PSELO:
Mediation for the
Edutella P2P Infrastructure
Talk overview

• Describe principles of PSELO system
• Short intro. to RDF and Edutella
• Overview how PSELO is integrated with Edutella
• P2P infrastructure for learning objects
• Mobile meta-data
• Publications
Problem Area

• Learning material, *Learning Objects*, are stored in several different data sources, e.g. web pages, databases, servers, etc.

• *Student* should be able to query teaching material on subject on his own.

• *Tutor* should be able to focus student on learning objects and terminology relevant for the subject.

• *Student* should be able to change focus as the learning progresses.
Approach

• Student explores material through middleware, PSELO.

• Learning objects described though semantic web meta-data descriptions, using RDF and RDF-Schema meta-data description languages.

• PSELO provides functionality for high-level ‘semantic’ queries and personal views in terms of taught subject

• Both tutor and student can define, modify, and extend semantic views
Personalized Search Engine for Learning Objects

Application interface

PSELO

Subject
RDFSchemas

Local data

Local semantic views

Learning sources

www
RDFAmos semantic web mediator


• RDFAmos can:
  - Access any RDF(S) data on web
  - Automatically generate semantic views of RDF-Schema meta-data descriptions
  - Process semantic queries over accessed data
  - Manage local user views and tables
  - Define mediating views combining RDF(S) sources with other kinds of data (e.g. relational databases)
PSELO semantic web mediator

- PSELO subsystems:
  - RDFAmos
  - Interfaces to Edutella P2P infrastructure for web-based learning material
RDF

• RDF describes web resources (URIs) using set of statement triples (in XML syntax):
  \(<subject,predicate,object>\)
where subject is a URI to be annotated.
  
  predicate is an annotation property
  
  object is value of property as URI
  or literal(string)

• NB: This is a binary relational model. Schema is ONE relation:
  
  statement(subject,predicate,object)

• Querying => Graph traversal through self joins, RQL
RDF-Schema

- Problem with RDF:
  - Properties not predefined.
  - Difficult to query and understand.

- RDF-Schema:
  - Classification of resources
  - Each class of resource has predefined properties
  - Defined using RDF
  - Data may be *mix* of RDF and RDF-Schema

- RDFAmos allows *semantic* queries:
  - Queries expressed in terms of schema, not data.
  - Over classes (types) and functions (properties)
RDF Schema Example

In http://user.it.uu.se/~udbl/RDFAmos/Schema/schema:

```xml
<rdf:RDF xml:lang="en"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
    <rdf:Description ID="Book">
        <rdf:type resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
        <rdfs:subClassOf rdf:resource="http://www.w3.org/2000/01/rdfschema#Resource"/>
    </rdf:Description>

    <rdf:Description ID="Title">
        <rdf:type resource="http://www.w3.org/2000/01/rdf-schema#Property"/>
        <rdfs:domain rdf:resource="#Book"/>
        <rdfs:range rdf:resource="http://www.w3.org/2000/01/rdf-schema#Literal"/>
    </rdf:Description>

    <rdf:Description ID="AI_Book">
        <rdf:type resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
        <rdfs:subClassOf rdf:resource="#Book"/>
    </rdf:Description>
</rdf:RDF>
```
Relational RDF representation

Statement

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>http://.../#Book</td>
<td>http://.../#type</td>
<td>http://.../#Class</td>
</tr>
<tr>
<td>http://.../#AI-Book</td>
<td>http://.../#type</td>
<td>http://.../#Class</td>
</tr>
<tr>
<td>http://.../#AI-book</td>
<td>http://.../#subClassOf</td>
<td>http://.../#Book</td>
</tr>
<tr>
<td>http://.../#Title</td>
<td>http://.../#type</td>
<td>http://.../#Property</td>
</tr>
<tr>
<td>http://.../#Title</td>
<td>http://.../#domain</td>
<td>http://.../#Book</td>
</tr>
<tr>
<td>http://.../#Title</td>
<td>http://.../#range</td>
<td>http://.../#literal</td>
</tr>
</tbody>
</table>
RDF Schema Data

http://user.it.uu.se/~udbl/RDFAmos/Schema/data:

<rdf:RDF xml:lang="en"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:sch="http://user.it.uu.se/~udbl/RDFAmos/Schema/schema#" >
  <sch:Book about="http://www.xyz.com/sw.html">
    <sch:Title>Software Engineering</sch:Title>
  </sch:Book>
  <sch:AI_Book about="http://www.xyz.com/ai.html">
    <sch:Title>Artificial Intelligence</sch:Title>
  </sch:AI_Book>
  <sch:Book about="http://www.xyz.com/pl.html">
    <sch:Title>Prolog</sch:Title>
  </sch:Book>
</rdf:RDF>
## Relational RDF representation

### Statement

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>http://#Book</td>
<td>http://#type</td>
<td>http://#Class</td>
</tr>
<tr>
<td>http://#AI-Book</td>
<td>http://#type</td>
<td>http://#Class</td>
</tr>
<tr>
<td>http://#AI-book</td>
<td>http://#subClassOf</td>
<td>http://#Book</td>
</tr>
<tr>
<td>http://#Title</td>
<td>http://#type</td>
<td>http://#Property</td>
</tr>
<tr>
<td>http://#Title</td>
<td>http://#domain</td>
<td>http://#Book</td>
</tr>
<tr>
<td>http://#Title</td>
<td>http://#range</td>
<td>http://#literal</td>
</tr>
<tr>
<td><a href="http://sw.html">http://sw.html</a></td>
<td>http://#type</td>
<td>http://#Book</td>
</tr>
<tr>
<td><a href="http://sw.html">http://sw.html</a></td>
<td>http://#title</td>
<td>Software Engineering</td>
</tr>
<tr>
<td><a href="http://ai.html">http://ai.html</a></td>
<td>http://#type</td>
<td>http://#AI-Book</td>
</tr>
<tr>
<td><a href="http://ai.html">http://ai.html</a></td>
<td>http://#title</td>
<td>Artificial Intelligence</td>
</tr>
<tr>
<td><a href="http://pl.html">http://pl.html</a></td>
<td>http://#type</td>
<td>http://#Book</td>
</tr>
<tr>
<td><a href="http://pl.html">http://pl.html</a></td>
<td>http://#title</td>
<td>Prolog</td>
</tr>
</tbody>
</table>
RDF Wrapper

• General Amos II wrapper from RDF to functional data model (FDM)

• Basic Functional RDF Representation (BFRR) can represent *any* accessed RDF meta-data description

• HP’s ARP2 RDF(S) parser used
RDFS Wrapper

• Each RDF Schema represented as *automatically generated* functional views over BFRR.

• *Functional queries over both* RDFS and semantically poorer RDF simultaneously supported.
RDFAmos basic functional RDF representation

BFRR

String

Resource

Object

Tuple

Statement

nickname

uri

object

subject

predicate

triple

Inheritance

Function
RDFAmos functional schema for an RDF Schema

Object

Resource

Book

Literal

AI-Book

↑ Inheritance

↑ Function

title
Example Semantic Query

```
select distinct X
from Book X, AI_Book Y
where title(X) = 'Artificial Intelligence'
    or X = Y;
```
Edutella

• Infrastructure for peer-to-peer distribution of learning material

• Application for SUN’s JXTA environment for P2P systems

• Uses an RDF-Schema for description of learning material

• Uses RDF based query language QEL3

• Peer data exchanges as RDF statements

• Can hook up Edutella *providers* which are peers that answer QEL3 queries
PSELO as Edutella provider

• PSELO is RDFAmos hooked up as Edutella provider

• PSELO delivers result through Edutella infrastructure

⇒ PSELO becomes QEL3 Edutella peer that can mediate many kinds of data sources

⇒ QEL3 queries over dynamically mediated RDF statements

• Also developing PSELO wrapper for Edutella sources
Edutella integration
Edutella Queries

• *QEL3* queries are data too.

• *QEL3* queries instances of *QEL3 meta-schema*

• *QEL3 meta-schema* imported as any other RDFS schema
  => Become type and function definitions in PSELO

• *QEL3 queries* imported as any other instances
  => Data describing queries in PSELO

• *QEL3 queries are objects* in mediator
QEL3 Query

<?xml version='1.0' encoding='ISO-8859-1'?>
<!DOCTYPE rdf:RDF >
<!ENTITY rdf 'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
<!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
<!ENTITY dc 'http://purl.org/dc/elements/1.1/'>
<!ENTITY edu 'http://www.edutella.org/edutella#'>
<!ENTITY lit 'http://www.lit.edu/types#'>
>
<rdf:RDF xmlns:rdf="&rdf;"
xmlns:rdfs="&rdfs;"
xmlns:dc="&dc;"
xmlns:edu="&edu;"
xmlns:lit="&lit;">
  <edu:QEL3Query rdf:about="#genQuery">
    <edu:hasRule rdf:resource="#r0"/>
    <edu:hasRule rdf:resource="#r1"/>
    <edu:hasQueryLiteral rdf:resource="#st0"/>
    <edu:hasResultType rdf:resource="&edu;TupleResult"/>
  </edu:QEL3Query>

  <edu:Variable rdf:about="#X"
rdfs:label="X"/>

  <edu:Rule rdf:about="#r0">
    <edu:hasHead rdf:resource="#st1"/>
    <edu:hasBody rdf:resource="#st2"/>
    <edu:hasBody rdf:resource="#st3"/>
  </edu:Rule>

  <edu:Rule rdf:about="#r1">
    <edu:hasHead rdf:resource="#st4"/>
    <edu:hasBody rdf:resource="#st5"/>
  </edu:Rule>

  <edu:StatementLiteral rdf:about="#st0">
    <edu:predicate rdf:resource="#aibook"/>
    <edu:arguments rdf:resource="#st0_seq"/>
  </edu:StatementLiteral>
  <rdf:Seq rdf:about="#st0_seq">
    <rdf:_1 rdf:resource="#X"/>
  </rdf:Seq>
  <edu:RDFReifiedStatement rdf:about="#st2">
    <rdf:subject rdf:resource="#X"/>
    <rdf:object rdf:resource="&lit;Book"/>
    <rdf:predicate rdf:resource="&rdf;type"/>
  </edu:RDFReifiedStatement>
  <edu:RDFReifiedStatement rdf:about="#st3">
    <rdf:object>Artificial Intelligence</rdf:object>
    <rdf:subject rdf:resource="#X"/>
    <rdf:predicate rdf:resource="&dc;title"/>
  </edu:RDFReifiedStatement>
  <edu:StatementLiteral rdf:about="#st4">
    <edu:predicate rdf:resource="#aibook"/>
    <edu:arguments rdf:resource="#st4_seq"/>
  </edu:StatementLiteral>
  <rdf:Seq rdf:about="#st4_seq">
    <rdf:_1 rdf:resource="#X"/>
  </rdf:Seq>
  <edu:RDFReifiedStatement rdf:about="#st5">
    <rdf:subject rdf:resource="#X"/>
    <rdf:object rdf:resource="&lit;AI-Book"/>
    <rdf:predicate rdf:resource="&rdf;type"/>
  </edu:RDFReifiedStatement>
</rdf:RDF>
QEL3 Query Translation

• Declarative PSELO function constructs PSELO query string from QEL3 query object structure:

  \texttt{query\_string(QEL3Query) \rightarrow Charstring}

• PSELO function \texttt{eval} executes query string
Mobile meta-data

• Meta-data so far assumes all properties location independent

• Mobile meta-data objects:
  E.g. Describe cars in given area

• Mobile client:
  E.g. Describe service stations close to this car’s current position

• Dynamic location dependent view of meta-data
New book:

• P. Gray, L. Kerschberg, P. King, and A. Poulavassilis (eds.):

3 chapters about group’s technology, incl. PSELO
Publications:


Sponsorship

PSELO is part of PADLR project

• Sponsored by Wallenberg Global Learning Lab.
• Cooperation Uppsala, KTH, Hannover, Karlsruhe, Stanford