Overview of the Semantic Web

Ralph R. Swick
World Wide Web Consortium (W3C)

11 January 2010
The current Web represents information using:
- natural language
- graphics, multimedia, page layout

Humans can process this easily:
- can deduce facts from partial information
- can create mental associations
- can deal with a variety of media
  - (see Web accessibility guidelines)
Example: airline reservation

- Your automatic airline reservation
  - knows about your preferences
  - builds up knowledge base using your past
  - can combine the local knowledge with remote services:
    - airline preferences
    - dietary requirements
    - calendaring
    - etc

- Interactions with other Web - or local - information
  - (M. Dertouzos: The Unfinished Revolution)
Example: data(base) integration

- Databases are very different in structure, in content
- Many applications require managing several databases
  - after company mergers
  - combination of administrative data for e-Government
  - biochemical, genetic, pharmaceutical research
  - etc.
- Most of these data are accessible from the Web - though not necessarily public yet
Example: “smart” portal

- Various types of “portals” are created (for a journal on-line, for a specific area of knowledge, for specific communities, etc)

- The portals may:
  - integrate many data sources
  - may have access to specialized domain knowledge

- Goal is to provide a better local access, search on the integrated data, reveal new relationships among the data
The structure of data integration

1. Map the various data onto an abstract data representation
   - make the data independent of its internal representation

2. Merge the resulting representations

3. Make queries on the whole
   - queries that could not have been done on the individual data sets
### A simplified bookstore data (dataset “A”)

<table>
<thead>
<tr>
<th>ID</th>
<th>Author</th>
<th>Title</th>
<th>Publisher</th>
<th>Year</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Home Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>id_xyz</td>
<td>Ghosh, Amitav</td>
<td><a href="http://www.amitavghosh.com">http://www.amitavghosh.com</a></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ID</th>
<th>Publ. Name</th>
<th>City</th>
</tr>
</thead>
<tbody>
<tr>
<td>id_qpr</td>
<td>Harper Collins</td>
<td>London</td>
</tr>
</tbody>
</table>
1st: expose your data as a set of relations

- **The Glass Palace**: a:title
- **2000**: a:year
- **London**: a:city
- **Harper Collins**: a:p_name
- **http://…/isbn/000651409X**: a:year
- **Ghosh, Amitav**: a:name
- **http://www.amitavghosh.com**: a:homepage
Some notes on exposing the data

- Relations form a graph
  - the nodes refer to the “real” data or contain some literal
  - how the graph is represented in machine is immaterial for now
- Data export does *not* necessarily mean physical conversion of the data
  - relations can be generated on-the-fly at query time
    - via SQL “bridges”
    - extracting from HTML pages
    - extracting data from Excel sheets
    - etc.
- One can export *part* of the data
Another bookstore data (dataset “F”)

<table>
<thead>
<tr>
<th>ID</th>
<th>Titre</th>
<th>Traducteur</th>
<th>Original</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>ID</th>
<th>Auteur</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISBN-0-00-651409-X</td>
<td>A12</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Nom</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ghosh, Amitav</td>
</tr>
<tr>
<td>Besse, Christianne</td>
</tr>
</tbody>
</table>
2nd: expose the second set of data

http://...isbn/000651409X

Le palais des mirroirs

http://...isbn/2020386682

Amitav Ghosh

Christiane Besse
3rd: start merging these data
3\textsuperscript{rd}: … Resource Identity

Same URI = Same Resources
3rd: identical resources connect
Start making queries…

- User of data “F” can now ask queries like:
  - “give me the title of the original”
- This information is not in the dataset “F”…
- …but can be retrieved by merging with dataset “A”!
However, more can be achieved...

- `a:author` and `f:auteur` should be the same relation
- But an automatic merge does not know that
- Add some extra information to the merged data:
  - `a:author` same as `f:auteur`
  - both identify a “Person”
    - a term that a community may have already defined:
    - a “Person” is uniquely identified by his/her name and, (e.g.), Web homepage
3rd revisited: use the extra knowledge
Permitting richer queries

- User of dataset “F” can now query:
  - “give me the home page of the original’s ‘auteur’”
- The information is not in datasets “F” or “A”…
- …but was made available by:
  - merging datasets “A” and datasets “F”
  - adding three extra statements as an extra “glue”
Merge with Wikipedia data
What did we accomplish?

- Combined different datasets that
  - are somewhere on the web
  - are of different formats (mysql, excel sheet, XHTML, etc)
  - have different names for relations
- We could combine the data because some URIs were identical (the ISBN in this case)
- We could add some additional information, using common terminologies that a community has produced
- Permitting new relations to be found and retrieved
Many Real Examples

- RPI Data-gov wiki
  
  http://data-gov.tw.rpi.edu/wiki/The_Data-gov_Wiki

- Contributed Case Studies and Use Cases
  
  http://www.w3.org/2001/sw/sweo/public/UseCases/

- Collection of more examples

- Semantic Web Challenge
Please make your raw data available

(internally)
Acknowledgements

- These slides are available on:
  http://www.w3.org/2010/Talks/0111-semweb-rrs

- Thanks to Ivan Herman,
  W3C Semantic Web Activity Lead

- And the Semantic Web community
  http://www.w3.org/2001/sw/
  http://www.w3.org/standards/semanticweb/
Future Work / Research

- Provenance
- Usage policy tagging
- Accountable Systems