Extracting data from RDBMS with R2RML
Task: Integrate data from relational DBMS with Linked Data

Approach: map from relational schema to semantic vocabulary with R2RML

Publishing: two alternatives –
- Translate SPARQL into SQL on the fly
- Batch transform data into RDF, index and provide SPARQL access in a triplestore
W3C RDB2RDF

The W3C made, last year, two recommendations for mapping between relational databases and RDF:

- **Direct mapping** directly exposes data as RDF
  - No allowance for vocabulary mapping
  - No allowance for interlinking (unless URIs used in relational data)
  - Not appropriate for this task
- **R2RML**, the RDB to RDF mapping language
  - Allows vocabulary mapping (subject, predicate and object maps with class options)
  - Allows interlinking – URIs can be constructed
R2RML: Relationship with ODBA

- **R2RML allows to syntactically encode OBDA systems where**
  - Target is an OWL or RDFS ontology
  - Source is a relational database (SQL)

- **Semantics/expressiveness is determined by the R2RML engine**
  - Various rewriting techniques
  - E.g. access to the ground data: UltraWrap
  - DL-Lite / OWL 2 QL: Quest / OnTop
MusicBrainz Next Gen Schema

Artist
As pre-NGS, but further attributes

Artist Credit
Allows joint credit

Release Group
Cf. ‘album’ versus:

Release • Track • Work

Medium • Track List • Recording

Source: https://wiki.musicbrainz.org/Next_Generation_Schema
Music Ontology

OWL ontology with following core concepts (classes) and relationships (properties):

Source: http://musicontology.com
R2RML Main Concepts
R2RML Class Mapping

Mapping tables to classes is ‘easy’:

```xml
lb:Artist a rr:TriplesMap ;
  rr:logicalTable [rr:tableName "artist"] ;
  rr:subjectMap
    [rr:class mo:MusicArtist ;
     rr:template
      "http://musicbrainz.org/artist/{gid}#_" ] ;
  rr:predicateObjectMap
    [rr:predicate mo:musicbrainz_guid ;
     rr:objectMap [rr:column "gid" ;
      rr:datatype xsd:string ]] .
```
R2RML Advanced Mapping

Mapping advanced relationships (SQL joins):

```
lb:artist_member a rr:TriplesMap ;
  rr:logicalTable [rr:sqlQuery
    """SELECT a1.gid, a2.gid AS band
      FROM artist a1
      INNER JOIN l_artist_artist ON a1.id =
        l_artist_artist.entity0
      INNER JOIN link ON l_artist_artist.link = link.id
      INNER JOIN link_type ON link_type.id
      INNER JOIN artist a2 ON l_artist_artist.entity1 = a2.id
      WHERE link_type.gid='5be4c609-9afa-4ea0-910b-12ffb71e3821'
      AND link.ended=FALSE""] ;
  rr:subjectMap lb:sm_artist ;
  rr:predicateObjectMap
    [rr:predicate mo:member_of ;
      rr:objectMap [rr:template "http://musicbrainz.org/artist/
        {band}#_" ;
        rr:termType rr:IRI]] .
```
R2RML Editor

Available in Information Workbench

Add Predicate Object Map(s).

SQL Preview for
musicbrainz:track

<table>
<thead>
<tr>
<th>id</th>
<th>recording</th>
<th>tracklist</th>
<th>position</th>
<th>name</th>
<th>artist_credit</th>
<th>length</th>
<th>edits_pending</th>
<th>last_updated</th>
<th>number</th>
</tr>
</thead>
<tbody>
<tr>
<td>13684154</td>
<td>5756782</td>
<td>1147981</td>
<td>1</td>
<td>3775130</td>
<td>12389</td>
<td>302000</td>
<td>0</td>
<td>2011-11-17 00:39:52</td>
<td>1</td>
</tr>
<tr>
<td>13684155</td>
<td>5756783</td>
<td>1147981</td>
<td>2</td>
<td>4119988</td>
<td>12389</td>
<td>247000</td>
<td>0</td>
<td>2011-11-17 00:39:52</td>
<td>2</td>
</tr>
<tr>
<td>13684156</td>
<td>5756784</td>
<td>1147981</td>
<td>3</td>
<td>1623030</td>
<td>12389</td>
<td>242000</td>
<td>0</td>
<td>2011-11-17 00:39:52</td>
<td>3</td>
</tr>
<tr>
<td>12758010</td>
<td>12564807</td>
<td>1100291</td>
<td>3</td>
<td>5261608</td>
<td>159756</td>
<td>241000</td>
<td>0</td>
<td>2011-07-13 11:42:22</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>4</td>
<td>7</td>
<td>3132883</td>
<td>4</td>
<td>254706</td>
<td>0</td>
<td>2011-05-16 18:08:20</td>
<td>7</td>
</tr>
</tbody>
</table>

Predicate: `http://purl.org/ontology/mo/public`  
Object type: Reference  
Parent Rule: node17yjiuq9dx28  
Join Condition:  
Child Column: recording  
Parent Column: id  

Add Another Predicate-Object Map
Data Integration in the Information Workbench: Data Provider Concept

Data providers support the **periodic extraction & integration** from external data sources into a **central repository**

- Lifting from arbitrary data formats to RDF (e.g., relational, XML, CSV)
- Parametrizable (e.g. connection information, refresh interval, ..)
- Built-in UI for instantiating providers
- Intuitive interfaces and APIs for writing own, custom providers

Examples:

- XML2RDF
- R2RML
- SOAP Provider
- REST Provider
- ScriptProvider

**Process Flow:**

1. **Connect to data source**
2. **Extract data from source**
3. **Convert data into RDF**
4. **Store RDF in repository**
Example: instantiating an RDF provider for the wine ontology located at http://www.w3.org/TR/owl-guide/wine.rdf.
Data Warehousing vs. Federation

**Data Warehousing**
- Data is copied from the source into the warehouse
- Query runs in the warehouse
- Supported in IWB using *data providers*

**Federation**
- Data remains in federated DB
- Query is pushed down to federated DB
- Supported in IWB using *SPARQL federation*
Enabling Data Integration: 
*Federation of Virtualized Data Sources*

*See also:* *FedX: Optimization Techniques for Federated Query Processing on Linked Data (ISWC2011)*
Linked Data Visualization
Motivation: Music!

Examples of machine-readable output:
Visualizations techniques are needed in order to transform the machine-readable data into this:

**Motivation: Music!**

The Red Hot Chili Peppers are a funk rock band based in Hollywood, California and were formed in 1983, in Los Angeles, California. The band currently consists of founding members Anthony Kiedis (vocals) and Michael "Flea" Balzary (bass) along with a longtime member Chad Smith (drums). Guitarist John Frusciante quietly left the band on good terms in late 2008. On January 2, 2010, Josh Klinghoffer was announced as Frusciante's replacement. [Read more about Red Hot Chili Peppers on Last.fm](http://musicbrainz.fluidops.net/).
Linked Data Visualization Techniques

Linked Data visualization techniques should provide graphical representations of the information within the LD datasets.

Visualization techniques should be selected accordingly to:

- **The type of data**: Specific types of data should be visualized in a certain way.

- **The purpose of the visualization**: Depending on the type of analysis/application to employ.
Overview of the Linked Data Visualization process

(Raw) RDF data: Instance data, taxonomies, ontologies, vocabularies.

Analytically extracted data: Subset of the data denominated region of interest (ROI), obtained via data extraction mechanisms, for example, SPARQL queries.

Visualization abstraction: It is obtained by applying visualization transformations to render the data into displayable information.

View: Final result. The visual mapping transformations obtain a graphic representation of the data using the selected visualization technique.

User interaction: The user interacts (click, zoom, etc.) with the visualization, which may trigger a new visualization process.

Linked Data Visualization Techniques (3)

Example of the Linked Data Visualization process

**SPARQL query:** *Retrieve number of releases per country of The Beatles*

```
SELECT ?country (COUNT(?release) AS ?releases)
WHERE {
  ?release a mo:Release ;
  mo:label ?label .
  ?label foaf:based_near ?country .}
GROUP BY ?country
ORDER BY DESC(?releases)
```

Formatting the names of the countries

?country_code2 := REPLACE(str(?country), "http://ontologi.es/place/", "", ";")
?country_code := REPLACE(?country_code2, "%", "", ";")

Selecting the visualization technique (input, output)

#widget : HeatMap | input = 'country_code' | output = {{'releases'}}
Linked Data Visualization Techniques (3)

Example of the Linked Data Visualization process

View
Challenges for 
Linked Data Visualization

**Enabling user interaction**
- Users must be able to *navigate* through the data by exploiting the connections between Linked Data resources
- The user might edit the underlying data to enrich it by:
  - Creating additional metadata
  - Highlighting or correcting errors
  - Validating data

**Supporting data reusability**
- The output (the plotted data or the visualization itself) might be encoded using standard ontologies and vocabularies

**Scalability**
- Linked Data visualization techniques should support the display of large amount of data in an efficient way
Challenges for Linked Open Data Visualization

Extracting data from different repositories

- A Linked Data set might be partitioned into several repositories
- The region of interest (ROI) might include data from different data sets, requiring the access to distributed repositories

Handling heterogeneous data

- The same data (concepts) might be modeled differently, for example, using different vocabularies
- Certain values might have different formats, for example, dates represented as DD-MM-YYYY, MM-DD-YYYY or just YYYY

Dealing with missing values

- Due to the semi-structuredness of Linked Data, some instances might have missing values for certain properties
## Classification of Visualization Techniques

<table>
<thead>
<tr>
<th>Task</th>
<th>Visualization techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comparison of attributes / values</td>
<td>• Bar/column and pie chart &lt;br&gt; • Line charts &lt;br&gt; • Histogram</td>
</tr>
<tr>
<td>Analysis of relationships and hierarchies</td>
<td>• Graph &lt;br&gt; • Arc diagram &lt;br&gt; • Matrix &lt;br&gt; • Node-link visualizations &lt;br&gt; • Space-filling techniques: Treemaps, icicles and sunburst, circle packing and rose diagrams</td>
</tr>
<tr>
<td>Analysis of temporal or geographical events</td>
<td>• Timeline &lt;br&gt; • Maps</td>
</tr>
<tr>
<td>Analysis of multi-dimensional data</td>
<td>• Parallel coordinates &lt;br&gt; • Radar/star chart &lt;br&gt; • Scatter plot</td>
</tr>
</tbody>
</table>
Comparison of Attributes / Values

**Bar/column chart**
Allows the comparison of values of different categories.

**Pie chart**
Useful for performing comparison of percentages or proportions.

**Line chart**
Allows visualizing data as a series of data points, where the measurement points (x-axis) are ordered.

**Histogram**
Graphical representation of the distribution of the data.
Analysis of Relationships and Hierarchies

Graph
The data entries are represented as nodes and the links as edges.

Arc diagram
The nodes are displayed in one dimension, and the arcs represent the connections.

Adjacency Matrix diagram
The nodes are displayed as rows and columns, and the links between the nodes are entries in the matrix.

Node-link visualizations
The data is organized in hierarchies.

Source of images: http://mbostock.github.io/protovis/
Analysis of Relationships and Hierarchies (2)

**Treemaps**
Subdivide area into rectangles.

**Icicles and sunburst**
Hierarchies are represented by adjacencies.

**Circle-packing**
Containment is used to represent the hierarchies.

**Rose diagrams**
Areas are equal angles and the data is represented by the extension of the area.

Source of images: http://mbostock.github.io/protovis/
Analysis of Temporal or Geographical Events

Timeline

Discrete data points in time

Continuous data in time

Source: http://musicbrainz.fluidops.net

Location maps
Display geo-points on a map

Source: Google Map API

Choropleth maps
Aggregate data by geographical area

Source: http://musicbrainz.fluidops.net

Dorling cartograms
Aggregate data and replace each area with a circle

Source: http://mbostock.github.io/protovis/
Analysis of Multidimensional Data

**Scatter plot**
Useful for performing comparison of percentages or proportions.

Source: http://mbostock.github.io/protovis/

**Radar/star chart**
Displays multivariate data as a two-dimensional chart. The axes correspond to the variables.

Source: http://mbostock.github.io/protovis/

**Parallel coordinates**
Allows visualizing high-dimensional data. Each vertical axis denotes a dimension, and a multidimensional point is represented as a polyline with vertices on the axes.

Source: http://mbostock.github.io/protovis/
Other Visualization Techniques

Text-based visualizations: tag clouds

DBpedia music genres

Phrase Net of Beatles Lyrics

Some of the previously presented techniques can be **combined** to produce more complex data visualizations
Linked Data Visualization Examples

Information Workbench: Browsing a music artist

(1) Search options

Keyword search

Fulltext Search
Artist Search

Artist

Search

Structured search

Artist Search by Country
Artist Search by Country and Group Member

Artist
Country

Search

(2) Search results

Artist

The Black Beatles
The Moog Beatles
The Beatles
Fab Beatles
The Beatles Revival
Beatles Chillout
The Tape-beatles
Tokyo Beatles
The Better Beatles
The Savage Young Beatles
Liverpool Lads a Beatles Tribute
The Beatles Revival Band
Jolly Joker and the Plastic Beatles of the Universe
Linked Data Visualization Examples

Information Workbench: Browsing a music artist

(3) Browsing the selected resource

The Beatles
Tags: classic rock, rock, british, 60s, pop

Summary

The Beatles were an iconic rock group from Liverpool, England. They are frequently cited as the most commercially successful and critically acclaimed band in modern history, with innovative music, a cultural impact that helped define the 1960s and an enormous influence on music that is still felt today. Currently, The Beatles are one of the two musical acts to sell more than 1 billion records, with only Elvis Presley having been able to achieve the same feat. Read more about The Beatles on Last.fm.

YouTube video

Paul McCartney
Ringo Starr
Pete Best
Stuart Sutcliffe
George Harrison
John Lennon
Linked Data Visualization Examples

Information Workbench: Visualization techniques

(3) Browsing the selected resource
Linked Data Visualization Examples

Information Workbench: User interaction
LD visualizations must support navigation through the data

UK Music Artists by Number of Releases

Black Sabbath  Coldplay  David Bowie
Deep Purple  Depeche Mode
Genesis  Georg Friedrich
Händel  Hawkwind  Iron Maiden
Jamiroquai  Jethro Tull  King Crimson  Led Zeppelin  London Symphony Orchestra
Marillion  Motörhead  Pet Shop Boys
Pink Floyd  Queen  Radiohead
The Beatles  The Cure
The Kinks  The Rolling Stones  The Who  Uriah Heep  XTC  Yes

Eric Clapton
Tags: classic rock, blues, rock, blues rock, guitar

Summary

Eric Clapton (born Eric Patrick Clapton in Ripley, Surrey, England on 30 March 1945), nicknamed "Slowhand", is a Grammy Award winning English composer, singer and guitarist who became one of the most respected artists of the rock era, winning three inductions into the rock and roll hall of fame. Clapton's musical style has changed during his career, but has remained rooted in the blues. Clapton is credited as an innovator in several phases of his career, which have included blues-rock with John Mayall & The Bluesbreakers and The Yardbirds (1963-1965). Read more about Eric Clapton on Last.fm.

Eric Clapton@last.fm

Similar artists

- B.B. King & Eric Clapton
- Derek and the Dominoes

Source: http://musicbrainz.fluidops.net/resource/Analytical5
Linked Data Visualization Examples

Information Workbench: SPARQL visualization
Top ten The Beatles releases according to the sum of track durations in minutes

SPARQL Query

```
SELECT ?release
  ((SUM(xsd:double(?duration/60000))) AS ?avg)
WHERE {
  <http://dbpedia.org/resource/The_Beatles>
GROUP BY ?release
ORDER BY DESC(?avg)
LIMIT 10
```
Linked Data Visualization Examples

Information Workbench: SPARQL visualization

Top ten The Beatles releases according to the sum of track durations in minutes

```
{{#widget: BarChart |
query = 'SELECT (COUNT(?Release) AS ?COUNT) ?label
WHERE {
  <http://musicbrainz.org/artist/8538e728-ca0b-4321-b7e5-
cff6565dd4c0#> foaf:made ?Release.
  ?Release dc:title ?label .}
GROUP BY ?label
ORDER BY DESC(?COUNT)
LIMIT 20'
| settings = 'Settings:barvertical_mb'
| asynch = 'true'
| input = 'label'
| output = 'COUNT'
| height = '300'}}
```
Linked Data Visualization Examples

Information Workbench: SPARQL visualization

Top ten The Beatles releases according to the sum of track durations in minutes

Other visualizations of the same result set ...

Line chart:

Pie chart:
Linked Data Visualization Examples

Information Workbench: Automated Widget Suggestion

Visualizing Analytical Queries

- Countries by Number of Beatles' Releases: (heat map)
- UK Music Artists by Number of Releases: (tag cloud)
- Top Ten Beatles Releases according to the sum of track durations in minutes: (bar chart) widget auto suggestion
- Number Of Bands per Band Size: (pie chart | widget auto suggestion)
- The number of releases per year for all German music artists: (timeplot | widget auto suggestion)

Table
Pivot view
Bar chart
Line chart
Pie chart

Select a suggested visualization
Visualization automatically built
Faceted Search and Analysis: Example

Information Workbench: Searching for artists in categories

Source: http://musicbrainz.fluidops.net/resource/mo:MusicArtist?view=pivot
Faceted Search

**Facets** = properties

Suitable for browsing **multi-dimensional taxonomies** based on the search attributes

Allows user to explore the data:

- User submits a (keyword) **query**
- Faceted system dynamically identifies the **relevant facets** (properties) for the given query and the **constraints** (values of those properties), and display the search results
- User may “**drill down**” by selecting specific constraints to the search results

Information can be accessed and ranked in multiple ways
Examples of Enterprise Linked Data Applications
London 2012

Spectacular close to London 2012 Games

24 August 2012 Last updated at 10:37

Headlines

'Strongest ever' GB finish third

Broadcasting revolution of the digital Olympics

TECHNOLOGY

Record visits to BBC Sport online

MODERN PENTATHLON

Murray wins Britain's 85th Olympic medal

AFRICA

The effect of Mo Farah's success

Who will be in Britain's class of 2016?

How can Rio follow London?

SAILING

Ainslie wants fifth Olympic gold medal

ATHLETICS

Holmes wants female recognition

School-clubs link 'key to legacy'

UK

Home towns welcome GB heroes home

Cuts expected to some GB sports

Lighthouse named after

Medal Table

<table>
<thead>
<tr>
<th>Rank</th>
<th>Country</th>
<th>Gold</th>
<th>Silver</th>
<th>Bronze</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>United States</td>
<td>46</td>
<td>29</td>
<td>29</td>
<td>104</td>
</tr>
<tr>
<td>2</td>
<td>China</td>
<td>36</td>
<td>27</td>
<td>23</td>
<td>86</td>
</tr>
<tr>
<td>3</td>
<td>Great Britain &amp; N. Ireland</td>
<td>29</td>
<td>17</td>
<td>19</td>
<td>65</td>
</tr>
</tbody>
</table>

Relive every Olympics moment

Catch up on every session of every event from the London 2012 Olympics in our video player

Olympics: Key info

London 2012 Results

Full medal table

Team GB medals roll of honour

In numbers: the 2012 Games

Lived London 2012 as it happened

Great Britain Golds in Video

Best moments of the Olympic Games

BBC Sport's pundits and commentators talk about their highlights of the London 2012 Olympic Games.

20 memories of the London 2012 Olympics

BBC Sport looks back at 20 of the most memorable moments of London 2012 - including Bolt, Ennis & Farah.
Open Sport Ontology
Dynamic Semantic Publishing with the BBC

Olympics 2012 requirements
- A lot of output... Page per Athlete [10,000+], Page per country [200+], Page per Discipline [400-500], Time coded, metadata annotated, on demand video, 58,000 hours of content
- Almost real time statistics and live event pages with too many web pages for too few journalists

Dynamic Semantic Publishing (DSP) architecture to automate content aggregation

Information Workbench for DSP
- Collaborative authoring and linking of unstructured and structured semantic data
- Ontology and instance data management
- DSP editorial workflows
- Automation of content creation and enrichment
Information Workbench DSP Architecture

Journalist, Data Architect, ...

Extensible Widget Pool

Authoring

Collaboration

Visualization

Search and Analytics

Data Management Modules

Information Extraction and Enrichment

Interlinking and Integration

Publishing Workflows

Data Access

Querying and Search

Data Layer

Unpublished Data

Published Data

Staging Database

Live Database

SPARQL/RDF HTTP
User Roles and Editorial Workflow

- **Journalist**
  View Instance Data

- **Subeditor**
  Edit instance data

- **Media Manager**
  Edit instance data
  Approve/reject instance data edits

- **Data Architect**
  Edit instance data and ontology data edits
  Publish instance data

![Workflow Diagram]

- Draft
  - Edit
  - Approve
  - Reject

- Approved
  - Publish

- Rejected
  - Reject
Demo Dynamic Semantic Publishing
Example: Data Center Management

Multitude of managed resources
• Hardware (physical storage, network, computational infrastructure)
• Virtualization capabilities (virtual clusters, live migration)
• Software applications

Multitude of APIs and data sources

Tool sprawl!

Source: http://www.fluidops.com/information-workbench/
Example: eCloudManager – Integrated View on the Data Center

- Integration of different SW and HW components, storage systems, compute infrastructures, applications, CRM systems, ticket systems, project catalogs.
- Automatic correlation of data retrieved from various systems.
- Unified view on data and metadata across the border of company units.
- Exploration, analysis, and actions based on the entire data corpus.

Source: http://www.fluidops.com/ecloudmanager/

Integrated view showing connections between hardware layer, application layer, projects, and customers
Data Center Management

- Support collaborative operations management in the data center
  - Link business data to technical data
  - Technical Documentation
  - Analytics and Reporting
  - Performance and Capacity Monitoring
  - Responsibility Management
  - Resource Management
  - Change Management
  - Technical Ticketing System
Optique – Scalable End-User Access to Big Data
Optique Project

- **Budget:** 13 800 000 €
- **Running until November 2016**
- **10 partners**
- **5 countries**
- **100 man years**
- **FP7 Large-scale IP project**
Optique Use Cases

Up to 80% time on data access

Turnaround time: weeks

Up to 70% time on data access

Turnaround time: days

© Siemens AG

© Harald Pettersen/Statoil
Scalable End-user Access to Big Data

Up to 80% of experts’ time spent accessing Big Data
Ontology Based Data Access: From Weeks to Hours
Optique Big Data

Optique Platform

SIEMENS

30GB/24h
Parallel Streams

3000 Tables

EPDS 10 TB
OpenWorks 100 TB
PetroBank 3 PB

150 TB

Diagnostic centres
turbines

raw data
structured data

Complexity

Statoil
Optique – Unique Combination of Techniques

OPTIQUE PLATFORM

- End-user-oriented Query Interface
- Scalable Query Rewriting
- Real-time Stream Processing
- Query Evaluation with Elastic Clouds
Optique (OBDA) Architecture

End-user

Application

Query Formulation

Ontology

Mappings

Ontology & Mapping Management

IT-expert

Data models
Std. ontologies...

Query Transformation

Query Planning

Stream Adapter

Query Execution

streaming data
Optique Demo

http://fact-pages.fluidops.net/resource/Start
http://www.cs.ox.ac.uk/isg/projects/Optique/demos/iswc2013/
Review of Agenda - Summary

• Introduction
  • Linked Data and Semantic Technologies at fluidOps
  • Information Workbench Platform

• Foundations of Linked Data Applications
  • Example: MusicBrainz
  • Building Linked Data Applications
  • Providing and Integrating Linked Data
  • Interacting with Linked Data

• Examples of Enterprise Applications
  • Dynamic Semantic Publishing
  • Enterprise Cloud Management
  • Optique: Scalable End-user Access to Big Data
Further Information

Information Workbench product page

• http://www.fluidops.com/information-workbench/

Demo system

• http://musicbrainz.fluidops.net/

Download a free Community Edition version

• http://www.fluidops.com/information-workbench/iwb-download/

Online documentation

• http://help.fluidops.com/