Resource Description Framework

More Semantic for the Web!

RDF and XUL in Mozilla/Netscape 6

Johannes Henkel
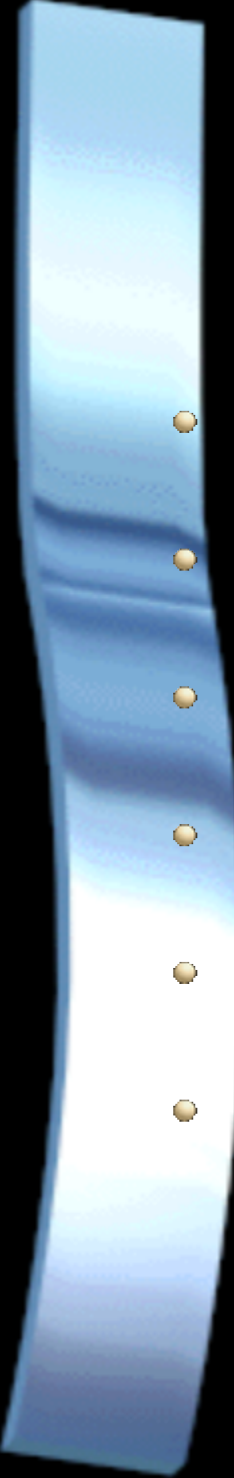jhenkel@jhenkel.de
www.jhenkel.de
Dec 14 2000, CSCI 7818

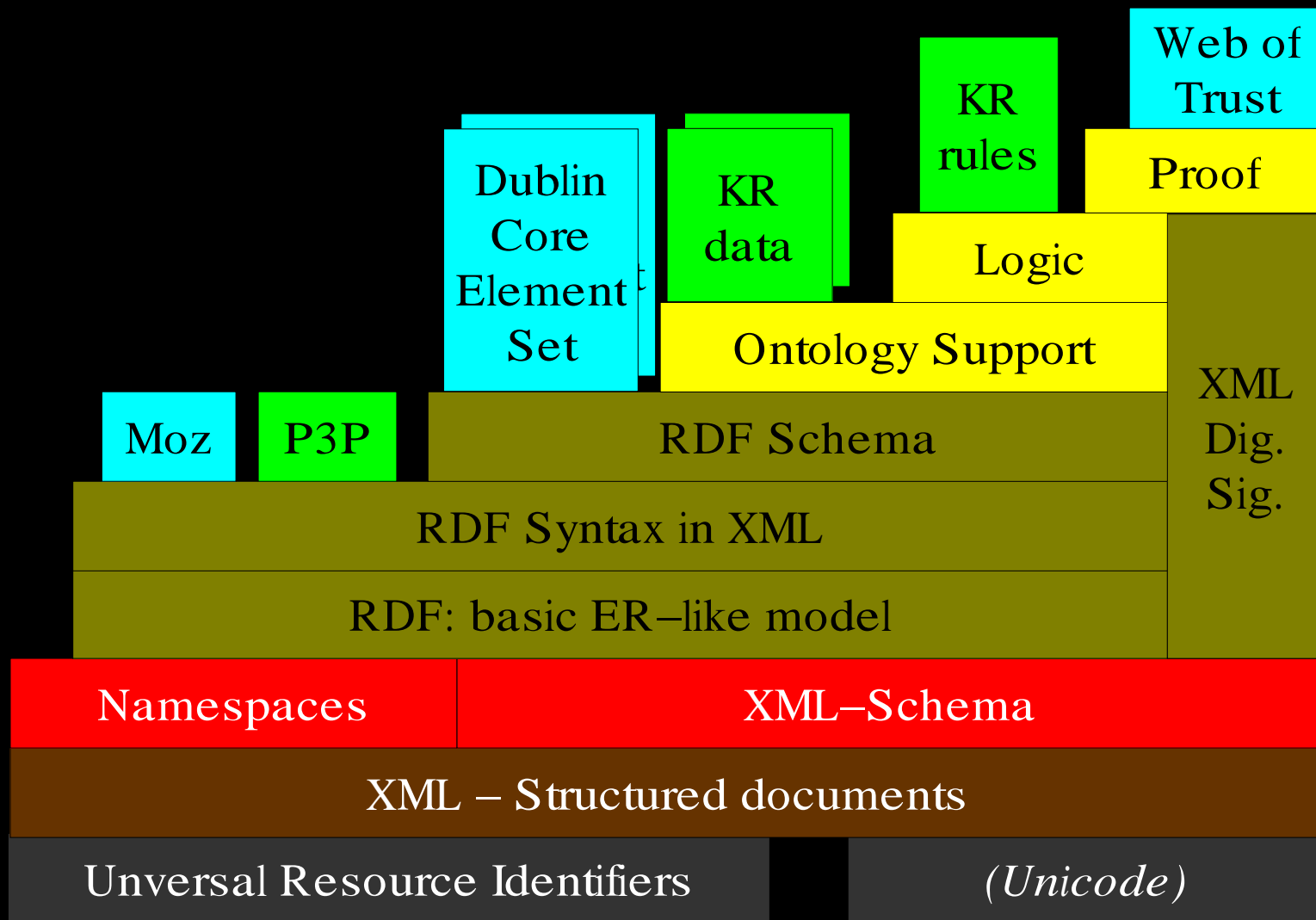# XML – a programmer's paradise?

- Representing data structures –
  gap between plain XML
  and OOP or OR models?

- Understanding <a1 a2="a3"><b1/></a1>?

- Exchanging data between applications
  without pain?

- Level of machine readability?

- Introduction of a new data model for every
  XML based standard?

What would you like to do with all that data out there in the web?

- Storage / Access

- Integration

- Manipulation

- Querying

- Inference / Proofing

- Do all these activities based on standards so that you spent the minimal amount of time with programming / porting etc.
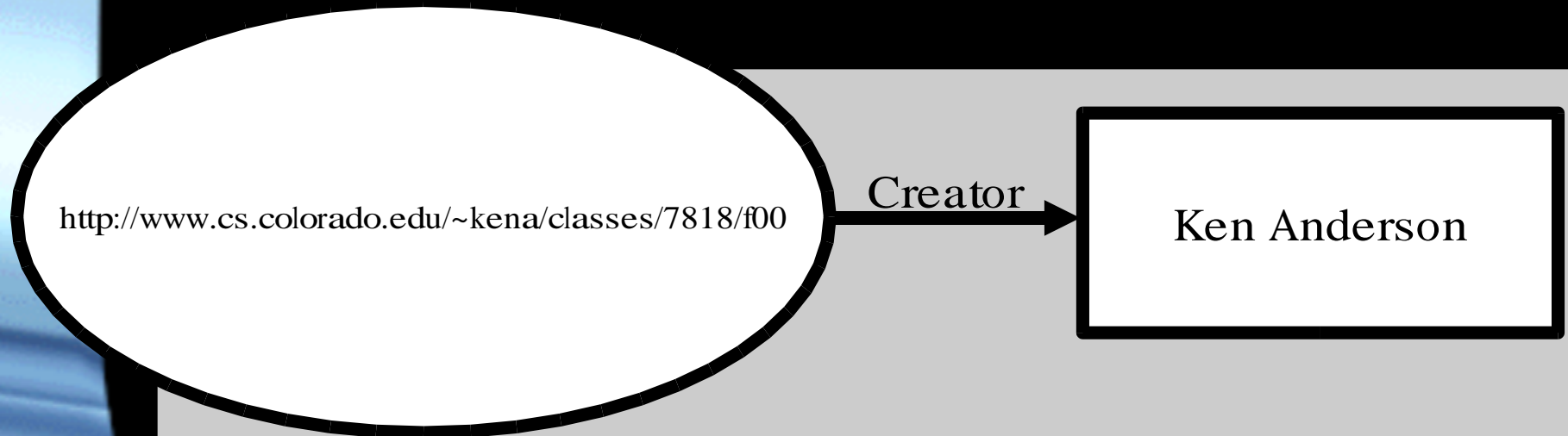
Tim Berners Lee's roadmap towards a "semantic web"

Web of Trust

KR rules

Dublin Core Element Set

KR data

Proof

Logic

Ontology Support

Moz P3P RDF Schema

XML Dig. Sig.

RDF Syntax in XML

RDF: basic ER–like model

Namespaces XML–Schema

XML – Structured documents

Unversal Resource Identifiers *(Unicode)*

References:
Tim Berners Lee: "Building the future" (this slide), http://www.w3.org/2000/Talks/0906–xmlweb–tbl/slide9–0.html
Tim Berners Lee: "Realising the Full Potential of the Web", http://www.w3.org/1998/02/Potential.html
Tim Berners Lee: "The future of the web", http://www.w3.org/1999/04/13–tbl.html
Homepage of Tim Berners Lee: http://www.w3.org/People/Berners–Lee/

Concepts of RDF: A first Example

http://www.cs.colorado.edu/~kena/classes/7818/f00 —— Creator ——▶ Ken Anderson

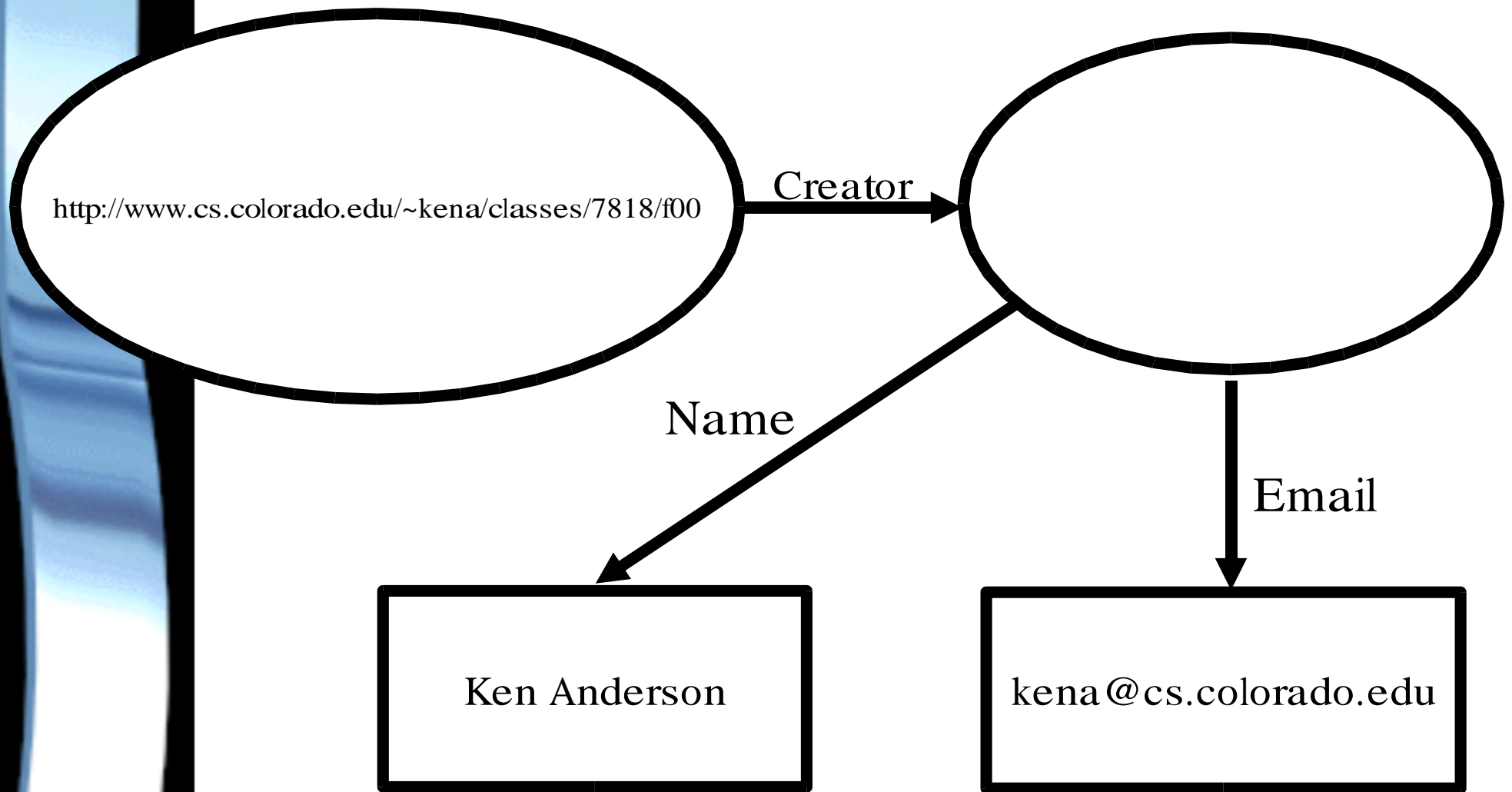| Subject (Resource)	Predicate (Property)	Object (Literal)
Sentence		

Creator(Resource("http://..."))="Ken Anderson"
HasCreator(Resource("http://..."), "Ken Anderson")
HasProperty(Resource("http://.../7818/f00/"),"Creator","KenAnderson")

<subject> HAS <predicate> <object>

Reference:
W3C Rec: "ResourceDescription Framework (RDF) Model and Syntax Specification", 22 Feb 1999,
http://www.w3.org/TR/1999/REC–rdf–syntax–19990222

Creator

Name

Email

http://www.cs.colorado.edu/~kena/classes/7818/f00

Ken Anderson

kena@cs.colorado.edu

!EXISTS x.
    HasProperty(Resource("http://.../f7818/f00"),"Creator",x)
    AND HasProperty(x,"Name","Ken Anderson")
    AND HasProperty(x,"Email","kena@cs.colorado.edu")

Semantic Web: Architectural Goals

- Scalability
  RDF: Scale KR up for worldwide use
  WWW: Scale hypertext up for worldwide use

- Everyone should be able to express anything

- No consistency guarantee

- No centralized approach

- Enable prooving without making it a requirement for the usefulness of the web

- But: You might express queries that are undecidable (no closed world, "A and not A")

Reference:
Tim Berners Lee: "What the Semantic Web can represent", 1998,
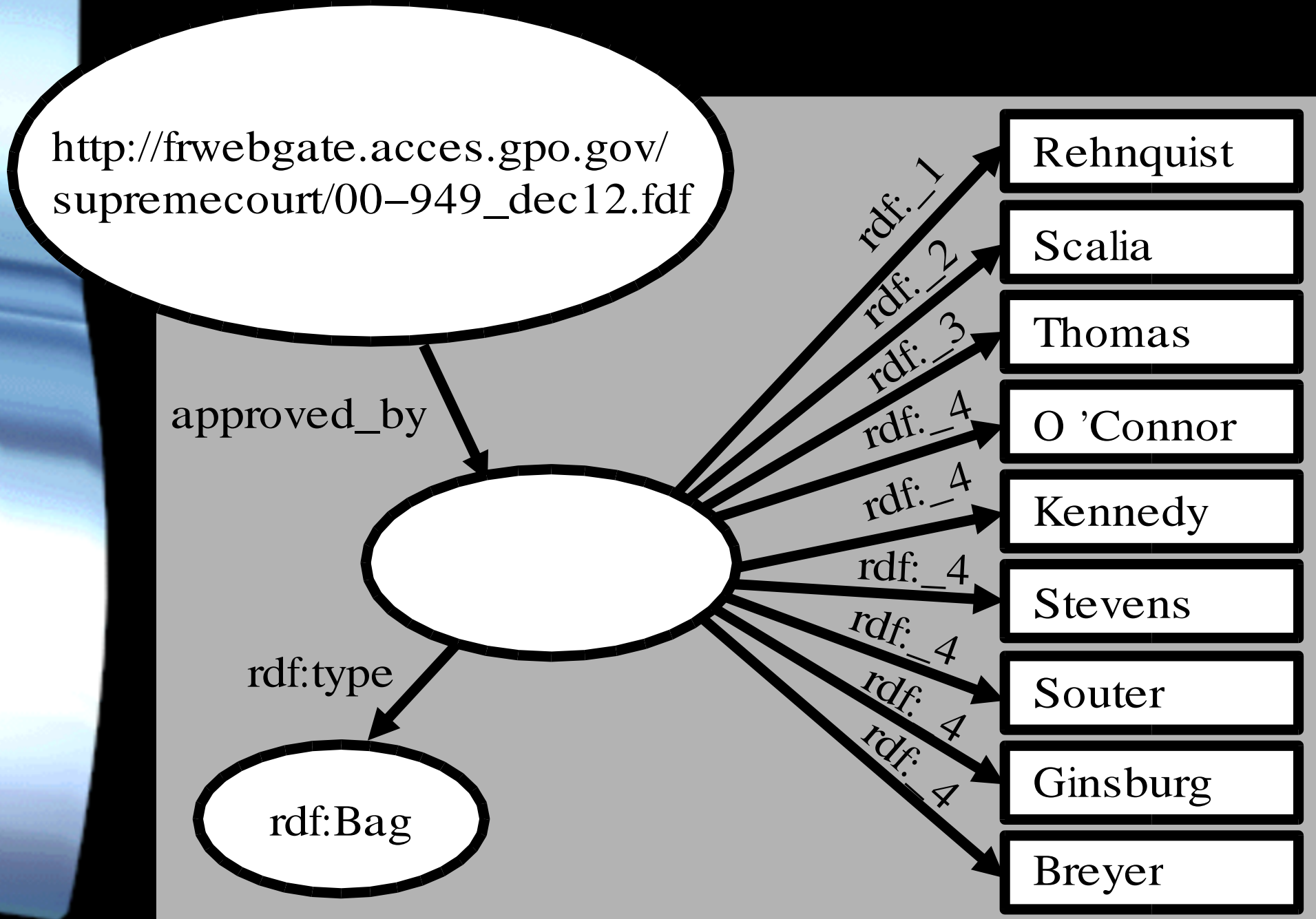http://www.w3.org/DesignIssues/RDFnot.html

How much semantic does our example contain thus far?

- Mapping to formulae in predicate calculus (not defined by standard)

- Using URLs allows for uniform access to locate resources: Resource("http://...")

- Doesn't "access" really mean "semantic"?

- The names of the properties are just arbitrary strings thus far: "Email" / "Name"

- Also, "Ken Anderson" is just a string.

- In RDF, properties (the relations itself) are first–class values, specified in RDF Schemas – we could add even more semantic declaratively

- Higher order statements possible

RDF containers: Bag, Sequence, Alternative

- Bag: An unordered list of resources or literals.

- Sequence: An ordered list of resources or literals.

- Alternative: A list of resources or literals that represent alternatives for the (single) value of a property.

- No Set.

- the rdf:type property is used to assign types specified in RDF Schema

- the containers are built–in types

- So why do we not just use the XML Schema mechanisms?

A rdf:Bag example

http://frwebgate.acces.gpo.gov/
supremecourt/00−949_dec12.fdf

approved_by

rdf:type

rdf:Bag

rdf:_1 → Rehnquist
rdf:_2 → Scalia
rdf:_3 → Thomas
rdf:_4 → O 'Connor
rdf:_4 → Kennedy
rdf:_4 → Stevens
rdf:_4 → Souter
rdf:_4 → Ginsburg
rdf:_4 → Breyer

What about this variation?

http://frwebgate.acces.gpo.gov/ supremecourt/00−949_dec12.fdf

approved_by

...

...

...

approved_by
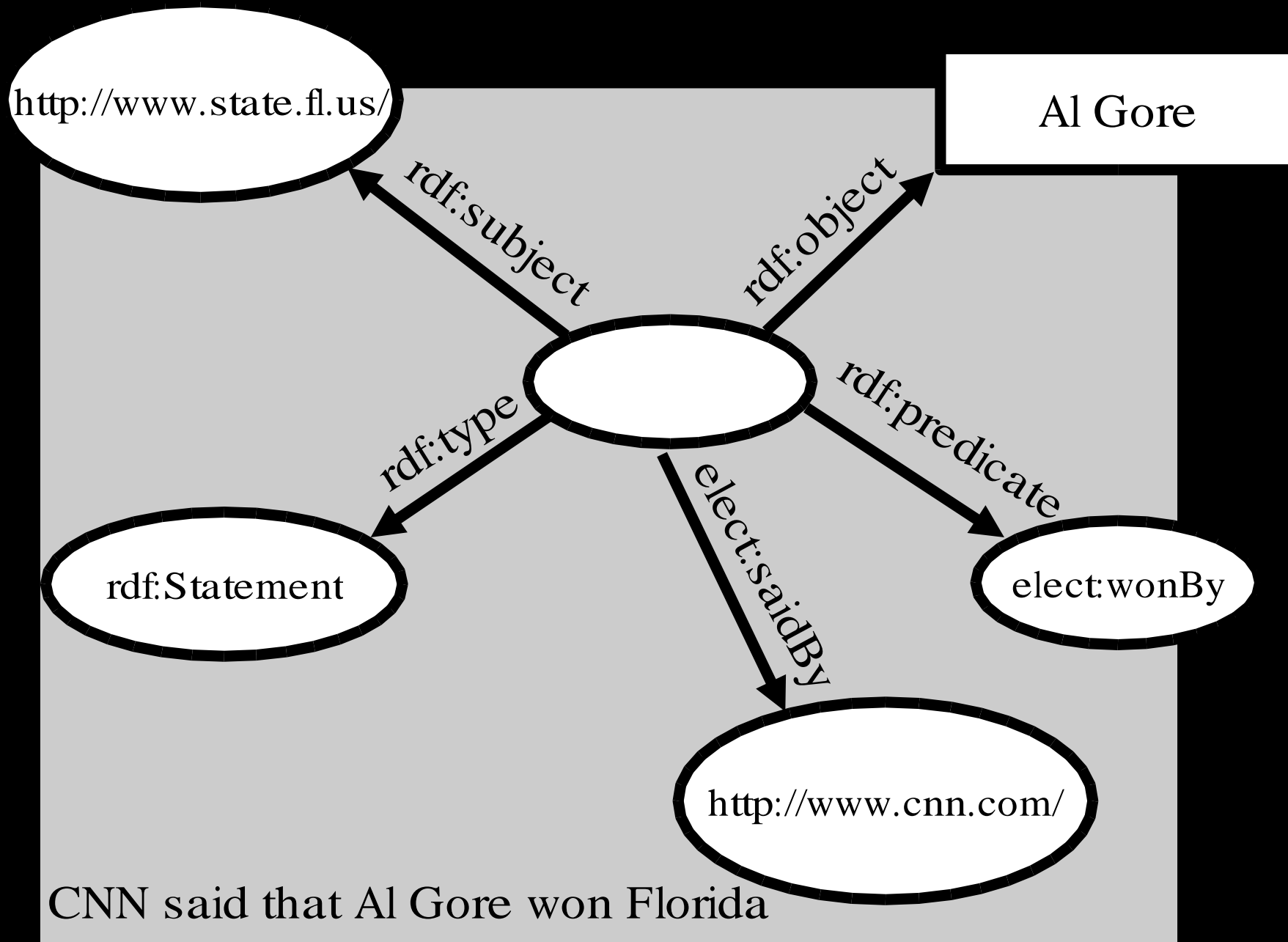
...

...

...

...

Rehnquist

Scalia

Thomas

O 'Connor

Kennedy

Stevens

Souter

Ginsburg

Breyer

Valid RDF, but Stevens, Souter Ginsburg and Breyer dissented!

A statement about a statement

http://www.state.fl.us/

Al Gore

rdf:subject

rdf:object

rdf:type

rdf:predicate

elect:saidBy

rdf:Statement

elect:wonBy

http://www.cnn.com/

CNN said that Al Gore won Florida

What about syntax?!

- Two Flavors:

  - Basic Serialization Syntax

  - Basic Abbreviated Syntax

- ```xml
  <?xml version="1.0"?>
  <RDF xmlns="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
      <Description about="http://www.cs.colorado.edu/~kena/">
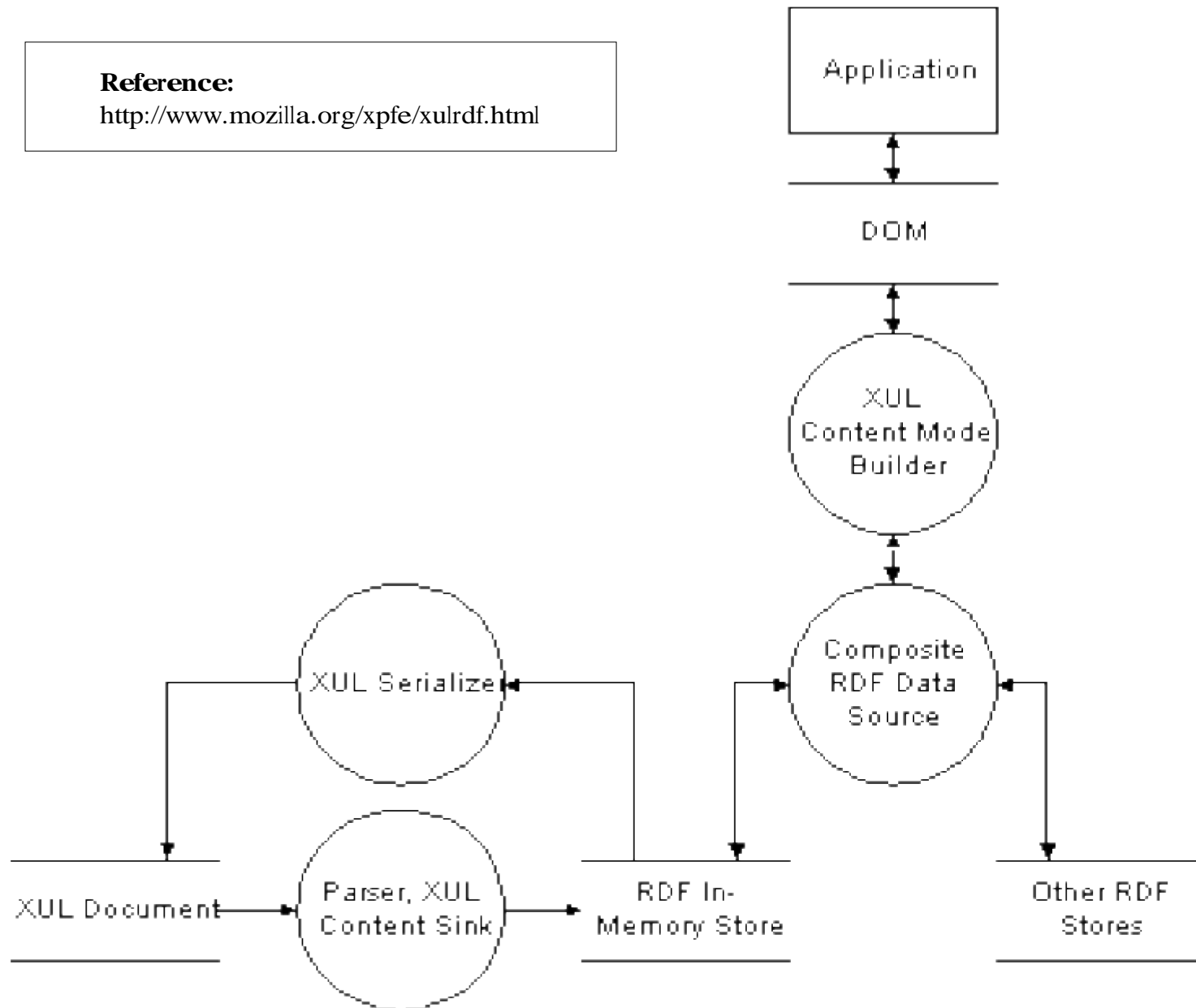   <Creator xmlns="http://www.jhenkel.de/webschema#">
            Ken Anderson
          </Creator>
      </Description>
  </RDF>
  ```

- This is the only example I give, since you will forget the syntax anyway. Look at the spec for reference

# RDF in Netscape

- Implementation of RDF is used as local database

- Can create and manimpulate RDF in memory using JavaScript interfaces

- good security model

- XUL can be used for rendering – there is a pattern language similar to XSLT called XUL template that maps RDF into XUL

- automatic synchronization etc.

# Mozilla RDF Architecture

Reference:
http://www.mozilla.org/xpfe/xulrdf.html

Application

DOM

XUL
Content Mode
Builder

XUL Serialize

Composite
RDF Data
Source

XUL Document

Parser, XUL
Content Sink

RDF In-
Memory Store

Other RDF
Stores

# Mozilla Pros and Cons

- Very flexible approach – UI can be completely engineered with scripting and declarations

- Performance is a problem

- RDF is probably not used in a way that gives much benefit for development

- XUL templates cannot do arbitrary transformations

- Very poor documentation
  (most serious problem)

- Quite a lot projects that use Mozilla as a starting point (Mozilla is a platform!)

# Useful Links

- "XUL Notes", http://www.mozilla.org/docs/xul/xulnotes/
  Check out the "Cheatsheets" – this stuff really works!

- "XUL Template Primer", http://www.mozilla.org/docs/xul/xulnotes/template–primer.html
  Pretty weird. None of the examples just works. Did not yet get fully behind it.

- "Resource Description Framework", http://www.mozilla.org/rdf/doc/

- "RDF–in–Mozilla FAQ", http://www.mozilla.org/rdf/doc/faq.html
  This is a useful page (up to date, the stuff on that page almost works).

- "RDF Model and Syntax Spec", "http://www.w3.org/TR/REC–rdf–syntax

- "Resource Description Framework", http://www.w3.org/RDF/
  Best and most comprehensive page I have seen thus far.

# Some Hints if you want to check out Mozilla

- If you read rdf files from a webserver, make sure that the server uses the correct mimetype

- Some of the examples caused security violations in Netscape 6. If nothing happens, check
  out the JavaScript console. Most times you will get the exact line where the error is located.
  Copy the piece of code where you think that the problem occurs right into google and
  search on it. Most times this gave me good links to mailinglists where people complained
  about the very same problem (and got an answer).