W3C PROV Constraints

ISWC 2013

Paul Groth

slide help from Ivan Herman
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.
Definition of the constraints

- An abstract data model for provenance (with its own, abstract notation) is also published
- http://www.w3.org/TR/prov-n/

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:wtRole='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
  – http://www.w3.org/TR/prov-constraints/
• 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
Normalization

From an RDF perspective
1. Expand
2. Everything is already merged by virtue of URIs
   • Blank nodes....

![Diagram showing relationships between Entity, Activity, Usage, prov:used, prov:entity, prov:qualifiedUsage, prov:atTime, xsd:dateTime]
Events on a Lifetime

generation

invalidation

:e1

start

:end

:a1
Uniqueness Constraints

select ?e where {
  FILTER (?gen1 != ?gen2)
}
(a) wasStartedBy-ordering
(b) wasEndedBy-ordering
Create the event timeline

```
CREATE

INSERT {
}
WHERE {
  ?act a prov:Activity .
}
```
Check for cycles

select ?x where {
  ?x (c:precedes+ | c:strictlyPrecedes+) /
    c:strictlyPrecedes ?x .
}
Impossibility Constraint

e.g. Activity & Entity disjoint

select ?e where {
  ?e a prov:Entity, prov:Activity .
}

Do you fill all the slots?

```sparql
select ?asc where {
  FILTER NOT EXISTS {
  }
}
```
Validators

• Prov-check
  – https://github.com/pgroth/prov-check

• Southampton Provenance Suite
  – https://provenance.ecs.soton.ac.uk