JENA: A Java API for Ontology Management

Hari Rajagopal
IBM Corporation
Agenda

- Background
- Intro to JENA
- Case study
- Tools and methods
- Questions
The State of the Web Today

- The web is more Syntactic (text based) than Semantic (context and relevance based)

- Goal is to make the content on the web machine understandable and accessible
The Syntactic Web Is...

- A hypermedia, a digital library
  - A library of documents called (web pages) interconnected by a hypermedia of links
- A database, an application platform
  - A common portal to applications accessible through web pages, and presenting their results as web pages
- A platform for multimedia
  - BBC Radio 4 anywhere in the world! Terminator 3 trailers!
- A place where computers do the presentation (easy) and people do the linking and interpreting (hard).

Why not get computers to do more of the hard work?

[Goble 03]
Semantic Web

“The Semantic Web will augment the existing human-readable Web with structured data that's easy for software to process”
Semantic Web - layers

- Resource Description Framework (RDF): lets you assert facts
  e.g. ‘Akbash is-a breed of dog’
- RDF Schema: lets you describe vocabularies and use them to describe things
  e.g. person X is a LivingPerson.
- Web Ontology Language (OWL): lets you describe relationships between vocabularies
  e.g. persons in schema A are the same thing as users in schema B.
Access to Web Resources

- HTTP
- FTP
- SMTP
- FILE (?)

One thing in common – ‘They all need to refer to a unique ID for a resource’
Resources on the Web

- A resource can be identified by a URI

- Some examples
  - www.synergy.com/resources/domain/#a
  - www.apex.com/alpha/#list
  - And so on............
URL - Definition

- A URI + Protocol = URL

- While a URL may be used for a resource that you can browse to, a URI can be used for anything (literally)
- Implications – we can model far more than a web resource
RDF Basics

RDF (Resource Description Framework) is a language for representing information about resources

- [A resource is any item of info that has a URI]

- W3C standard
Resource?

Which leads to the immediate response –

- A resource is any entity that one wishes to refer to, on the web a resource could be a web page, a link, a particular user identity, .....
In the RDF world it is a bit more stringent

- A resource is either a
  - Literal (any XML Schema allowed type)
  - A URI itself

- A resource has
  - A URI
  - Properties that qualify it
The properties mentioned here have values that are literals

Statements can be made about the resources and their properties thus linking them together

These form what are known as RDF graphs

Resource (Continued)
RDF Representation

- Directed graph

- Reasons for adoption:
  - Extremely easy to follow
  - Clear representation (in space) of the Subject-predicate-Object triplet
RDF – Directed Graph

http://www.abc.com/#res

property#ID

http://www.abc.com/#value
Usability of This Notation

- Very easily understood

  ➢ BUT

- Not easily serialized or transported [this is meant to be used on the web among other things after all]

- So, RDF/XML is used
RDF Format

```
<rdf:RDF
    xmlns:j.0="http://www.functions.com/
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
  <rdf:Description rdf:about="http://www.ccs.example/dog#akbash">
    <j.0:FunctionalType>Flock guard</j.0:FunctionalType>
  </rdf:Description>
</rdf:RDF>
```
RDF and XML Schemas

- Define a schema using XSD

- Reuse the XML data-types in defining the RDF description

- The example in the previous page could be rewritten as follows
Use of an XML Schema

```
<rdf:RDF
    xmlns:j0="http://www.functions.com/
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
  <rdf:Description rdf:about="http://www.ccs.example/dog#akbash">
    <j0:FunctionalType>Flock guard</j0:FunctionalType>
  </rdf:Description>
</rdf:RDF>
```
RDF Statements

- http://www.example.org/index.html has a creator whose value is John Smith
- http://www.example.org/index.html has a creation-date whose value is August 16, 1999
- http://www.example.org/index.html has a language whose value is English
Corresponding Graphs

http://www.example.org/index.html

http://www.example.org/terms/creation-date

August 16, 1999

http://purl.org/dc/elements/1.1/creator

http://purl.org/dc/elements/1.1/language

http://www.example.org/staffid/85740

en
RDF Validation and Visualization

- [http://www.w3.org/RDF/Validator](http://www.w3.org/RDF/Validator)

- An online service for validating your RDF schemas and documents and graphing them

- Uses the ‘ARP’ RDF parser
<?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/1.1/"/>
    <rdf:Description rdf:about="http://www.w3.org/">
    </rdf:Description>
</rdf:RDF>
Node-arc Representation

Graph of the data model

http://www.w3.org/

http://purl.org/dc/elements/1.1/title

World Wide Web Consortium
Model

- A collection of these RDF graphs gives us a set of statements that can be queried and manipulated
- In effect:
  - A knowledge representation

OR

'Model'
Representation of a Statement

- The resource 'chaise-lounge' has a property 'is-made-of' with a value of 'rattan'

http://www.furniture.com/products/#chaise_lounge

is-made-of

rattan
Point to note:

- Just like resources, properties are namespace and URI qualified too
- Values can be just that – ‘literals’

- However, if a value needs to be further qualified by properties and values, then it too must:
  - Be a resource (URI and all)
http://www.furniture.com/products/#chaise_lounge

is-made-of

http://www.furniture.com/#wicker

reed-type
rattan

imported-from

Indonesia
RDF Models

- Each model is made up of a number of statements
- A statement asserts (not the same as a C++ or JAVA Assert, there is no boolean outcome here) a **fact** about a resource
- A statement basically links a Subject and an Object together using a Predicate
Ontology

- Textbook definition of an ‘Ontology’ = a specification of a concept’

- To a developer
  - A system of declaring and defining meaningful relationships (aka rules) between domain objects
  - In the RDF-S world, these domain objects are resources
Ontology Models

- An ontology model is an extension of the RDF model that provides extra capabilities for handling ontology data.

- Allows customization as to:
  - Storage (disk vs memory)
  - Reasoning vs non-reasoning
  - Different Ontology formats used
Today..

- Search for ‘offenhauser’ and AMC and you end up with ...

  "Search for 'offenhauser' and AMC and you end up with ..."
What We Really Wanted ...

- What the torque/rpm and hp/rpm curves for the intake on a stock motor were supposed to be

How did we get so far off track?
- No contextual search, simply a text keyword lookup

Even with Google???
- Getting better every day
Points to Note

- We did not come up with any ‘inferences’
- Everything was a direct conclusion
- So, how would we achieve that?
Web “Schema” Languages

- Existing Web languages extended to facilitate content description
  - XML → XML Schema (XMLS)
  - RDF → RDF Schema (RDFS)

- XMLS not an ontology language
  - Changes format of DTDs (document schemas) to be XML
  - Adds an extensible type hierarchy
    - Integers, Strings, etc.
    - Can define sub-types, e.g., positive integers
- RDFS is recognisable as an ontology language
- It has
  - Classes and Properties (and sub-classes thereof)
  - Ranges and domains for the above
- OWL allows the characteristics of resources' properties to be expressed
- So, inverse properties, like-as and similar-to are definable using OWL
- This gives more relationship status to a RDF schema based model
OWL Versions

- OWL full
- OWL DL (Description Logic)
- OWL Lite

OWL DL is the one used in our examples
JENA

- Open source
  - [http://jena.sourceforge.net](http://jena.sourceforge.net)
- Based on development done at the HP Labs Semantic web development effort and released into the open source community
What Features Are Offered?

- A RDF API
- Reading and writing RDF in RDF/XML, N3 and N-Triples
- An OWL API
- In-memory and persistent storage
- RDQL – a query language for RDF
Using JENA

- Get the package (latest version is 2.2) from http://jena.sourceforge.net/downloads.html

- The unzipped folder structure is shown here

- Eclipse is our IDE of choice
Using JENA *(Continued)*

- All the examples to follow assume some knowledge of Eclipse and JDK 1.4.xx
- Download the JENA 2.3 release
- Create a project within the Eclipse workspace with the jar dependencies
Jena has object classes to represent graphs, resources, properties and literals

Oddly enough, these classes are named:

- Resource
- Property

and

- literal
The entire graph under consideration is however represented by the class ‘Model’

In JENA, Models can be
- Created
- Loaded
- Saved
- Operated upon and queried using plain JAVA
Querying a Model

- RDQL
  - Very SQL like, straightforward
Case Studies

Once defined and refined, it may be used for diverse purposes:

A designer creates an ontology for furnishings, 2 uses that come to mind:

- A web-enabled interior design application
- A manufacturer can link the ontology to a supplier list to determine best prices for rare fabrics and hence lower his per unit cost during manufacture
Architecture

- Application Logic
- JENA based access
- Domain Model

1: Create Model

Protege

DB store
JENA and OWL

- JENA has a special model class for this purpose
  - OntModel
- Using this JENA can derive meaningful relationships that the model does NOT express **directly** !!!

- This is the true power of an OWL model
JENA and OWL  *(Continued)*

- Using a Reasoner and a previously defined (or imported) model, JENA can infer relationships

- Swappable inference engines
  - OwlReasoner (micro, mini and regular)
  - Transitive reasoner
  - RDFS reasoner
Reasoner

Ontology model

Jena Graph interface

Reasoner

Jena Graph interface

Base RDF graph
Walk thru of implementation within Eclipse
Practical Applications

- Create a model (OWL ?)
- Persist model
- Import model
- Process and perform operations on model
Model Creation

- Typically done using a standalone editor
  - RDF-Editor
  - Protégé

- Protégé by itself uses a proprietary format
  - Plugins available for format as well as persistence mechanisms
Protégé Demo

- Creating an OWL model
Case Study – SofaWorld.com

- An online furnishings retailer wishes to zoom in on customer requirements and tailor their offerings to the customers tastes
- So far the model has been to use a DB backed catalog of offerings
- Matching is done on a text-tag basis
- Preferences of returning shoppers have been stored as text
OWL versus RDF

- When would you use one over the other?
Use RDF When...

- You want a simple interoperable language
- Assertion based model
- Simple API to query, extract and use data represented
Use OWL When...

- You want to share the model

- The usage reflects some amount of inferencing
Further Topics…

- Pluggable reasoners
- Performance issues
- Applicability and persistence
Questions?
Resources

- The JENA toolkit
  - http://jena.sourceforge.net

- Protégé Ontology editor
  - http://protege.stanford.edu/

- Infer->Ed RDF editor
  - http://www.intellidimension.com

- Racer inference engine
  - http://www.racer-systems.com
References

- **Practical RDF**

- **Introduction to JENA**

- **The JENA Ontology API**

- **RDF primer** [http://www.w3.org/TR/rdf-primer/](http://www.w3.org/TR/rdf-primer/)
Further Info / Questions

- My email address:
  - harir@us.ibm.com