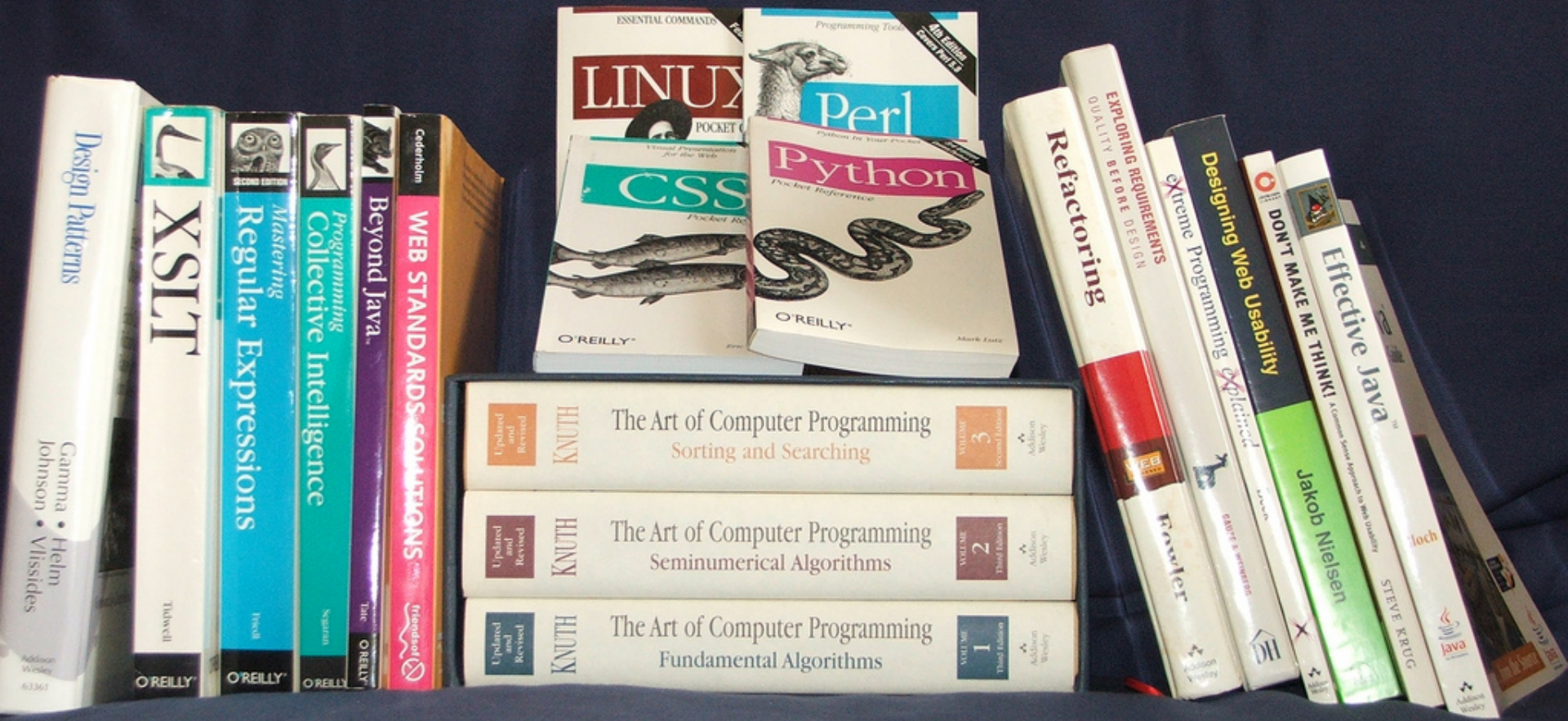


Photo credit "Abizern", Flickr

Documents published by the Group

- Major documents are:
  - PROV Primer (http://www.w3.org/TR/prov-primer/)
  - PROV Ontology(*) (http://www.w3.org/TR/prov-o/)
  - PROV Data Model(*) (http://www.w3.org/TR/prov-dm/)
  - PROV Notation(*) (http://www.w3.org/TR/prov-n/)
  - PROV Constraints(*) (http://www.w3.org/TR/prov-constraints/)
  - PROV Access and Query (http://www.w3.org/TR/prov-aq/)
- Some other notes are also in preparation:
  - PROV XML Serialization
  - PROV DC Mapping

(*)Rec-track documents

Working Group Status

- The Rec Track documents are almost in CR
- Plan is to finish the work in March 2013

An interesting extension

- “Provenance Vocabulary”
  - http://purl.org/net/provenance/ns
  - specialized for provenance of data on the Web
    - subclasses for Agents, Entities, Activities
    - subproperties for PROV properties

Thank you for your attention

These slides are also available on the Web:

http://www.w3.org/2012/Talks/1009-MIT-IH/