

Semantic Web Tools

Based on the slides by Dipsy Kapoor

Motivation



Semantic Web

- “The **Semantic Web** provides a common framework that allows **data to be shared and reused** across application, enterprise, and community boundaries” - <http://www.w3.org/2001/sw/>
- RDF & OWL – w3c standards for encoding data and its semantics
- There isn't enough semantic web content on web

Outline

- Describing resources using RDF/N3
- Semantic Web Tools
 - Piggy Bank
 - Solvent
 - Jena
 - ARQ

RDF

- RDF stands for Resource Description Framework
- RDF is a framework for describing resources on the web
- RDF provides a model for data, and a syntax so that **independent parties can exchange and use it**
- RDF is designed to be read and understood by computers
- RDF is **not designed for being displayed to people**
- RDF is **written in XML**
- RDF is a W3C Recommendation

From w3school.com

See good tutorials at <http://www.w3schools.com/rdf/>

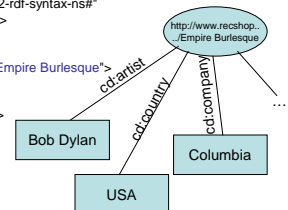
RDF example

Title	Artist	Country	Company	Price	Year
Empire burlesque	Bob Dylan	USA	Columbia	10.90	1985
Hide your heart	Bonnie Tyler	UK	CBS Records	9.90	1980
...					

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

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:cd="http://www.recshop.fake/cd#">
```

```
<rdf:Description
  rdf:about="http://www.recshop.fake/cd/Empire_Burlesque">
  <cd:artist>Bob Dylan</cd:artist>
  <cd:country>USA</cd:country>
  <cd:company>Columbia</cd:company>
  <cd:price>10.90</cd:price>
  <cd:year>1985</cd:year>
</rdf:Description>
```



The Dublin Core

- A set of predefined properties for describing documents in RDF form

```
<?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.w3schools.com">
    <dc:title>D-Lib Program</dc:title>
    <dc:description>W3Schools - Free tutorials</dc:description>
    <dc:publisher>Refsnes Data as</dc:publisher>
    <dc:date>1999-09-01</dc:date>
    <dc:type>Web Development</dc:type>
    <dc:format>text/html</dc:format>
    <dc:language>en</dc:language>
  </rdf:Description>
</rdf:RDF>
```

N3 (Notation 3)

- This is a language which is a compact and readable alternative to RDF's XML syntax

```
@prefix rdf <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@prefix cd <http://www.recshop.fake/cd/>

<http://www.recshop.fake/cd/Empire Burlesque>
  cd:artist Bob Dylan;
  cd:country USA;
  cd:company Columbia;
  cd:price 10.90;
  cd:year 1985.
<http://www.recshop.fake/cd/Hide your heart>
  cd:artist Bonnie Tyler;
  cd:country UK;
  cd:company CBS Records;
  ....
```

<http://www.w3.org/DesignIssues/Notation3.html> (Berners-Lee)

Outline

- Describing resources using RDF/N3
- [Semantic Web Tools](#)
 - Piggy Bank/Solvent
 - Jena/ARQ

Piggy Bank



- “Semantic Web Browser” by SIMILE project at MIT
- Available as a Firefox plug-in at <http://simile.mit.edu/piggy-bank/>
- Can ingest RDF data directly from websites that contain meta information, and can execute screen-scrapers to extract content from normal websites and then convert that data to RDF

Piggy Bank

- Features of the browser:
 - Users can find, collect, annotate RDF
 - Search and faceted browse of local RDF
 - Export data in RDF/N3

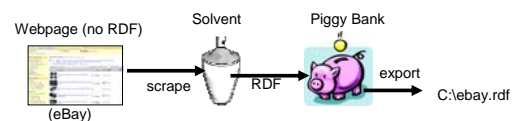
But this tool is **REALLY USELESS** when websites have no RDF describing objects in them!

Solvent

<http://simile.mit.edu/wiki/Solvent>



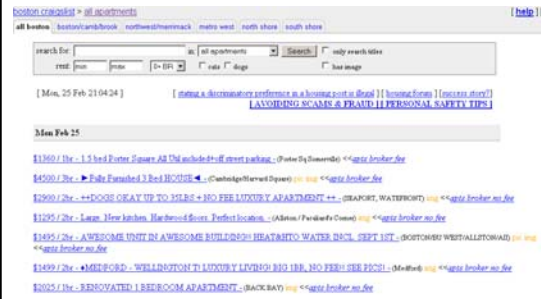
- Screen-Scraper (javascript-based) developed by the SIMILE project
- An extension of Piggy Bank
- Extracts data and exports it as RDF/n3



Piggy Bank and Solvent Demo

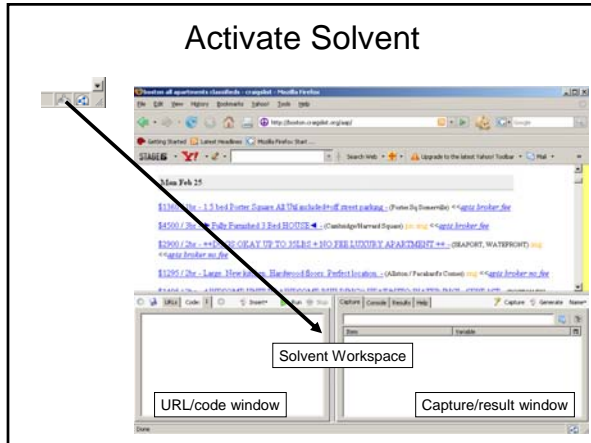
Scrape data from <http://boston.craigslist.org/aap/> and export to RDF

The data

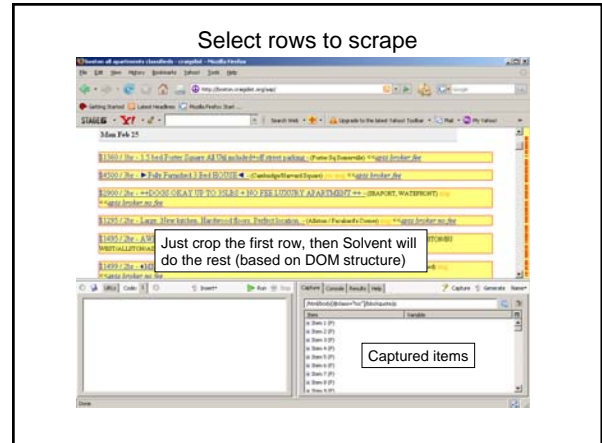


Objective: turn this data to RDF file

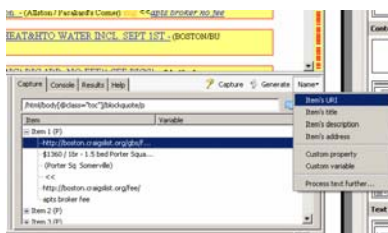
Activate Solvent



Select rows to scrape

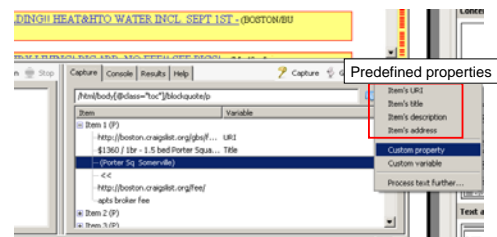


Name each extracted attribute



Every item must have URI in URI format! (if not, Solvent will not scrape anything)

Custom properties



You can define custom properties. Remember the name of a property should be in URI form e.g. <http://boston.craigslist.org/aap/area>

Jena

- A Semantic Web Framework that includes:
 - A RDF API
 - Reading and writing RDF in RDF/XML, N3 and N-Triples
 - An OWL API
 - In-memory and persistent storage
 - SPARQL query engine
- Documentation available at <http://jena.sourceforge.net/documentation.html>

Using Jena API – An Example

```
import com.hp.hpl.jena.query.*;
import com.hp.hpl.jena.rdf.model.*;

//Create a default model and add statements from a file
into it
Model model = ModelFactory.createDefaultModel();
InputStream in = new FileInputStream(new File(filename));
model.read(in,null);

//Now can add statements to the model, can query the model
etc

//Write the model to a PrintStream, in say N3 notation
model.write(System.out, "N3");
```

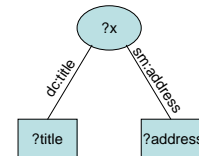
ARQ

- ARQ is Jena's implementation of SPARQL.
- ARQ documentation is available at: <http://jena.sourceforge.net/ARQ/documentation.html>
- Can write queries using the Jena Java API or can write them in text files and execute them using ARQ's command line utility.

SPARQL Example

```
PREFIX SM: <http://simile.mit.edu/2005/05/ontologies/location#>
PREFIX DC: <http://purl.org/dc/elements/1.1/>
```

```
SELECT DISTINCT ?title ?address
FROM <sources/starbucks.n3>
WHERE {
  ?x DC:title ?title .
  ?x SM:address ?address .
}
```



SPARQL Example

```
PREFIX SM: <http://simile.mit.edu/2005/05/ontologies/location#>
PREFIX DC: <http://purl.org/dc/elements/1.1/>
```

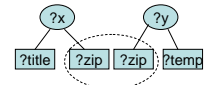
```
SELECT DISTINCT ?title ?address
FROM <sources/starbucks.n3>
WHERE {
  ?x DC:title ?title .
  ?x SM:address ?address .
  FILTER (regex(?address, "Figueroa", "i")) .
}
```

regex is a X-Path function. Other functions available at <http://jena.sourceforge.net/ARQ/library.html>

Querying multiple sources, Using simple JOIN

```
PREFIX SM: <http://simile.mit.edu/2005/05/ontologies/location#>
PREFIX DC: <http://purl.org/dc/elements/1.1/>
PREFIX fn: <http://www.w3.org/2005/xpath-functions#>
```

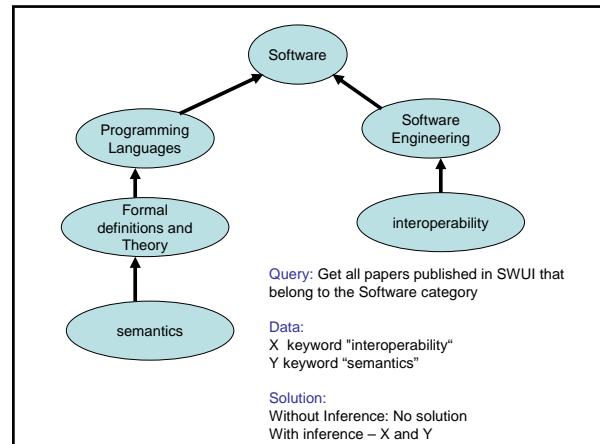
```
SELECT DISTINCT ?title ?zip ?temperature
FROM NAMED <sources/starbucks.n3>
FROM NAMED <sources/weather.n3>
WHERE {
  GRAPH <sources/starbucks.n3> {
    ?x DC:title ?title .
    ?x DC:zipcode ?zip .
  } .
  GRAPH <sources/weather.n3> {
    ?y DC:zipcode ?zip .
    ?y DC:temperature ?temperature .
  } .
}
```



More examples (might be useful for your hw): <http://www.ibm.com/developerworks/xml/library/i-sparql/>

Example of Inference

- <http://swui.semanticweb.org/swui.rdf>
 - List of publications accepted at the SWUI workshop , giving its title, author and keywords
- <http://139.91.183.30:9090/RDF/VRP/Examples/ACM-CSS.rdf>
 - ACM computer Classification System



Using Inference

```
import com.hp.hpl.jena.query.*;
import com.hp.hpl.jena.rdf.model.*;

Model model = ModelFactory.createDefaultModel();
model.read("http://swui.semanticweb.org/swui.rdf");
model.read("http://139.91.183.30:9090/RDF/VRP/Examples/ACM-CSS.rdf");

InfModel inf = ModelFactory.createRDFSModel(model);
//RDFS Model includes RDFS entailments like subclassOf reasoning

Query query = QueryFactory.create(queryString);
QueryExecution qe = QueryExecutionFactory.create(query, inf);
ResultSet results = qe.execSelect();
ResultSetFormatter.out(System.out, results, query);
```

Links

- Piggy Bank – <http://simile.mit.edu/piggy-bank/>
- Solvent - <http://simile.mit.edu/solvent/>
- Jena Documentation - <http://jena.sourceforge.net/documentation.html>
- ARQ Documentation - <http://jena.sourceforge.net/ARQ/documentation.html>
- Article by IBM on ARQ - <http://www-128.ibm.com/developerworks/xml/library/j-sparql/>

Extra slide: class project

- Just come see TAs; we can help refine your idea for your class project based on our experiences!
 - Anon: Mon, Wed (1-2 pm.)
 - Rattapoom: Fri (10.30-12.30)
- Before coming to see TAs you should:
 - Have your problem statement ready (what problem you want to solve). Make sure that you plan to use what you have learned in this class to solve the problem.
 - Speculate what possible sub problems you may encounter. Know roughly how to solve them.
 - Know what data sources you plan to use. What output you expect to return.
- If you did just OK for the HW3, perhaps you need to think a little bit more to make your idea more interesting.
- Prof. Knoblock has taught this class for several years. He saw tons of good projects – so be creative!
- If you plan to do project on research track (i.e. build a new information integration tool). We do highly recommend to see Prof. or TAs early.