

Università degli Studi di Bari



*Laboratorio di Acquisizione
della Conoscenza e
Apprendimento delle Macchine*



Tutorial

Daniele Capursi

Luigi Iannone

Dip. Informatica - Università di Bari



Overview

- **What is RDF ?**
- **Resources**
- **RDF Data Model**
- **Interchange syntax**
- **RDF Containers**



What is RDF?

- **Resource Description Framework (RDF) is a simple framework for the description of Knowledge in the World Wide Web, enabling the encoding, exchange, and reuse of structured information.**
- **It was specifically designed for the description of “metadata” (data about data) related to resources available through the World Wide Web.**
- **It is a W3C recommendation.**
- **It is at the base of what is called “the Semantic Web”, as it permits to share knowledge on the Web.**
- **Latest version drafts released on 22 February 1999**



The Web and the Semantic Web...

- **The World Wide Web was invented by Tim Berners-Lee in the 1989 allowing CERN scientists to share scientific documents across the World.**
- **Nowadays the Web is the largest collection of document ever produced by the Humanity.**
- **Web should allow (says Berners – Lee)**
 - to share information between men,
 - to accomplish tasks using the Web



What's wrong with the Web...

- **The Web is big**
- **The Web contains information human tailored**
 - **Machine cannot understand the meaning of a document**
 - **Only syntactic techniques can be exploited (keywords as such)**



...What's wrong with the Web...

- Polisemy

– A term can have multiple meanings



Low precision

- Synonymy

– Several terms
Having same
meaning



Low recall

Netscape: Google Search: MIT computer science chair

File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print Security Shop Stop

Bookmarks Location: <http://www.google.com/search?q=MIT+cor> What's Related

Advanced Search Preferences Search Tips

Google™ MIT computer science chair Google Search

I'm Feeling Lucky

Searched the web for MIT computer science chair. Results 1 - 10 of about 99,500. Search took 0.84 seconds.

Category: [Science](#) > [Math](#) > [Logic and Foundations](#) > [Conferences](#)

[UniDo, Informatik](#)
... technischer Prozesse und Systeme mit Methoden der Computational Intelligence ... in der Networked Computer Science Technical Reports Library (NCSTRL) ... [www.cs.uni-dortmund.de/ - 11k - Cached - Similar pages](#)

[Committee on the Status of Women in Computing Research](#)
... CRA-W co-chair, is a Professor in the Computer Science Department at the ... by the National Science Foundation and EOT ... Nancy G. Leveson MIT Susan Owicki ... Description: Takes positive action to increase the number of women participating in computer science and engineering... Category: [Society](#) > [People](#) > [Women](#) > [Career](#) > [Science](#) [www.cra.org/Activities/crawl/ - 21k - Cached - Similar pages](#)

[Professor Deborah Estrin, UCLA Computer Science Department and ...](#)
... MS (1982) from MIT and her BS ... a member of the Computer Networks Division at ... received the National Science Foundation, Presidential ... and study-chair for DARPA's ... [lecs.cs.ucla.edu/~estrin/ - 22k - Cached - Similar pages](#)

[IEEE Symposium on Logic in Computer Science](#)
... to bibliographies at MIT and University of ... ox.ac.uk, Publicity Chair Martin Grohe Department ... Mathematics, Statistics, and Computer Science University of ... Description: IEEE Symposium on Logic in Computer Science Category: [Computers](#) > [Computer Science](#) > [Conferences](#) [sun1.mathematik.uni-freiburg.de/lics/ - 4k - Cached - Similar pages](#)

[IEEE Symposium on Logic in Computer Science](#)

100%



...What's wrong with the Web

- How to integrate the information from several sources?
- How to perform task integrating several services available on the Web? E.g.:
 - “I want to book a ticket for “Matrix Reloaded” and a dinner at an Italian Restaurant within five blocks from the cinema”.
- So I need computers understanding information reasoning about it (Computers dealing with Knowledge).



The Semantic Web...

- **The Semantic Web is a vision for the future of the Web in which information is given explicit meaning, making it easier for machines to automatically process and integrate information available on the Web.**
- **The Semantic Web turns the Web into a huge Knowledge Base.**
- **The Web is a network of information linked each other, to be consumed by Humans.**
- **The Semantic Web is a network of information semantically linked, to be consumed by computers.**



...The Semantic Web

- **The Web already offer the infrascture to publish information that everyone can reach and to integrate her information with the one provided by others (via the URL and links mechanisms).**
- **We need a framework that permits to express knowledge in a way that can be machine-processable.**
- **HOW????**



What we need to share Knowledge on the Web

- **We need a language that:**
 - Has a syntax that ensures syntactical interoperability
 - Has a semantic that is shared between computers
 - Is able to express ontologies
 - Permits very efficient elaborations, since the knowledge in the Web is very big
- **We need an infrastructure that:**
 - Permits to add semantic to the existing documents
 - Permits to refer to things unambiguously on the entire Web
 - Permits to integrate several source of knowledge in the distribute environment which is the Web



A possible solution: RDF

- **RDF principles:**
 1. **“Everything must be identified by URIs”.**
 2. **The “Least Power” principle:**
“When expressing anything, use the least expressive language”.
 3. **“Anything can say anything about anything”.**



RDF consists of...

- **Formal data model**
- **Interchange syntax**
- **Schema type systems (in RDF)**
- **Query language (planned)**



Resources

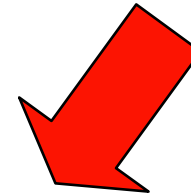
- All things described by RDF expressions are called *resources*.

- Example:

- Web pages
- Collections of pages
- Web sites
- Books
- People
- Feelings

not directly accessible via the Web

RDF was born to describe principally resources available via the Web, but not only these!





Resources and URIs...

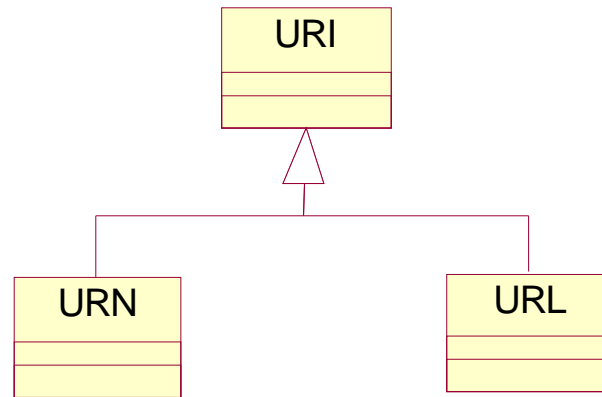
Everything must
be identified by
URIs

- RDF define a *Resource* anything that can be identified by an URI (Universal Resource Identifier).
- A URI is a name that refers to a unique object in the World.
- A URI can be a URN (Universal Resource Name) or a URL (Universal Resource Locator).
- URLs are URIs that can be used to retrieve the resource they refers to.

Not true when
they are used
just as URIs!!!



...Resources and URIs...



urn:IT_CPRDLG55D02A669S

urn:...

http://digilander.iol.it/daniele

ftp://ftp.microsoft.com

mailto:semeraro@di.uniba.it

http://www.daniele.com/amici#Luigi



...Resources and URIs

- Using URLs as URIs, it's easy to ensure uniqueness of the name
- Only the IETF can assign top level domains (e.g. .it, .com, .org, .gov)
- Only national organizations can assign secondary level domains (e.g. libero.it, tin.it, gazzetta.it)
- Only the owner of a domain can assign sub-domains and URLs of that domain (e.g. <http://www.libero.it>,
<http://www.gazzetta.it/fantacalcio>,
<http://digilander.iol.it/daniele/amici#Roberto>)



RDF data model

Least Power Principle

- Resources have associated
 - properties that have associated
 - » values
- Resources: identified by URIs
- Properties: identified by URIs
- Values: resources or primitive data types
(string)



RDF Statements: the triples

- The basic unit of information in RDF is the *Statement*:
 - an assertion that states that a certain resource (the *Subject*) has a certain value (the *Object*) for a certain property (the *Predicate*).
- So a statement is actually a *Triple* (Subject, Predicate, Object).
- A set of statements is called *Model*.



Finally some examples!!!

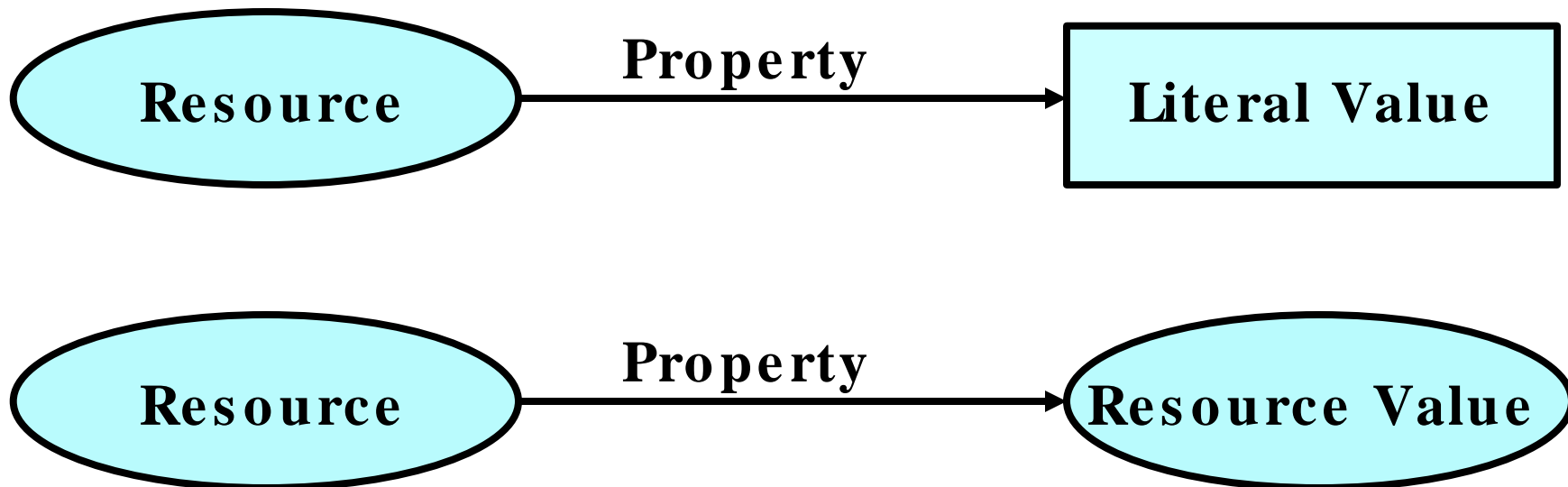
- **“The author of the RDF tutorial is Daniele Capursi”**
- **“The first name of Daniele Capursi is ‘Daniele’”**

Subject	Predicate	Object
http://www.di.uniba.it/~gci/RDF_Tutorial.ppt	http://purl.org/dc/elements/1.1/creator	mailto:capursi@sword.it
mailto:capursi@sword.it	http://www.myspace.com/properties#firstName	Daniele



RDF data model: directed labelled graph...

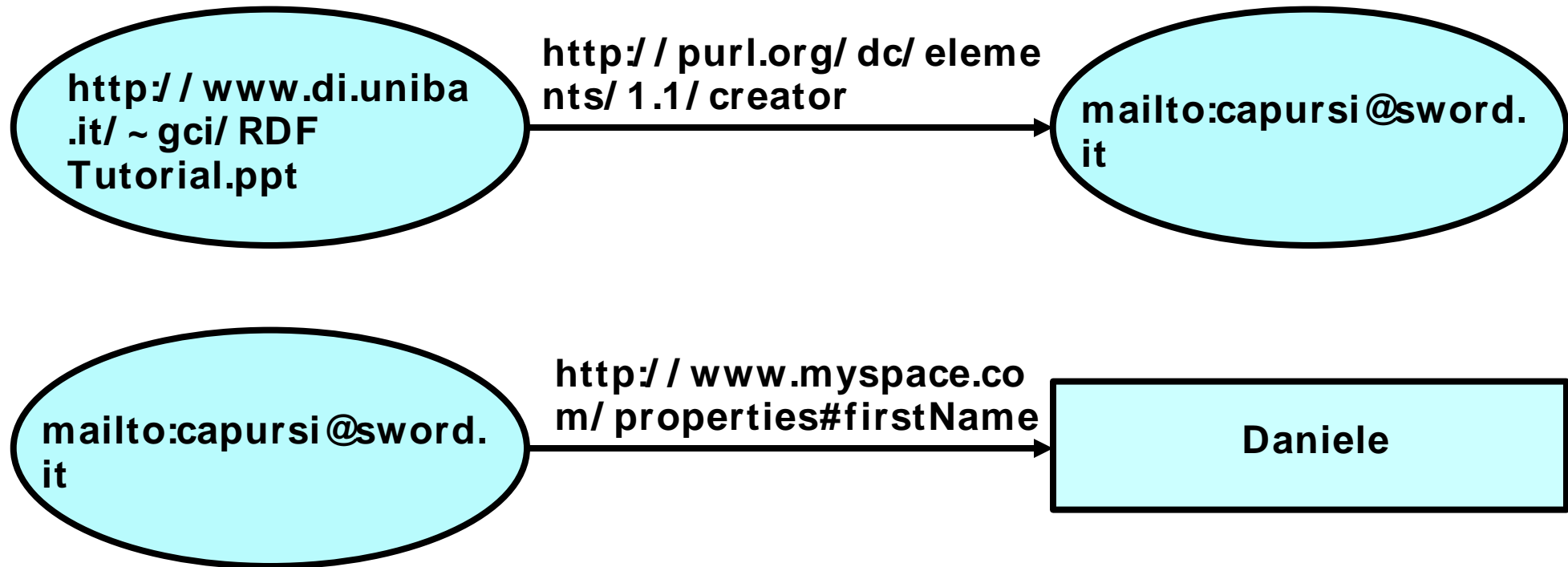
- One or more RDF Statements can be displayed as a very simple directed labelled graph:





...RDF data model: directed labelled graph...

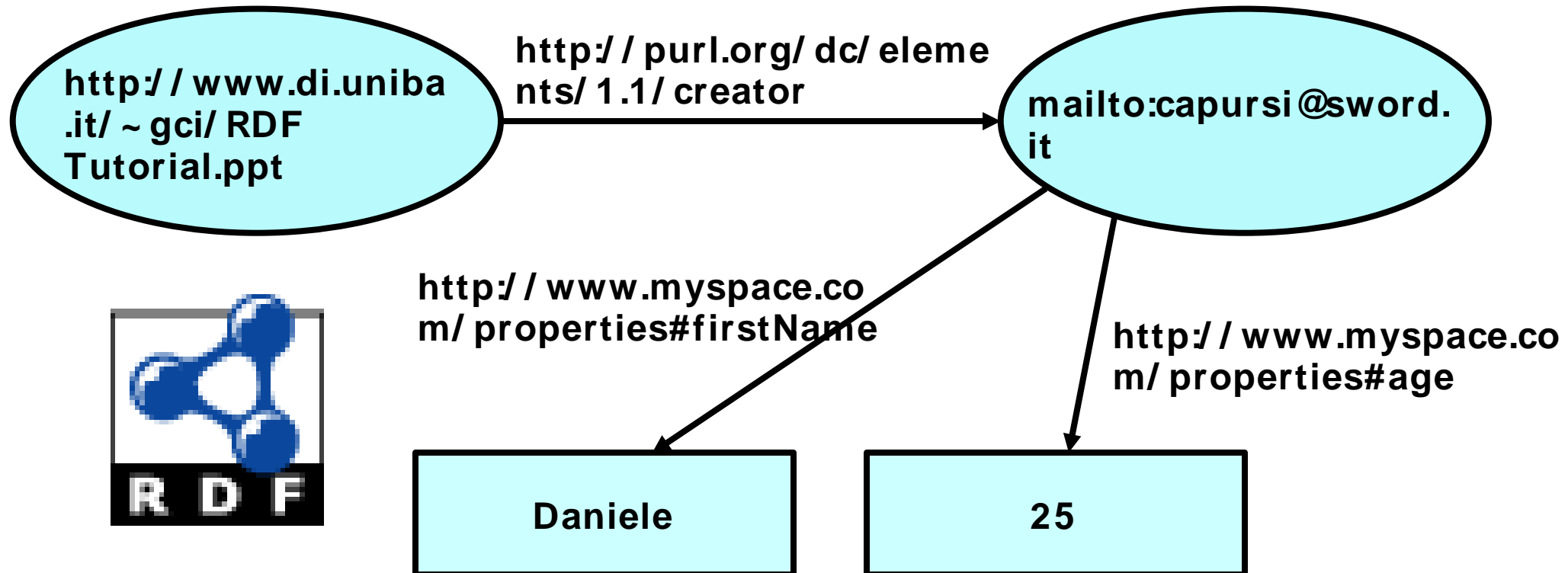
- The previous examples:





...RDF data model: directed labelled graph

- This permits to display the semantic relationships in a compact way:





Interchange syntax

- A serialization syntax is needed:
 - To save RDF Statements into a persistent form (e.g. a file)
 - To transfer them from one application to another
 - To permit their retrieval from the Web
- RDF Model & Syntax specification defines a standard XML encoding of the data model.
- Why XML encoding?
 - Parsers are easy to implement (are based on XML parsers)
 - XML deals with interoperability and internationalization issues
- Other syntax encoding exist
 - NTRIPLE
 - N3



The example encoded in XML

```
< ?xml version= '1.0'?>
< rdf:RDF
  xmlns:rdf= 'http:// www.w3.org/ 1999/ 02/ 22- rdf- syntax- ns#'
  xmlns:dc= 'http:// purl.org/ dc/ elements/ 1.0/ '
  xmlns:NS0= 'http:// www.myspace.com/ properties#'
>
  < rdf:Description rdf:about= 'http:// www.di.uniba.it/ ~ gci/ RDFTutorial.ppt'>
    < dc:creator rdf:resource= 'mailto:capursi@sword.it' / >
  </ rdf:Description>
  < rdf:Description rdf:about= 'mailto:capursi@sword.it'>
    < NS0:firstName> Daniele </ NS0:firstName>
    < NS0:age> 25 </ NS0:age>
  </ rdf:Description>
</ rdf:RDF>
```




The example encoded in Abbreviated XML

```
<?xml version='1.0'?>
```

```
<rdf:RDF
```

```
  xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
```

```
  xmlns:dc='http://purl.org/dc/elements/1.0/'
```

```
  xmlns:NS0='http://www.myspace.com/properties#'
```

```
>
```

```
<rdf:Description rdf:about='http://www.di.uniba.it/~gci/RDF  
Tutorial.ppt'>
```

```
  <dc:creator rdf:resource='mailto:capursi@sword.it'  
  NS0:firstName='Daniele' NS0:age='25'/>
```

```
</rdf:Description>
```

```
</rdf:RDF>
```



The example encoded in N-TRIPLE

```
< http://www.di.uniba.it/~gci/RDF Tutorial.ppt>  
< http://purl.org/dc/elements/1.0/creator>  
<mailto:capursi@sword.it> .
```

```
<mailto:capursi@sword.it>  
< http://www.myspace.com/properties#firstName>  
"Daniele" .
```

```
<mailto:capursi@sword.it>  
< http://www.myspace.com/properties#age>  
"25" .
```



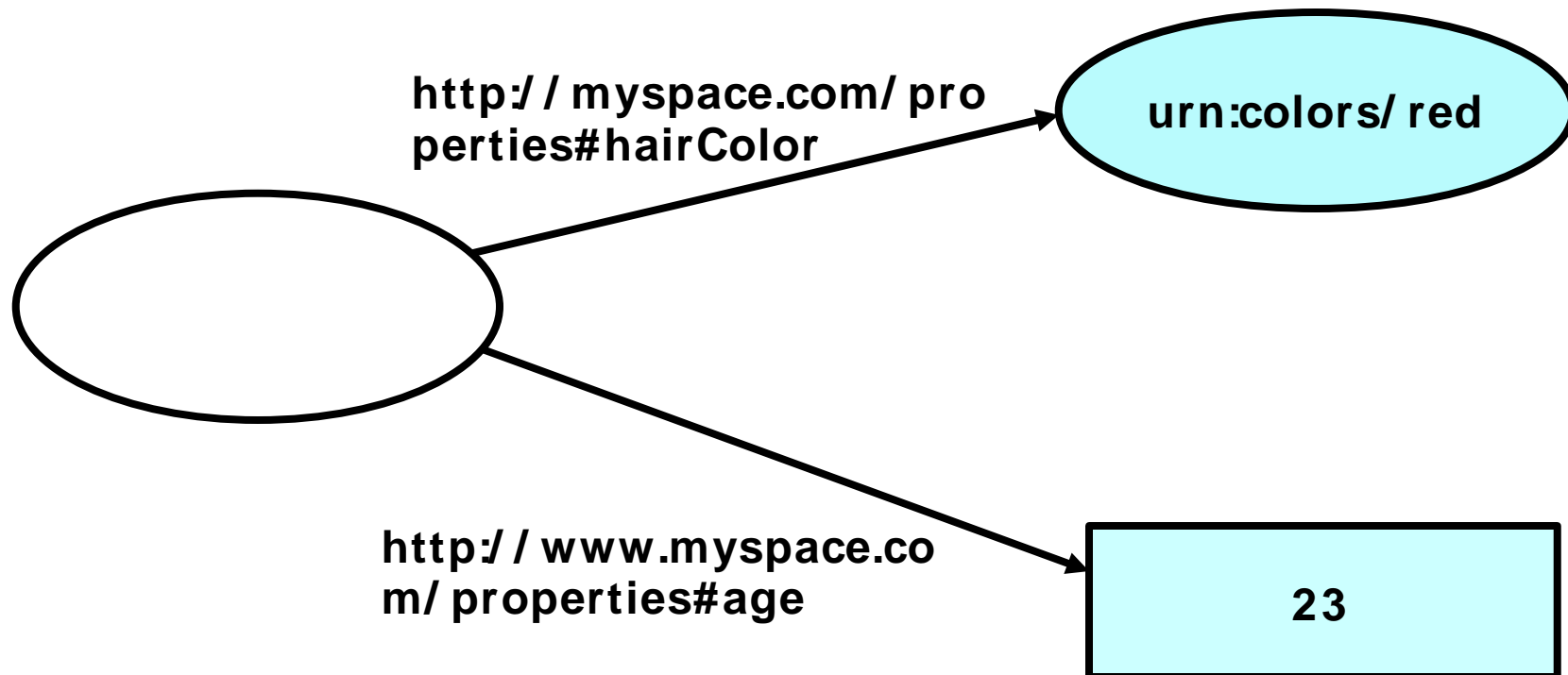
RDF IS NOT XML!!!

- **XML is just one of the possible way to serialize RDF.**
- **XML is the standard way to encode RDF for putting it into the Semantic Web, not to process it.**
- **There are more than one way to serialize RDF in XML.**
- **If a program manipulates RDF as XML, the code must be repeated for every possible XML serialization!**
- **Manipulating RDF as XML, we break the abstraction of RDF data model.**



More RDF features: blank nodes

“The girl with red hair is 23 years old”
What is the URI of the girl?





More RDF features: RDF Containers

- **Bag**
 - unordered list of resources or literals

- **Sequence**
 - ordered list of resources or literals

- **Alternatives**
 - alternate values
 - need to choose
 - at least one value
 - first value is default or preferred value

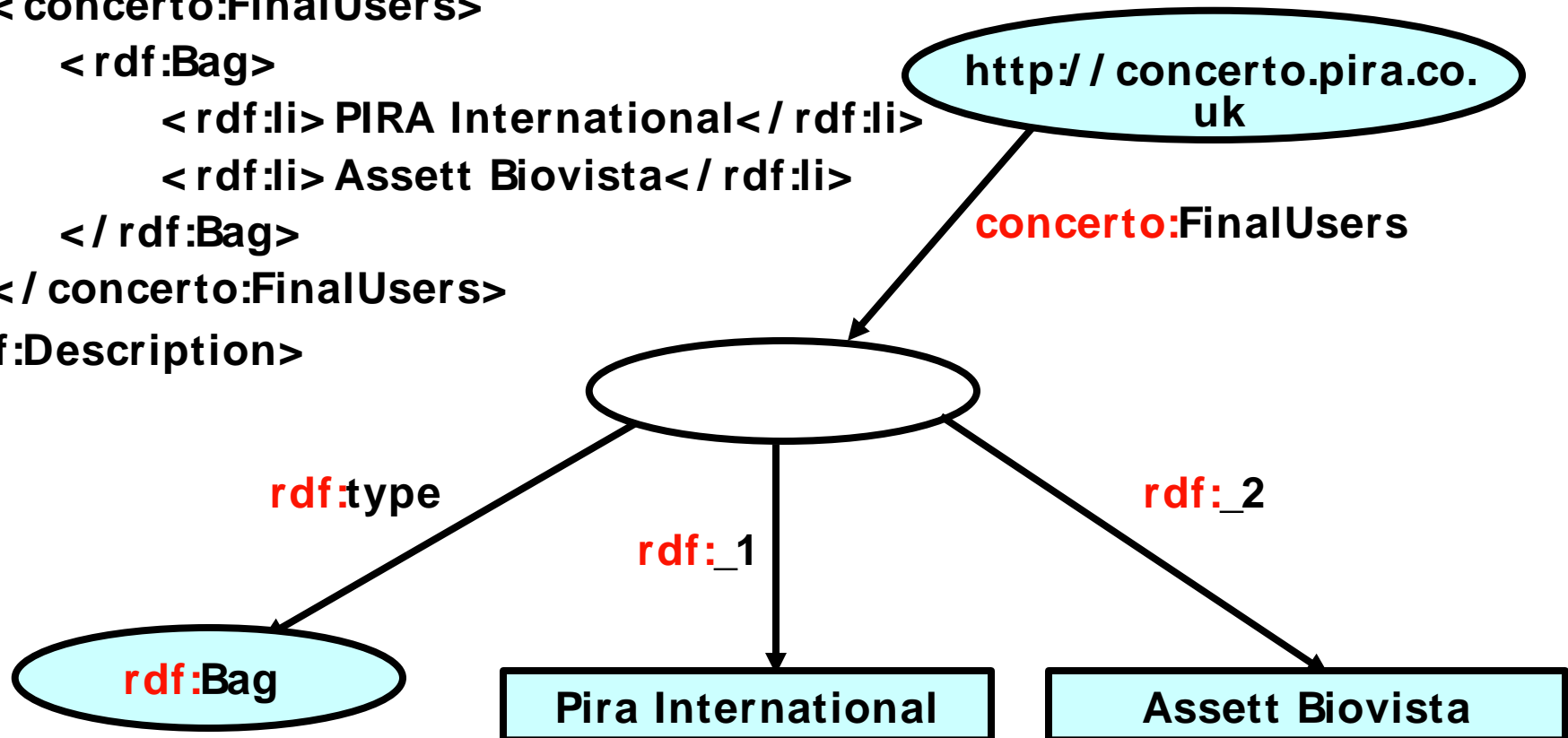


Collection example in RDF

```

<rdf:Description about="http://concerto.pira.co.uk">
  <concerto:FinalUsers>
    <rdf:Bag>
      <rdf:li> PIRA International</rdf:li>
      <rdf:li> Assett Biovista</rdf:li>
    </rdf:Bag>
  </concerto:FinalUsers>
</rdf:Description>

```

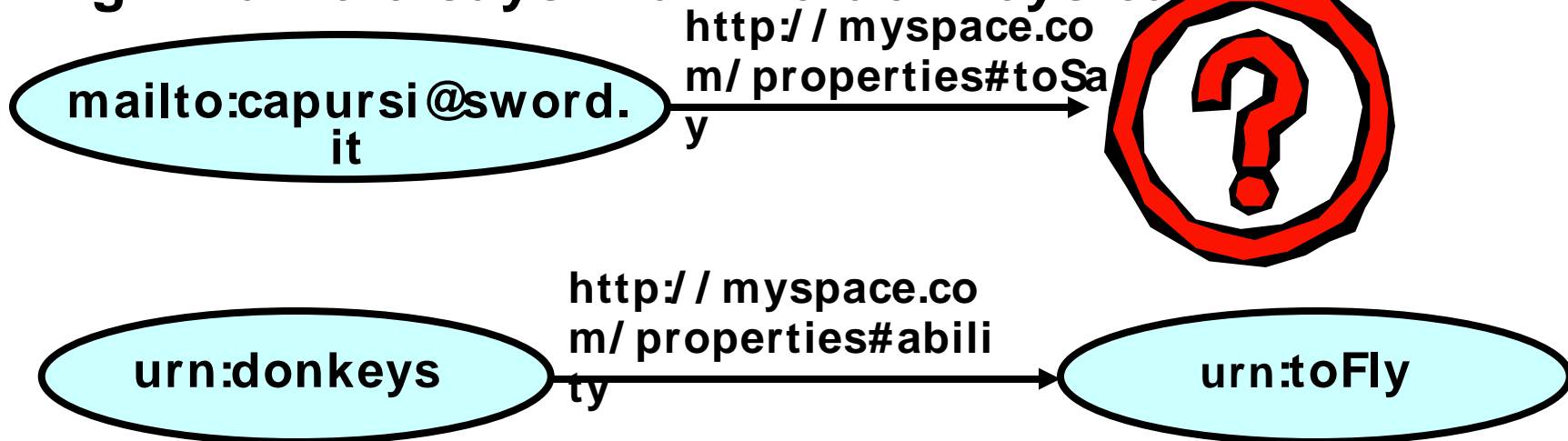




More RDF features: reification...

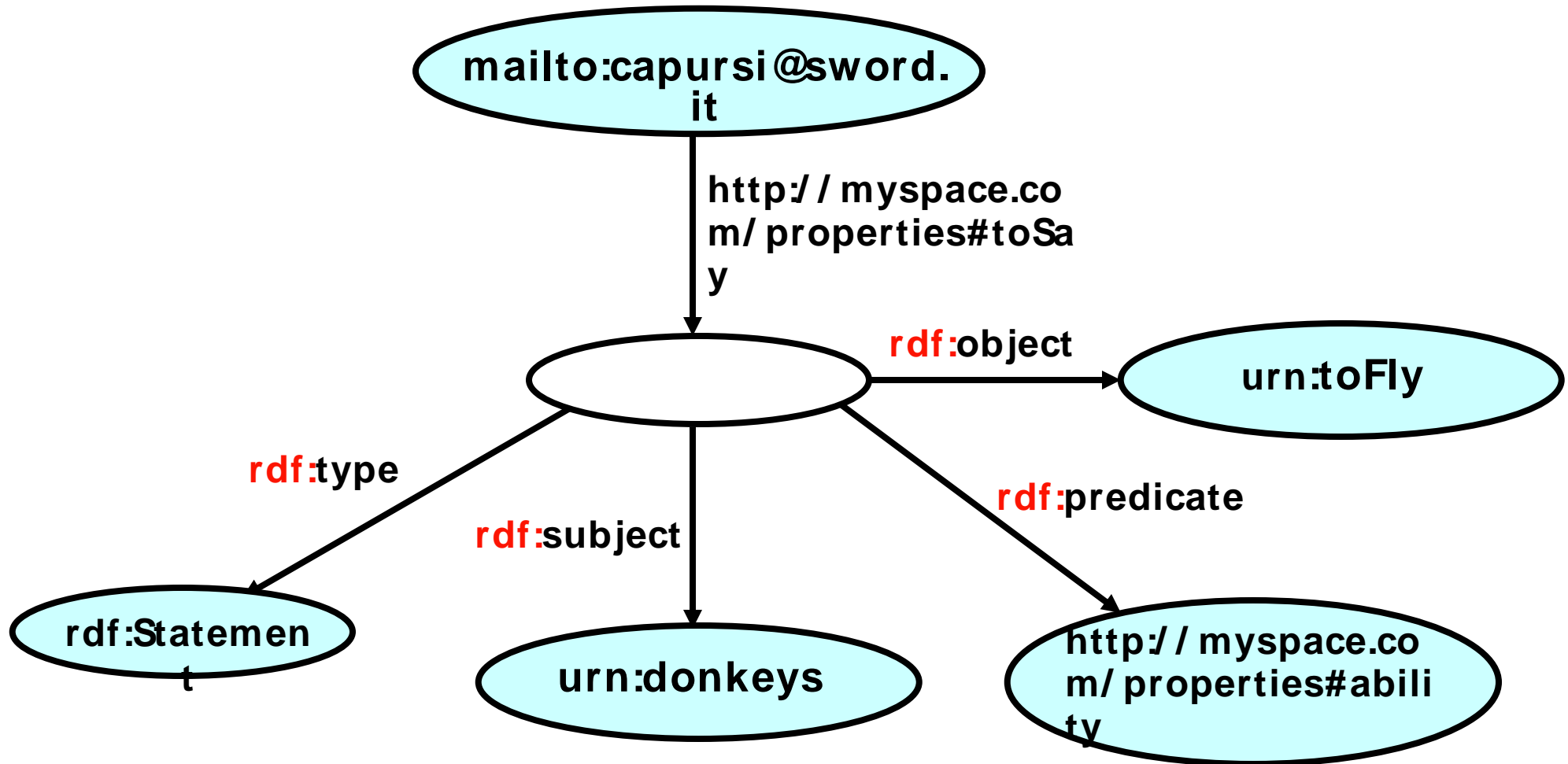
- A feature that is rarely used and has a cumbersome syntax: statements about statements.
- A reified statement is NOT asserted!

E.g. “Daniele says that the donkeys can fly”





...More RDF features: reification





RDF is good

- Now we see that RDF can give machine-understandable semantics to resources. Do we?
- Now we know that we need RDF for Semantic Web, while XML is not enough. Do we?

NO!!!



Is RDF good?...

- How does RDF give semantics?
- Why is RDF better than XML?
- XML tags and attributes doesn't have a semantics.
- Everybody can define his own tags or interpret the meaning of a tag in the way he likes.
- XML leaves too much freedom on how to represent information.





...Is RDF good?

- **RDF syntax is very poor: it's very likely that different people uses the same concepts (Resources and Properties) to describe the same things, apart from the URIs.**
- **If some people (or some programs) agree on how to assign URIs to some resources, they perfectly share the meaning of that resources (they know what objects correspond to that URIs),**
- **If they agree on the meaning of some properties, they cant perfectly understand what they represent.**



Still not enough



So, where is semantics?

- RDF has been designed to be used together with shared vocabularies.
- A Vocabulary is a set of Properties that have a precise meaning.
- There are lots of standard vocabularies for many domains.
- If we use only standard vocabularies, we share the semantics of our RDF statements.
- What if we want to define our own vocabularies?





The most famous example: The “Dublin Core Metadata Initiative” Vocabulary

- “Dublin Core” is a standard for the representation of essential metadata of Web documents
- Dublin Core 1.0 properties:
“<http://purl.org/dc/elements/1.0/>” + ...

–contributor

–coverage

–creator

–date

–description

–format

–identifier

–language

–publisher

–relation

–rights

–source

–subject

–title

–type



How to define a vocabulary: RDF Schema...

- RDF allow people to define new vocabularies (called “Schema”).
- RDFS (RDF Schema) is a language, based on RDF itself, to describe RDF vocabularies. An instance of RDFS is called “RDF Schema” too!
- A *Schema* is a set of *Classes* of resources and *Properties* with a precise semantics.
- A schema can define new elements from scratch or by “importing” concepts from other schemas and extending them.



...RDF Schema...

RDFS is a lightweight ontology language, since it allows to define Classes and Property of classes:

- **RDFS allow to define Classes:**
 - From scratch
 - As subclasses of existing ones
 - As equivalent to existing ones (latest RDF version)
- **RDFS allow to define Properties:**
 - From scratch
 - As sub- properties of existing ones
 - As equivalent to existing ones (latest RDF version)



...RDF Schema...

- **RDFS allow to specify properties of properties:**
 - **Domain**
I.e. the classes it applies to (the classes of the resources that can be the subject of a statement with that property as predicate)
 - **Range**
I.e. the classes of the resources that can be value of that property (the classes of the resources that can be the object of a statement with that property as predicate)



RDF Schema and RDF

- An RDF Schema is written in RDF itself.
- Obviously using properties and classes whose semantics is standard.

==>

- There is an RDF Schema that describes the elements of the RDF Schema language itself!
- So RDFS can be described in RDFS, that is still RDF.



RDFS: what for?

- An RDF Schema can be used to (semantically!) validate an RDF Model, in the same way we can validate a knowledge base against an ontology
- RDFS semantics defines some simple inferences, typical of an ontology language, e.g.:
 - If r is of class A and class A is subclass of class B , then r is of class B
 - If A is subclass of B and B is subclass of C then A is subclass of C
 - If the domain of a property p is the class A and p is the subject of a statement in which p is the predicate, then p is of class A



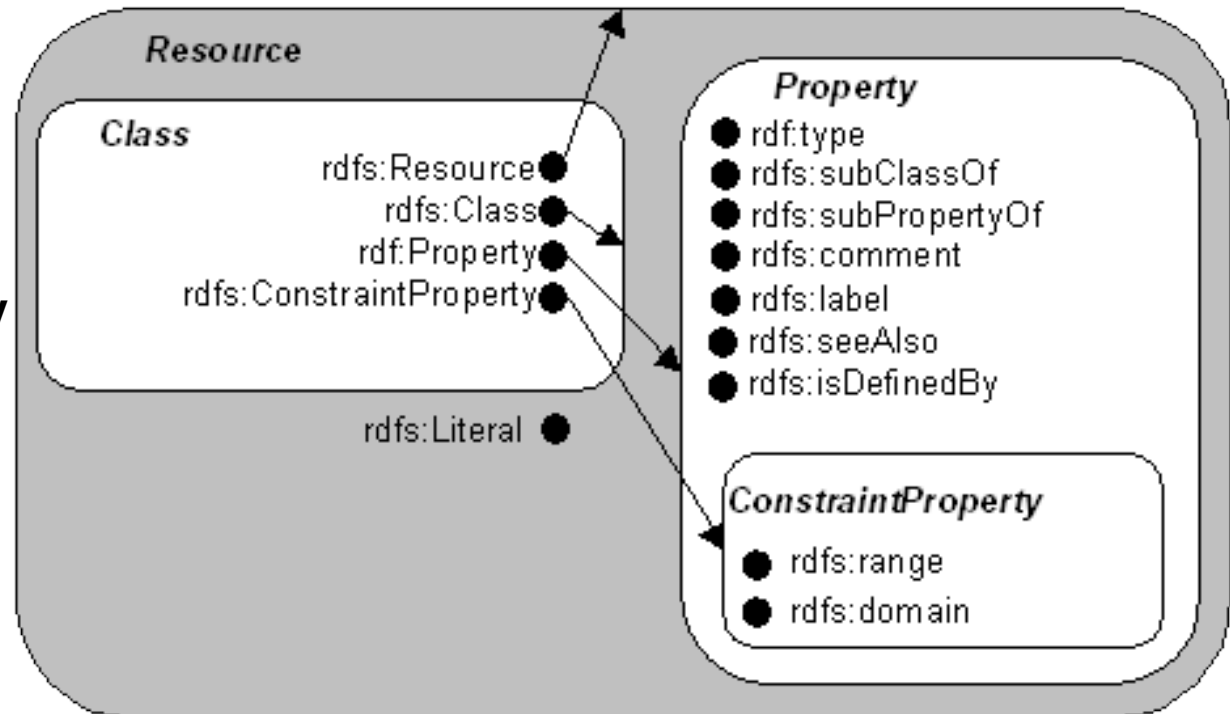
RDFS makes RDF smarter...

- **The possibility to reuse and extend existing schemas + such inferences makes RDF agents much smarter:**
 - If an agent encounters a property it doesn't know, but it knows one super-property of such property, he can perform meaningful processing as well



RDFS vocabulary

- **Classes:**
 - Class
 - Resource
 - Literal
 - **rdf:Property**
 - **ConstraintProperty**
- **Properties:**
 - type
 - subClassOf
 - subPropertyOf
 - comment
 - label
 - seeAlso
 - isDefinedBy
 - domain
 - range





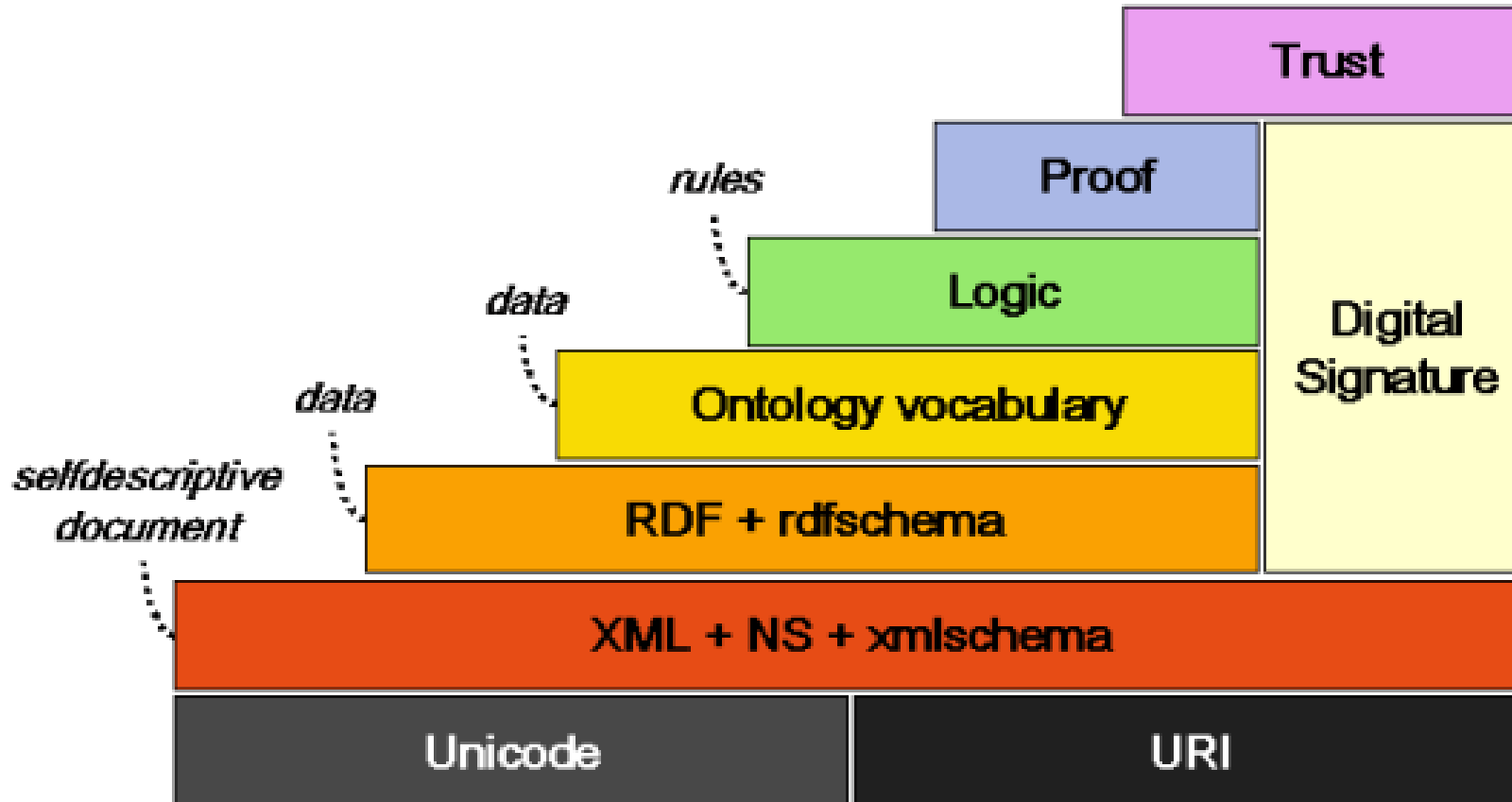
Some comments about RDFS

Anything can
say anything
about anything

- RDFS is “property- centric” VS OO is “class- centric”:
 - RDFS defines a property in terms of what classes can have that property and what classes can be objects of that property
 - Object Oriented defines classes in terms of property they own and of that type are the values of these properties
- Everybody can reference and reuse a Class or a Property defined by others, just using its URI. Everybody can reference a Resource defined by others.



Structure of the Semantic Web





RDF Core: A component for effective management of RDF Models



Overview

- **RDFCore functionalities**
- **RDFCore architecture**
- **RDFCore empirical evaluation**
- **Conclusions and future work**

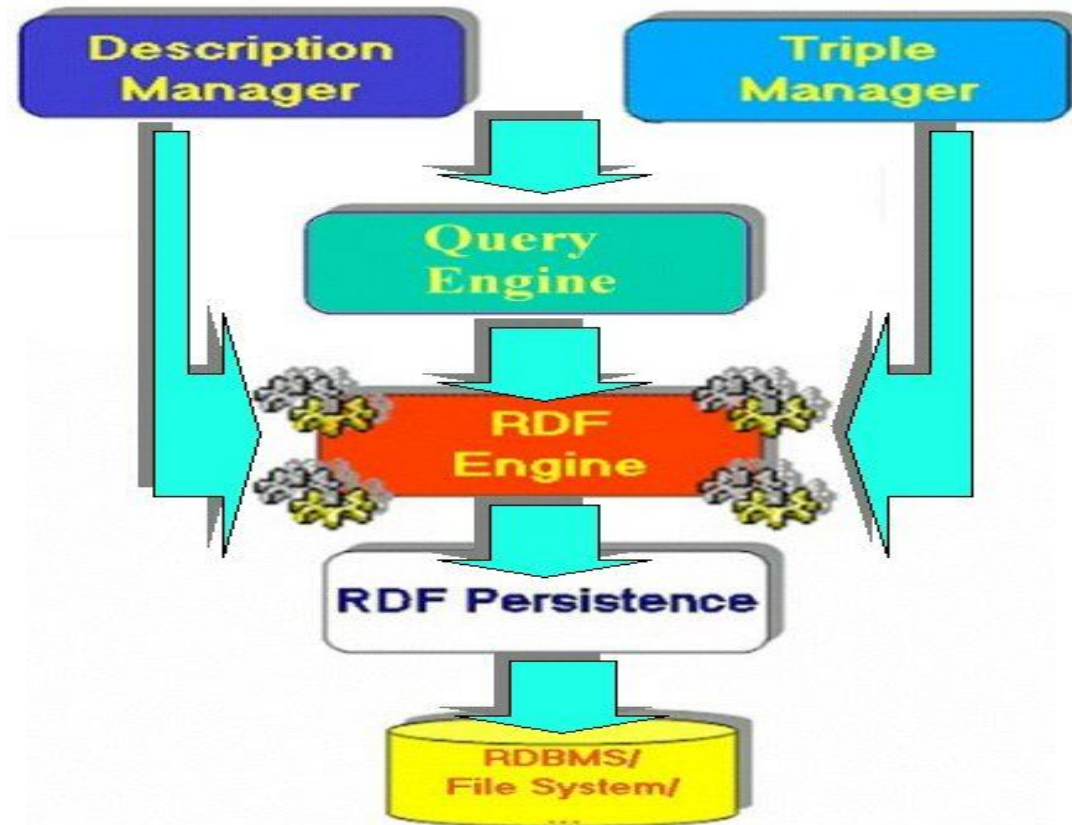


RDFCore functionalities

- **Support for all typical operations on RDF at two level of granularity:**
 - **Model (Description)**
 - **Statement (Triple)**
- **Multi- user environment**
- **Support for multiple query languages**
 - **RDQL**
 - **SquishQL**
 - **RQL (in progress)**



RDFCore architecture...





... RDFCore architecture

Strategy Pattern

- **Compliance with W3C Object Model**
- **Pluggable modules for:**
 - **Persistence**
 - **Query engines**



RDF Persistence implementations

- **Currently available:**
 - **File system XML based (relying on PDOM) used as baseline**
 - **Jena 2 based (more efficient especially in queries)**
 - **Plain File system (not relying on PDOM)**
 - **Jena 2 based**



Sneak Comparison

	RDF Suite	Jena Semantic Web Toolkit	RDFCore
Statements Granularity API	n	y	y
Model Granularity API	n	y	y
Multi-user complex support	n	n	y
RDQL	n	y	y
RQL	y	n	n
SquishQL	n	n	y
Scales up to (no of Triples)	6000000*	800000*	100000

* = Taken from W3C surveys



Conclusions

We achieved:

- **A standard framework that allows all typical operations on RDF (Models and Statements)**
- **Flexible architecture with the possibility of plugging in new modules seamlessly**
- **Linear scalability for the basic operations and queries even in this early prototype**



Future work

- **New persistence layers (e.g.: RSSDB based)**
- **Support for RQL**
- **More extensive tests and comparisons with existing systems**
- **Integration of RDFCore within wider architectures**



Further Readings



<http://www.w3.org/RDF>

Latest version drafts:

<http://www.w3.org/TR/2002/WD-rdf-syntax-grammar-20021108/>