RUL: A Declarative Update Language for RDF

Matoula Magiridou[1], Stavros Sahtouris[2], Vassilis Christophides[2], Manolis Koubarakis[1]

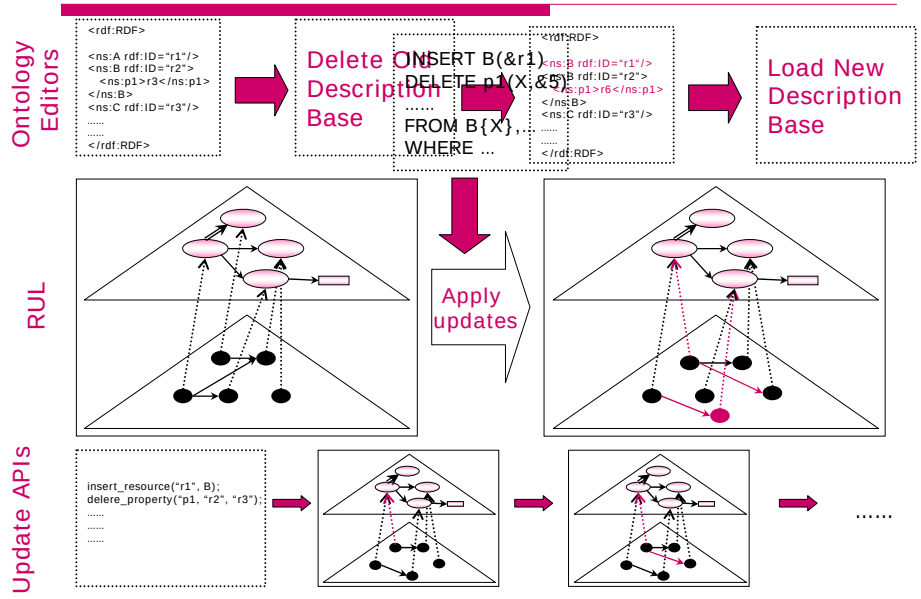
[1] Dept. of Electronic and Computer Engineering, Technical University of Crete

[2] Institute of Computer Science, FORTH

Outline

- Motivation
- RUL Design Choices
- A Tour of RUL Through Examples
- Semantics (See Paper)
- Previous Work
- Conclusions – Future Work

Motivation



RUL Design Choices

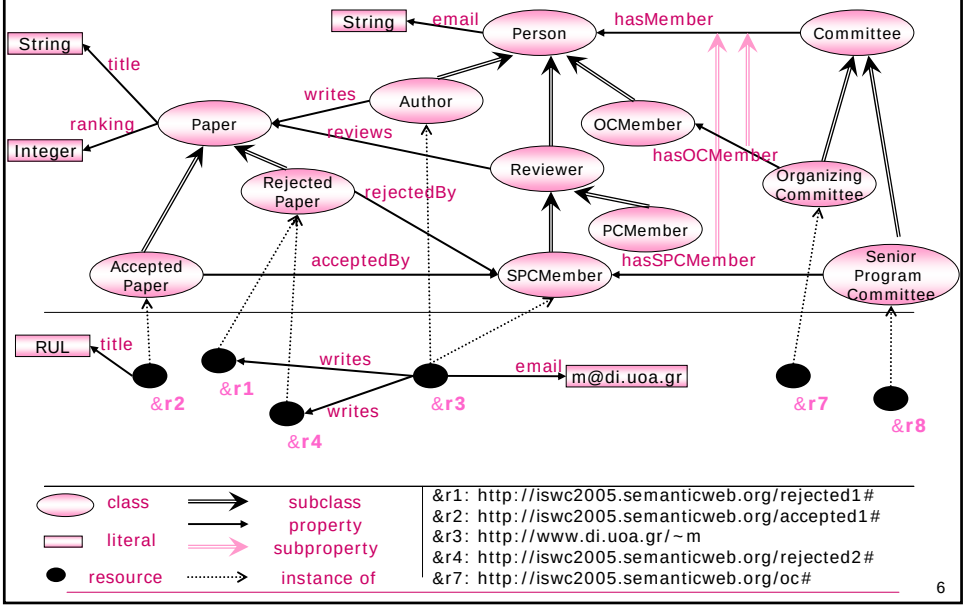
- Granularity:
  - offers primitive updates at the class instance and property instance level and bulk updates through complex language constructs
- Type safety:
  - relies on RQL typed data model
- Smooth integration with underlying RDF/S query language:
  - benefit of the full expressive power of RQL for powerful pattern matching
- Non-determinism?
  - deterministic behavior of update statements

A Tour of RUL Through Examples

- Motivating Application
- Insertions
  - Insert Class Instances
  - Insert Property Instances
- Deletions
  - Delete Class Instances
  - Delete Property Instances
- Modifications
  - Modify Class Instances
  - Modify Property Instances
- Set-Oriented Updates
- More Expressive Updates

5

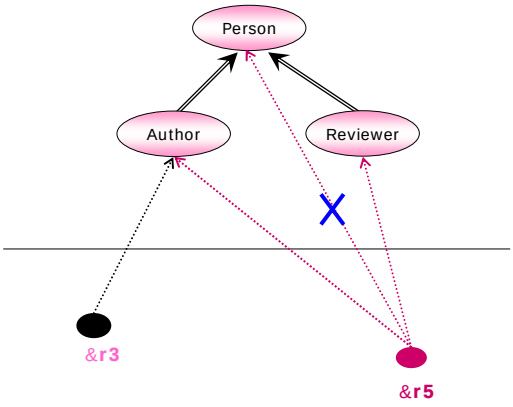
Motivating Example



6

Insert Class Instances

INSERT Person(&r5)
INSERT Reviewer(&r5)
INSERT Person(&r5)
INSERT Author(&r5)

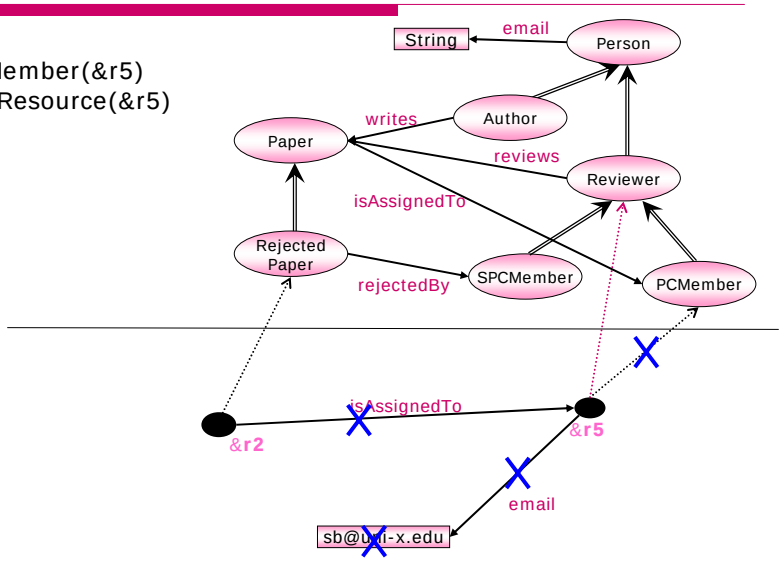


&r5: http://www.ics.forth.gr/~c

7

Delete Class Instances

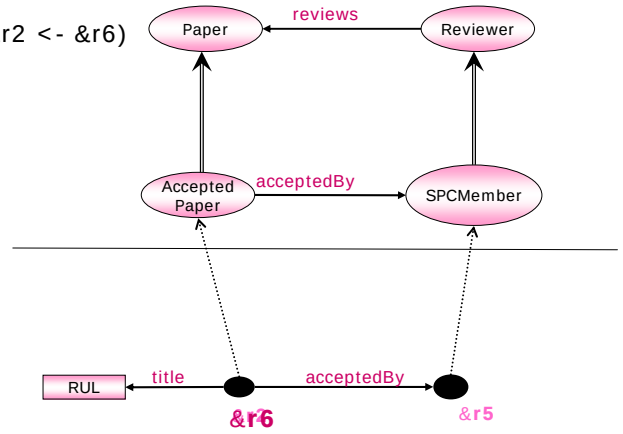
DELETE PCMember(&r5)
DELETE rdf:Resource(&r5)



8

Modify Class Instances

MODIFY AcceptedPaper(&r2 <- &r6)

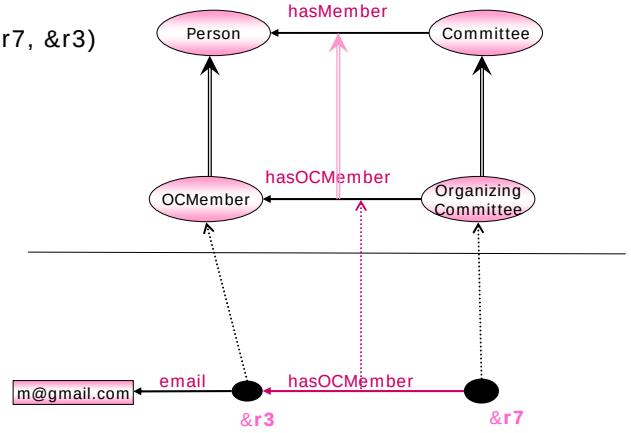


&r6: http://iswc05.semanticweb.org/accepted2#

9

Insert Property Instances

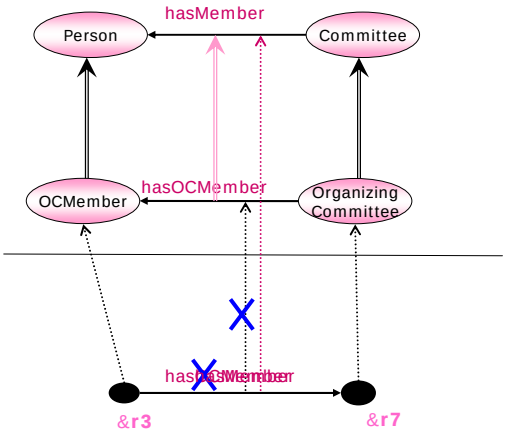
INSERT hasOCMember(&r7, &r3)



10

Delete Property Instances

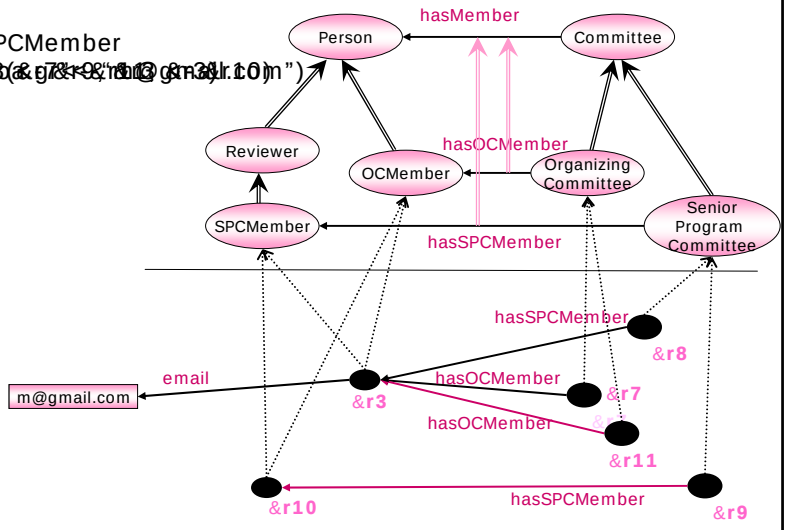
DELETE hasOCMember(&r7, &r3)



11

Modify Property Instances

MODIFY hasSPCMember



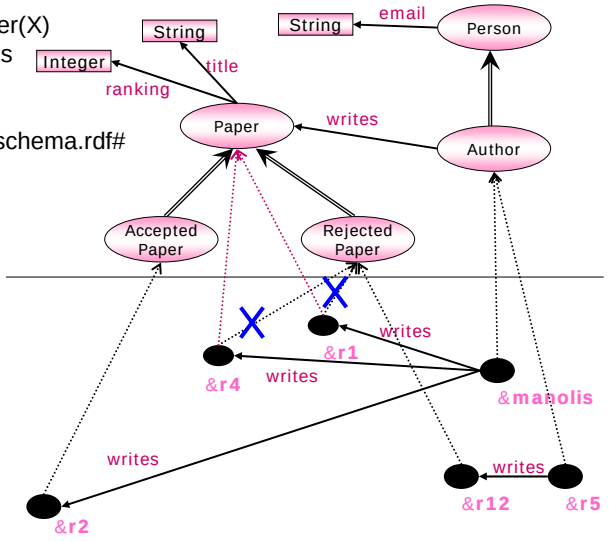
&r8: http://iswc05.semanticweb.org/spc#
&r9: http://iswc06.semanticweb.org/spc#
&r10: http://www.tuc.gr/~k

12

Set-Oriented Updates

Delete all instances of Rejected Paper(X)
Rejected Paper(X) whose author is
&manolis

```
DELETE ns:RejectedPaper(X)
FROM {Y}:ns:writes{X}
WHERE Y = &manolis
USING NAMESPACE
        ns=www.ex.org/schema.rdf#
```



&r12: http://iswc05.semanticweb.org/rejected3#

13

More Expressive Updates (I)

- Sequences of updates of the same kind

  Example:

  Reject all papers with ranking less than 4, and add the SPC Member responsible for the paper as the person who made the final recommendation

  RUL statement:

  INSERT RejectedPaper(X), rejectedBy(X, Y)

  FROM {X;Paper}ranking{Z},
  {X}submittedTo.hasSPC.hasMember{Y;SPCMember},
  {Y}isResponsibleFor{X}

  WHERE Z < 4

14

More Expressive Updates (II)

- Sequences of updates of different kinds

  Example:

  Form the Program Committee of ISWC06 by taking the set of all PC members of ISWC05 except those that reviewed less than 5 papers for ISWC05, and adding to these set the member of the OC of ISWC05.

  RUL statement:

```
INSERT hasPCMember(&http://iswc06.semanticweb.org#pc, X)
DELETE hasPCMember(&http://iswc06.semanticweb.org#pc, Y)
INSERT hasPCMember(&http://iswc06.semanticweb.org#pc, Z)
FROM {W}hasPCMember{X}, {W}hasPCMember{Y},
{W}hasOCMember{Z}
WHERE W = &http://iswc05.semanticweb.org#pc and
          count(SELECT Q FROM {Y}reviews{Q},
{Q}submittedTo{W}) < 5
```

15

Previous Work

- Other RDF update languages
  - Modification Exchange Language (MEL) [Nejdl et al. 2002]
  - rdfDB Query Language [Guha]
- RDF Store update APIs
  - RDF NetAPI [Seaborne 2002]
  - Sesame [Broekstra et al. 2002]
  - KAON [Oberle et al. 2002]
- Updates in data models similar to RDF
  - Telos [Mylopoulos et al. 1990, Plexousakis 1993]

16

Conclusions – Future Work

- RUL is an expressive declarative update language for RDF graphs.
- Our approach can be used to add update functionalities to other RDF query languages.
- There is an ongoing implementation using the existing ICS-FORTH RQL code base.
- Future plans:
  - What is the expressive power of RUL?
  - Schema updates and evolution.
  - Active rule languages for RDF.

17

RUL Syntax

- Class Instances

INSERT/DELETE QualClassName(ResourceExp) /
MODIFY QualClassName(OldResourceExp <- NewResourceExp)
[FROM VariableBinding] [WHERE Filtering]
[USING NAMESPACE NamespaceDefs]

- Property Instances

INSERT/DELETE QualPropertyName(SubjectExp, ObjectExp) /
MODIFY QualPropertyName([OldSubjectExp <-] NewSubjectExp,
[OldObjectExp <-] NewObjectExp)
[FROM VariableBinding] [WHERE Filtering]
[USING NAMESPACE NamespaceDefs]

18