SPARQL + RDF
Based on:
Prof. Benny Kimelfeld’s lecture notes
And Lee Feigenbaum’s "SPARQL By Example" Tutorial

The Semantic Web
• Vision: Web data will entail semantics in a manner that is understood (and processed, and linked) automatically by computers
• Data providers should collaborate: properly publish their data and link it to existing data

Some Freely Available RDF Repositories
• DBPedia (~1.2b triples)
  • "A crowd-sourced community effort to extract structured information from Wikipedia and make this information available on the Web"
• Freebase (~340m triples)
  • "A community-curated DB of well-known people, places, and things"
• DBLP (~15m triples)
  • Computer science bibliography
• WordNet (~3m triples)
  • English lexical db: synonyms, antonyms, POS, ...
• GeoNames (~14m triples)
  • "Contains all countries, continents over eight million placenames"
• Yago (~120m triples)
  • Information from Wikipedia, Wordnet, GeoNames

RDF Graph - recap
• An RDF graph is a finite set of triplets
• Given two sets:
  • $U$ is a set of URIs (unique resource identifier)
  • $L$ is a set of literals (strings, integers, etc.)
A triplet is in the form $(U \times U \times (U \cup L))$

RDF Example from DBPedia
Structure of a SPARQL Query

A SPARQL query comprises, in order:

- Prefix declarations, for abbreviating URIs
- Dataset definition, stating what RDF graph(s) are being queried
- A result clause, identifying what information to return from the query
- The query pattern, specifying what to query for in the underlying dataset
- Query modifiers, slicing, ordering, and otherwise rearranging query results

Endpoints

- http://yasgui.org/
- http://dbpedia.org/sparql

Theory In Practice

Yup, OK, I think I got how the model works... But what about the ACTUAL QUERY?!!

Maccabi Tel Aviv & NBA Players

Basics
Projection

\[ \text{SELECT } \pi_{x_1, \ldots, x_k}(P_1(G)) \]

Maccabi Tel Aviv players

\begin{verbatim}
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dbp: <http://dbpedia.org/property/>
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>

SELECT DISTINCT ?player {
  ?player dbp:team <http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C.>.
}
\end{verbatim}

Maccabi's players and their height

Given

<table>
<thead>
<tr>
<th>Player URI</th>
<th>Predicate</th>
<th>Maccabi URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>player1:dis_team</td>
<td><a href="http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C">http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C</a></td>
<td></td>
</tr>
<tr>
<td>player2:dis_team</td>
<td><a href="http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C">http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C</a></td>
<td></td>
</tr>
</tbody>
</table>

Player URI | Predicate | Height |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>player1:dis_height</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>player2:dis_height</td>
<td>2.10</td>
<td></td>
</tr>
</tbody>
</table>

Retrieve the player's URI and height

Maccabi’s players and their height

<table>
<thead>
<tr>
<th>Player URI</th>
<th>Predicate</th>
<th>Maccabi URI</th>
</tr>
</thead>
<tbody>
<tr>
<td>player1:dis_team</td>
<td><a href="http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C">http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C</a></td>
<td></td>
</tr>
<tr>
<td>player2:dis_team</td>
<td><a href="http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C">http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C</a></td>
<td></td>
</tr>
</tbody>
</table>

Player URI | Predicate | Height |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>player1:dis_height</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>player2:dis_height</td>
<td>2.10</td>
<td></td>
</tr>
</tbody>
</table>

Left Outer Join

\[ P_1 \text{ OPTIONAL } \{ P_2 \} = P_1(G) \bowtie P_2(G) \]
Maccabi’s players height and nationality

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dbp: <http://dbpedia.org/property/>
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>

SELECT DISTINCT ?player ?height ?nationality {
  ?player dbp:team <http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C.>.
  OPTIONAL {
  }
}
```

Selection

```
(P1 FILTER ( F )) = σ_{P}(G)
```

Maccabi’s players and their teams

For each Maccabi’s player, return the player and the team he played in

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dbp: <http://dbpedia.org/property/>
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>

SELECT DISTINCT ?player ?team {
  ?player dbp:team <http://dbpedia.org/resource/Maccabi_Tel_Aviv_B.C.>.
  ?player dbp:team ?team.
  ?team rdf:type { ?type }.
}
```

Subtraction

```
[P1] MINUS [P2] = P1(G) \cdot P2(G)
```

Remove all players with height > 2

```
SELECT DISTINCT ?player ?team {
  FILTER ( ?height > 2)
}
```
Remove all players with height > 2, and team in NBA

```
SELECT DISTINCT ?player ?team
{ ?player dbo:height ?h .
  FILTER (?h > 2) .
  ?team a dbo:Team .
  FILTER (sameAs(?team, ?player)) .
  FILTER (?player rdfs:label ?team) .
  LANG ?team.
}
```

Same as before...