Towards Easy Matching Between Statistical Linked Data: Dimension Patterns

Hideto Sato and Wen Wen

First International Workshop on Semantic Statistics
(SemStats 2013)
22 October 2013, Sydney
Introduction

- For matching statistical data from **different sources**, **upper concepts** and **schema-level links** are important.

- **Three Problems**
  1. A small number of upper concepts are available.
  2. Certain patterns of dimension description prevent some schema-level links.
  3. Usage of **external codes** is hard to find in a schema-level.

- This paper focuses on (2) and (3), and propose **patterns of dimension description** to improve them.
Trial Matching

• Italian Immigration Statistics
  ⇒ the numbers of immigrants to Italy by birth country by year

• World Bank Statistics
  ⇒ the total population by country by year

• Integrated Statistics
  Percentage of Immigrants to Italy by country by year
(1) What role does the dimension play?

- place of residence
- place of birth
(2) What type of code does the dimension use?
- Countries
- Domestic Administrative Areas
- River Basins, and so on.
(3) What common codes are available?

- Geonames
- DBPedia

preferably in the schema-level
Matching Data from Different Sources

The following questions are important for each dimension. As for an area dimension,

**For Dimension Properties**
What role does the dimension play?
- Place of Birth
- Place of Residence

**For Code Class (Range of Dimension)**
What type of code does the dimension use?
- Countries
- Domestic Administrative Areas
- River Basins

**For Code Values**
What common codes are available?
- Geonames
- DBPedia
Matching Data from Different Sources

The following questions are important for each dimension. As for an area dimension:

**For Dimension Properties**
- What role does the dimension play?
  - Place of Birth
  - Place of Residence

**For Code Class (Range of Dimension)**
- What type of code does the dimension use?
  - Countries
  - Domestic Administrative Areas

**For Code Values**
- What common codes are available?
  - Geonames
  - DBPedia

2013/10/22
QB and Upper Concepts

QB: The RDF Data Cube Vocabulary

QB provides a bridge to upper concepts by referring to the SDMX-RDF vocabulary.
### Upper Concepts and SDMX-RDF

<table>
<thead>
<tr>
<th>Upper concept</th>
<th>Upper resource in SDMX-RDF</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dimension Property</strong></td>
<td></td>
</tr>
<tr>
<td>Place of Birth</td>
<td>sdmx-dimension:visArea</td>
</tr>
<tr>
<td>Place of Residence</td>
<td>sdmx-dimension:refArea</td>
</tr>
<tr>
<td><strong>Code Class (Range of Dimension)</strong></td>
<td></td>
</tr>
<tr>
<td>Area</td>
<td>sdmx-code:Area</td>
</tr>
<tr>
<td>Country</td>
<td>(not defined)</td>
</tr>
<tr>
<td>Domestic Area</td>
<td>(not defined)</td>
</tr>
<tr>
<td>River Basin</td>
<td>(not defined)</td>
</tr>
</tbody>
</table>

*(sdmx-dimension:visArea has been removed in the current version of SDMX-RDF.)*
Dimension Description in QB

**Upper**

- `sdmx-dimension:refArea` (upper: abstract DimensionProperty)

**Local**

Data Structure Definition

- `eg:refArea` (local: DimensionProperty)
  - `rdfs:subPropertyOf` qb:dimension
  - `rdfs:range` qb:codeList

Dimension Property

- `eg:UnitaryAuthority` (local: CodeClass)
  - `rdfs:subClassOf` Code Class
  - `rdf:type` `eg:UnitaryAuthority`

Code Class

- `sdmx-code:Area` (upper: AbstractCodeClass)
  - `rdfs:subClassOf` Code Class

Code List

- `eg:areaCodeList` (local: codeList)
  - `rdf:type` `eg:areaCodeList`
  - `skos:hasTopConcept` | qb:hierarchyRoot

Code

- `eg:cardiff_00pt` (local: code)
  - `rdf:type` `eg:cardiff_00pt`
Anti-Patterns

• Two Anti-Patterns prevent describing schema-level links properly.
  — Direct use of an abstract upper resource
  — Direct use of an external code class
Anti-Pattern: Direct Use of an Upper Resource

**Upper**

- `sdmx-dimension:refArea`
  - `(upper: abstract DimensionProperty)`

**Local**

**Data Structure Definition**

- `sdmx-code:Area`
  - `(upper: AbstractCodeClass)`

**Dimension Property**

- `rdfs:subClassOf`
- `qb:codeList`
- `rdfs:range`

**Code Class**

- `eg:UnitaryAuthority`
  - `(local:CodeClass)`

**Code List**

- `eg:areaCodeList`
  - `(local:codeList)`

**Code**

- `skos:hasTopConcept`
  - `| qb:hierarchyRoot`
- `eg:cardiff_00pt`
  - `(local:code)`

2013/10/22
The Pattern for Using a Local Code Class

**Upper**
- sdmx-dimension:refArea (upper: abstract DimensionProperty)

**Local**
- Data Structure Definition
  - qb:dimension
  - rdfs:range
  - eg:refArea (local: dimensionProperty)
- Code Class
  - eg:UnitaryAuthority (local: CodeClass)
- Code List
  - eg:areaCodeList (local: codeList)
- Code
  - eg:cardiff_00pt (local: code)

- rdfs:subPropertyOf
  - eg:refArea (local: dimensionProperty)
  - qb:codeList
  - sdmx-code:Area (upper: AbstractCodeClass)
  - rdfs:subClassOf
  - eg:UnitaryAuthority (local: CodeClass)
  - rdf:type
  - eg:refArea (local: dimensionProperty)
  - skos:hasTopConcept | qb:hierarchyRoot
Realization of Dimension Property

Direct Use of an External Code Class
The Pattern for Using an External Code Class

**Upper**
- sdmx-dimension:refArea
  - (upper: abstract DimensionProperty)

**Local**
- eg:refArea
  - (local: dimensionProperty)
  - rdfs:subPropertyOf qb:dimension

**External**
- Dimension Property
- Data Structure Definition
- Code Class
- Code List
- Code

- eg:UnitaryAuthority
  - (local: CodeClassAdapter)
  - rdfs:subClassOf sdmx-code:Area
    - (upper: AbstractCodeClass)
  - rdfs:range qb:codeList
  - qb:hierarchyRoot
  - owl:equivalentClass
    - <http://www.geonames.org/ontology#Feature>
      - (external:CodeClass)

- eg:areaCodeList
  - (local: codeList)
  - rdf:type sdmx-code:Area
    - (upper: AbstractCodeClass)

2013/10/22
Alternate Code Class

When using both local and external code classes, it is difficult to find whether an external code class is employed or not.

We need a schema-level description for an alternate code class.
Using Local and External Code Classes

**Local**

- **Dimension Property**
  - eg:refArea
    - rdfs:range
      - eg:UnitaryAuthority
        - qb:codeList
          - rdf:type
            - Local

- **Code List**
  - eg:areaCodeList
    - skos:hasTopConcept
      - qb:hierarchyRoot

- **Code**
  - eg:cardiff_00pt
    - skos:exactMatch
      - owl:sameAs

**External**

- **Data Structure Definition**
  - qb:dimension
    - rdfs:range
      - eg:UnitaryAuthority
        - rdf:type
          - External

- **Code List**
  - eg:unitaryAuthority
    - qb:codeList
      - rdf:type
        - External

- **Code**
  - eg:cardiff_00pt
    - skos:exactMatch
      - owl:sameAs

2013/10/22
Proposal of an additional link (ext:altClass)
### From Our Survey

<table>
<thead>
<tr>
<th>Area Dimension</th>
<th>Time Dimension</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct Use of an Upper Resource</td>
<td>3/12</td>
</tr>
<tr>
<td>Direct Use of an External Code Class</td>
<td>2/12</td>
</tr>
<tr>
<td>Use of Alternate Code Classes</td>
<td>10/12</td>
</tr>
</tbody>
</table>

The counts are DSDs (Data Structure Definitions) found in the endpoints listed at http://www.w3.org/2011/gld/wiki/Data_Cube_Implementations.
Conclusion

• We introduced **dimension patterns** for describing **schema-level links** including **references to upper resources** and **alternate class links**.

• These will extract the QB's power of description to its full extent.

• However, only **a few upper resources** are available now. Therefore, the part of the patterns concerning to upper concepts are **preparatory** for the future.

• We think that it is an urgent task to **enrich upper resources** suitable for statistical data.