Publishing OLAP Cubes on the Semantic Web

Alejandro Vaisman
Instituto Tecnológico de Buenos Aires
avaisman@itba.edu.ar

Outline

- The Semantic Web
- RDF and SPARQL Basics
- Vocabularies for OLAP on the SW: QB and QB4OLAP
- Modeling Data Cubes on the Semantic Web using QB4OLAP
- Querying Data Cubes on the Semantic Web
- Summary

Outline

- **The Semantic Web**
- RDF and SPARQL Basics
- Vocabularies for OLAP on the SW: QB and QB4OLAP
- Modeling Data Cubes on the Semantic Web using QB4OLAP
- Querying Data Cubes on the Semantic Web
- Summary

The Web

- Huge amounts of information on the Web.
- Mostly for consumption by **humans** who must:
  - Recognize the meaning behind content and draw conclusions
  - Infer new knowledge using context
  - Understand background information
  - Manually link to other related sources

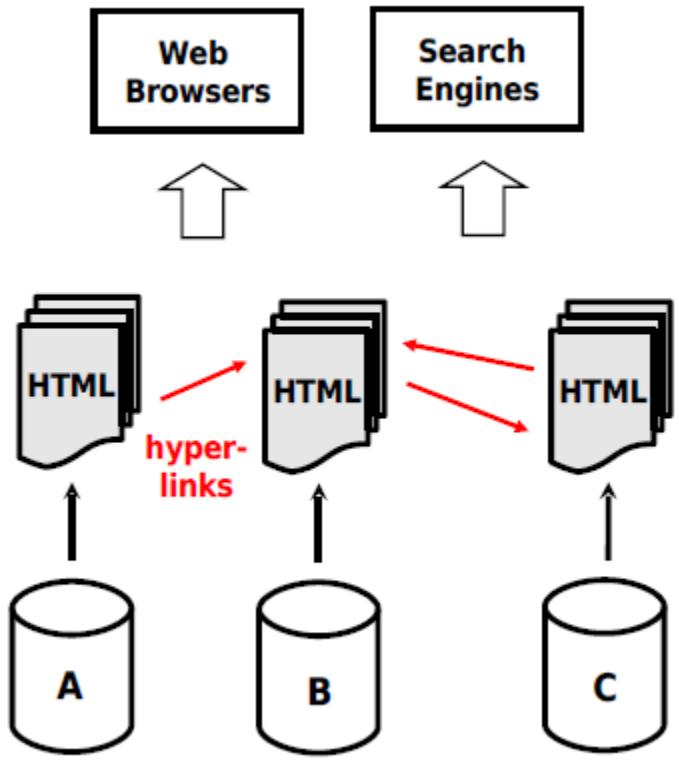
Germany – Wikipedia
de.wikipedia.org/wiki/**Germany**
Germany ist: die englische Bezeichnung für Deutschland. **Germany** (Mondkrater), ein Mondkrater; **Germany** (Rapper), deutscher Rapper. Zudem steht der Name ...

Germany - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/**Germany** - Diese Seite übersetzen
Song of the Germans. Location of **Germany** (dark green)– in Europe (green & dark grey). Location of **Germany** (dark green). – in Europe (green & dark grey) ...
Flag of Germany - History of Germany - Geography of Germany - German cuisine

Tourism in **Germany** – travel, breaks, holidays
www.**germany**.travel/ - Diese Seite übersetzen
Tourism in **Germany** – travel, breaks, holidays. ... BMWi Logo **Germany**, the travel destination. Towns, cities & culture Towns, cities & culture; Leisure and ...

Startseite: Das Deutschland-Portal
www.deutschland.de/
deutschland.de ist das offizielle und unabhängige Portal der Bundesrepublik Deutschland im Internet. Es bietet in fünf Sprachen eine Sammlung wichtiger ...

The Classic Web



Single global information space

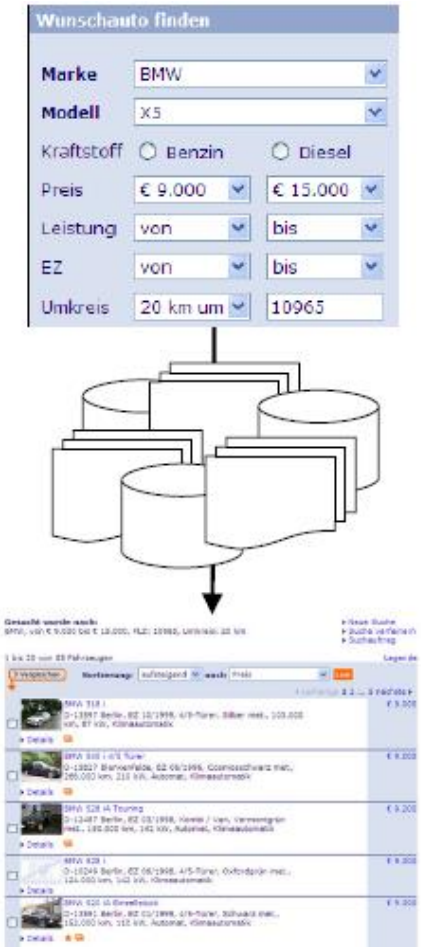
- URLs as
  - Globally unique IDs
  - Retrieval mechanism
- HTML as shared content format
- Hyperlinks

Shortcomings

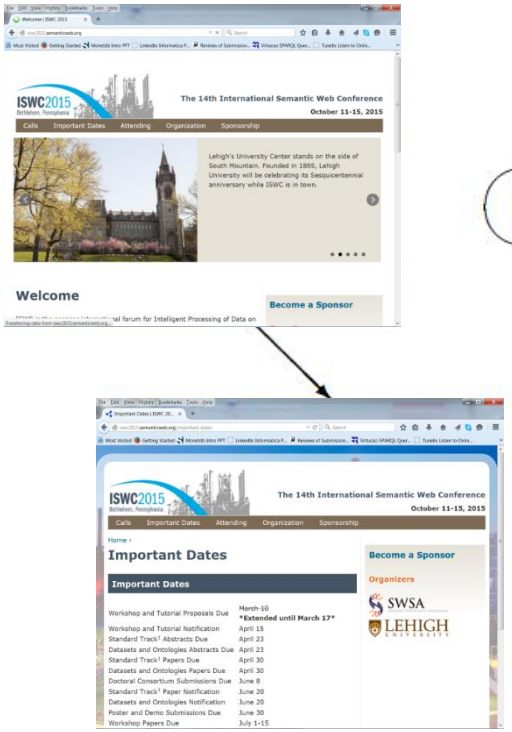
- Content is not well structured
- Cannot ask expressive queries

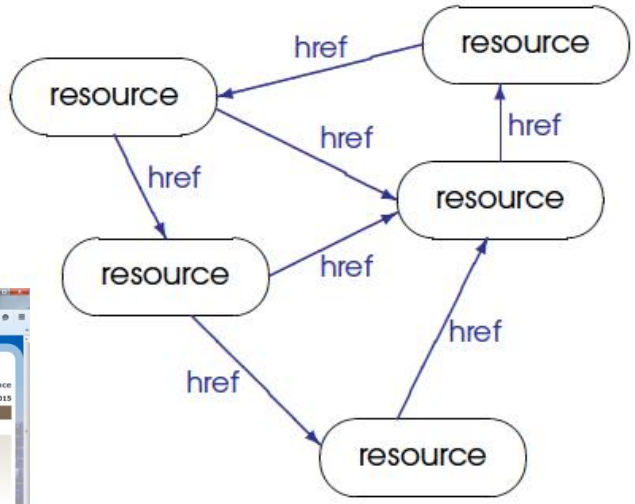
What do we want?

Use the web as a single global database

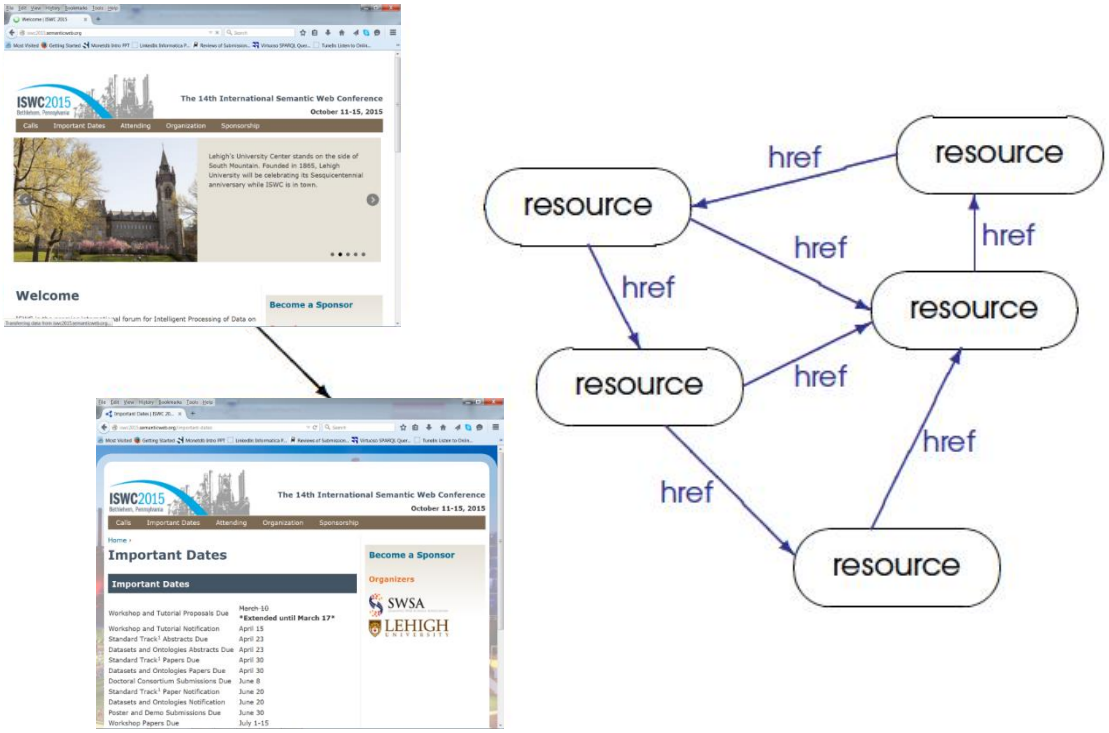


Today: the “syntactic” web





Today: the “syntactic” web

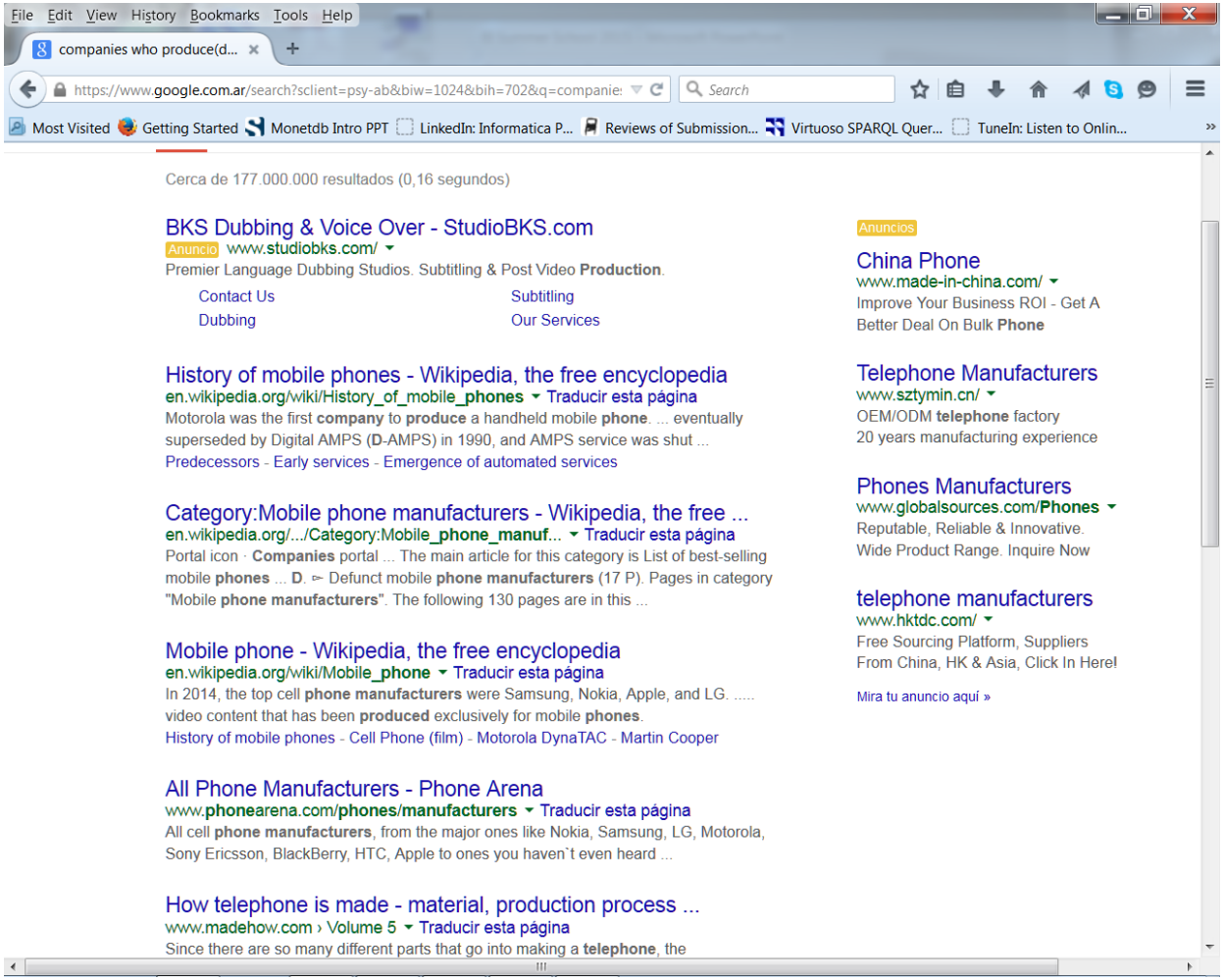


Can we answer “*Last ISWC Conference held in a city with more than 1000 K people*” ?
(i.e., complex queries involving background knowledge)

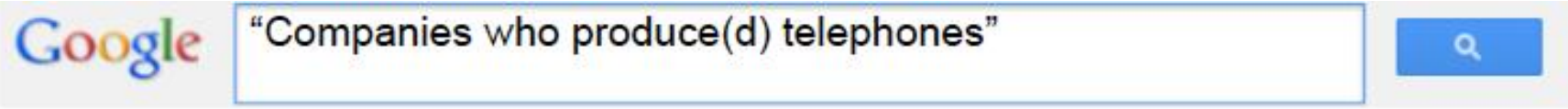
Example (with the “syntactic web”)

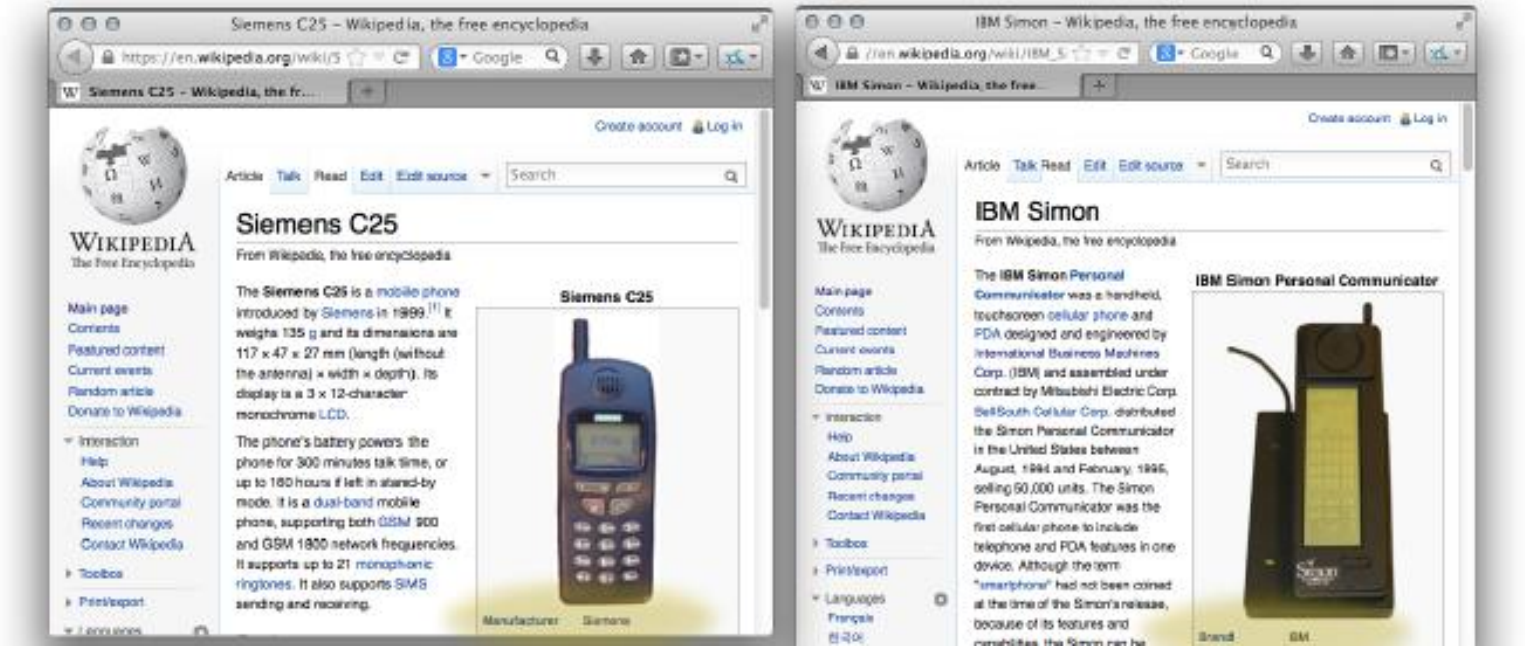
- “Companies which produce(d) telephones”

We obtain…

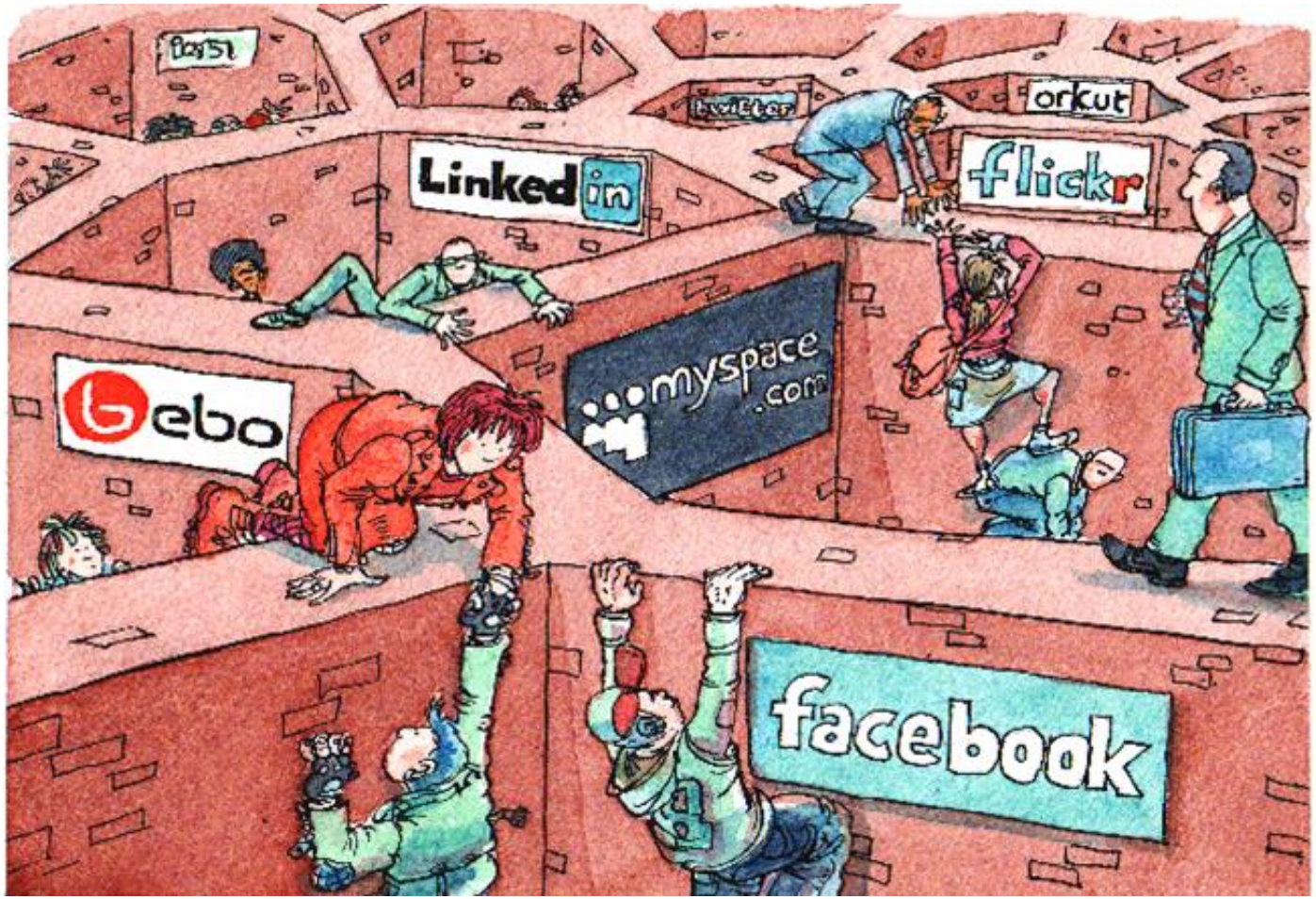

File Edit View History Bookmarks Tools Help
companies who produce(d...
https://www.google.com.ar/search?sclient=psy-ab&biw=1024&bih=702&q=companie:
Search
Most Visited
Getting Started
Monetdb Intro PPT
LinkedIn: Informatica P...
Reviews of Submission...
Virtuoso SPARQL Quer...
TuneIn: Listen to Onlin...
Cerca de 177.000.000 resultados (0,16 segundos)
BKS Dubbing & Voice Over - StudioBKS.com
Anuncio www.studiobks.com/
Premier Language Dubbing Studios. Subtitling & Post Video Production.
Contact Us
Subtitling
Dubbing
Our Services
History of mobile phones - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/History_of_mobile_phones Traducir esta página
Motorola was the first company to produce a handheld mobile phone. ... eventually superseded by Digital AMPS (D-AMPS) in 1990, and AMPS service was shut ...
Predecessors - Early services - Emergence of automated services
Category:Mobile phone manufacturers - Wikipedia, the free ...
en.wikipedia.org/.../Category:Mobile_phone_manuf... Traducir esta página
Portal icon · Companies portal ... The main article for this category is List of best-selling mobile phones ... D. ▻ Defunct mobile phone manufacturers (17 P). Pages in category "Mobile phone manufacturers". The following 130 pages are in this ...
Mobile phone - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/Mobile_phone Traducir esta página
In 2014, the top cell phone manufacturers were Samsung, Nokia, Apple, and LG. ..... video content that has been produced exclusively for mobile phones.
History of mobile phones - Cell Phone (film) - Motorola DynaTAC - Martin Cooper
All Phone Manufacturers - Phone Arena
www.phonearena.com/phones/manufacturers Traducir esta página
All cell phone manufacturers, from the major ones like Nokia, Samsung, LG, Motorola, Sony Ericsson, BlackBerry, HTC, Apple to ones you haven`t even heard ...
How telephone is made - material, production process ...
www.madehow.com › Volume 5 Traducir esta página
Since there are so many different parts that go into making a telephone, the
Anuncios
China Phone
www.made-in-china.com/
Improve Your Business ROI - Get A Better Deal On Bulk Phone
Telephone Manufacturers
www.sztymin.cn/
OEM/ODM telephone factory
20 years manufacturing experience
Phones Manufacturers
www.globalsources.com/Phones
Reputable, Reliable & Innovative. Wide Product Range. Inquire Now
telephone manufacturers
www.hktdc.com/
Free Sourcing Platform, Suppliers From China, HK & Asia, Click In Here!
Mira tu anuncio aquí »

We would like….

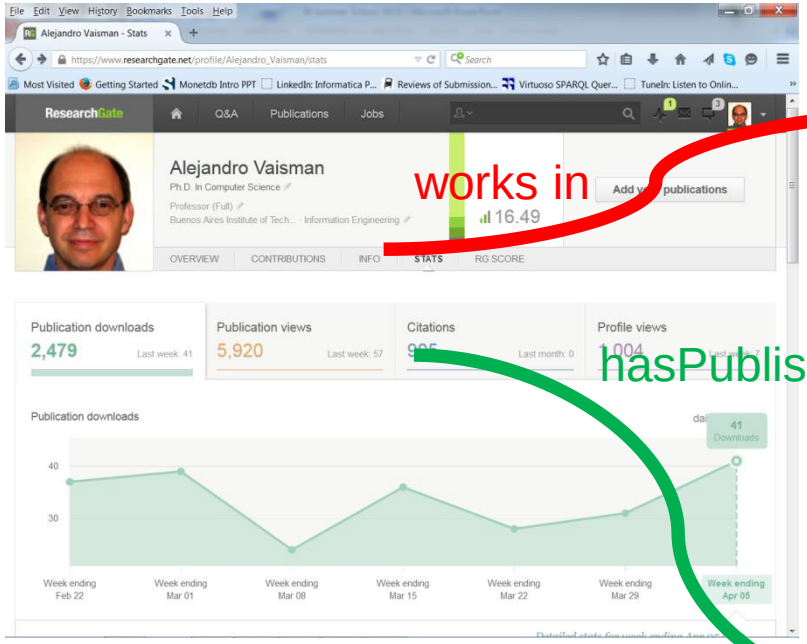


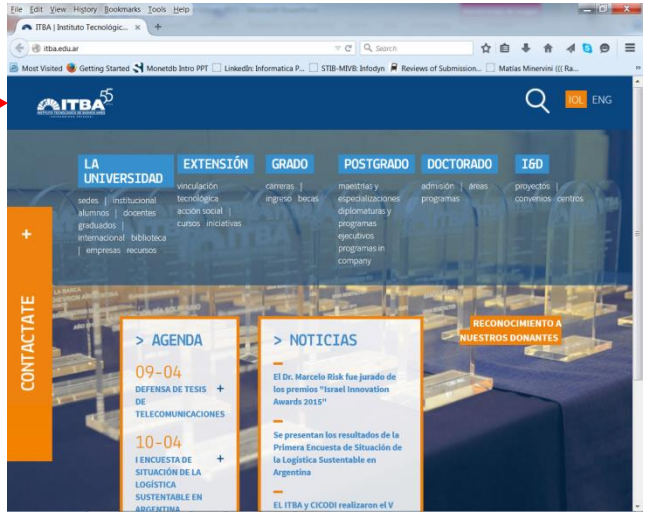


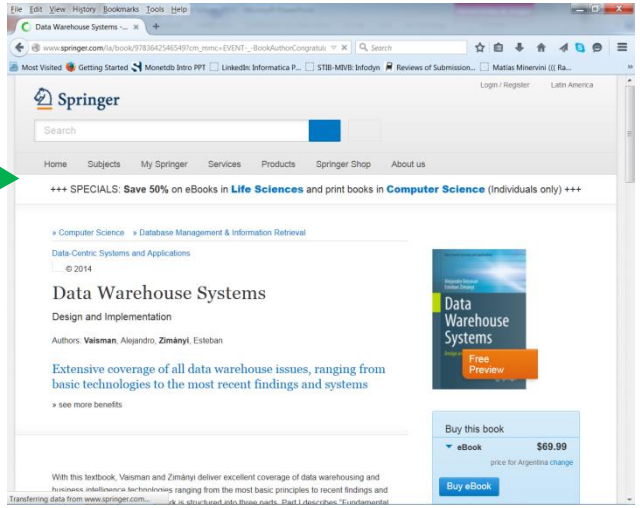
Problem: The web today looks like…



What is missing?







- Information must be added to links/data
- Information must be machine-readable
- This added info should follow standards

What is the Semantic Web, then?

The SW can be thought as a collection of standard technologies
to implement a Web of Data.

1. Formal languages to describe and query the data and their connections
2. Formal rules that allow machines to extract information from data
3. Ontologies and vocabularies must be created to describe data
4. Corresponding technologies and efficient tools

Sir Tim's Vision

- TBL's vision of the Web was much more ambitious:

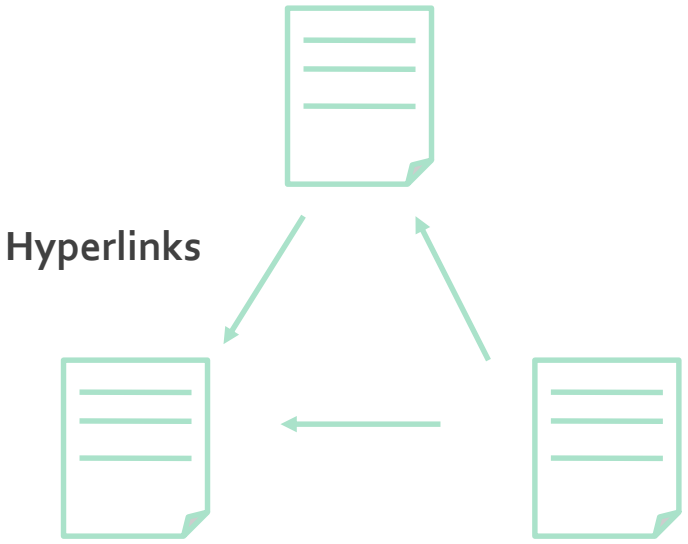
"I have a dream for the Web (in which computers) become capable of analyzing all the data on the Web – the content, links, and transactions between people and computers. A **Semantic Web**, which should make this possible, has yet to emerge, but when it does, the day-to-day mechanisms of trade, bureaucracy and our daily lives will be handled by machines talking to machines. The intelligent agents people have touted for ages will finally materialize." (Berners-Lee, 1999)

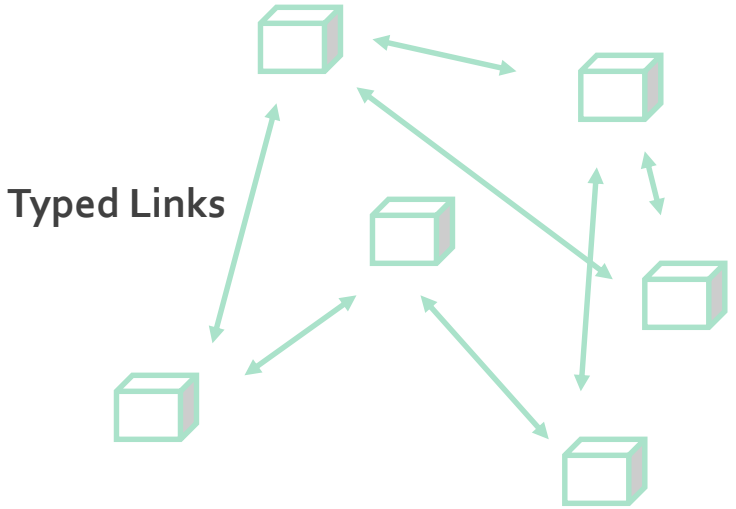
The **Semantic Web** is a 'web of data' that facilitates machines to understand the semantics, or meaning, of information on the WWW. It extends the network of hyperlinked human-readable web pages by inserting machine-readable metadata about pages and how they are related to each other, enabling automated agents to access the Web more intelligently and perform tasks on behalf of users

The term **Semantic Web** was coined by Berners-Lee, now the director of the World Wide Web Consortium (W3C), which oversees the development of proposed Semantic Web standards.

The Web: Evolution

Web of Documents → Web of Data





"Documents" "Things"

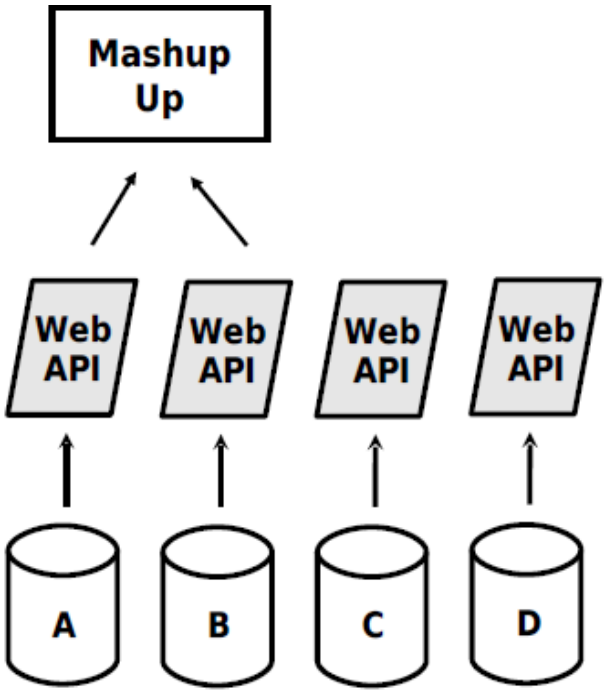
How do we achieve this?

A First Approach: Web APIs

- Provide simple query access to structured data over http

Web APIs

- Consequence: An explosion of small specialized applications that combine data from several sources (mashups) => each new dataset = a new application



Positive:

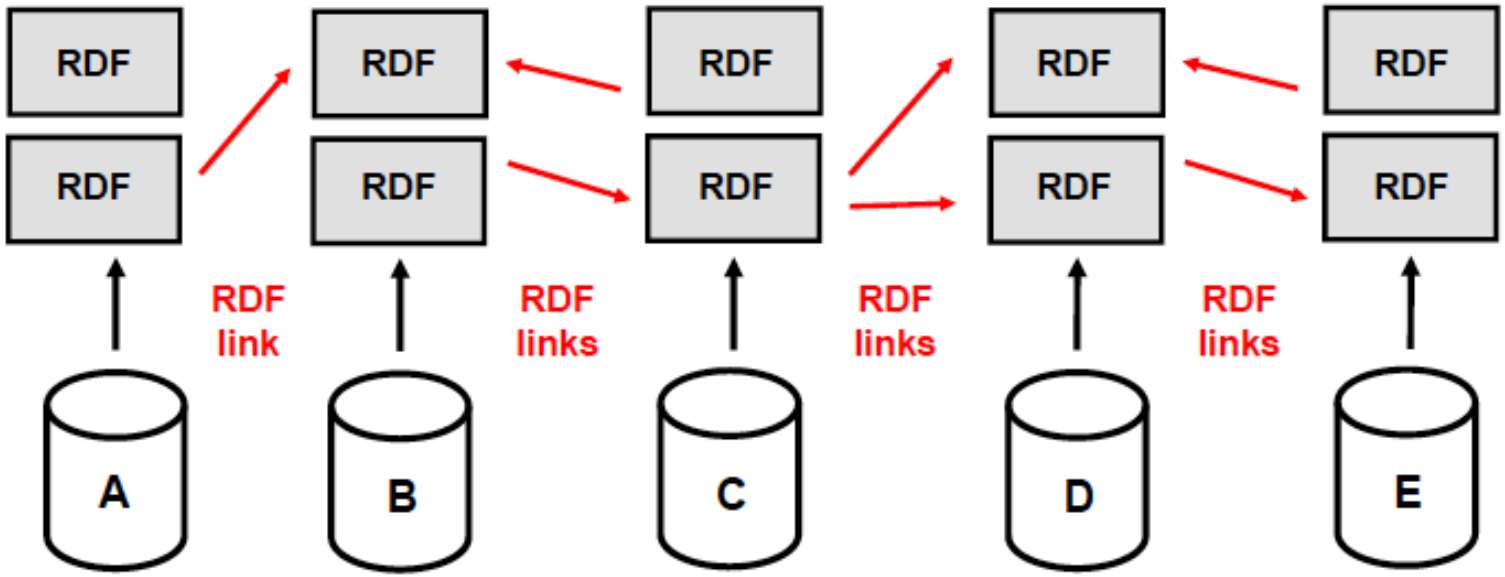
APIs expose structure data
APIs enable new applications

Negative:

Proprietary interfaces
Mashups based on fixed set of sources
Cannot set hyperlinks between objects

Linked Data

- Extends the web as a single global dataspace
  - Using RDF to publish structured data on the web
  - Setting links between data items within different data sources

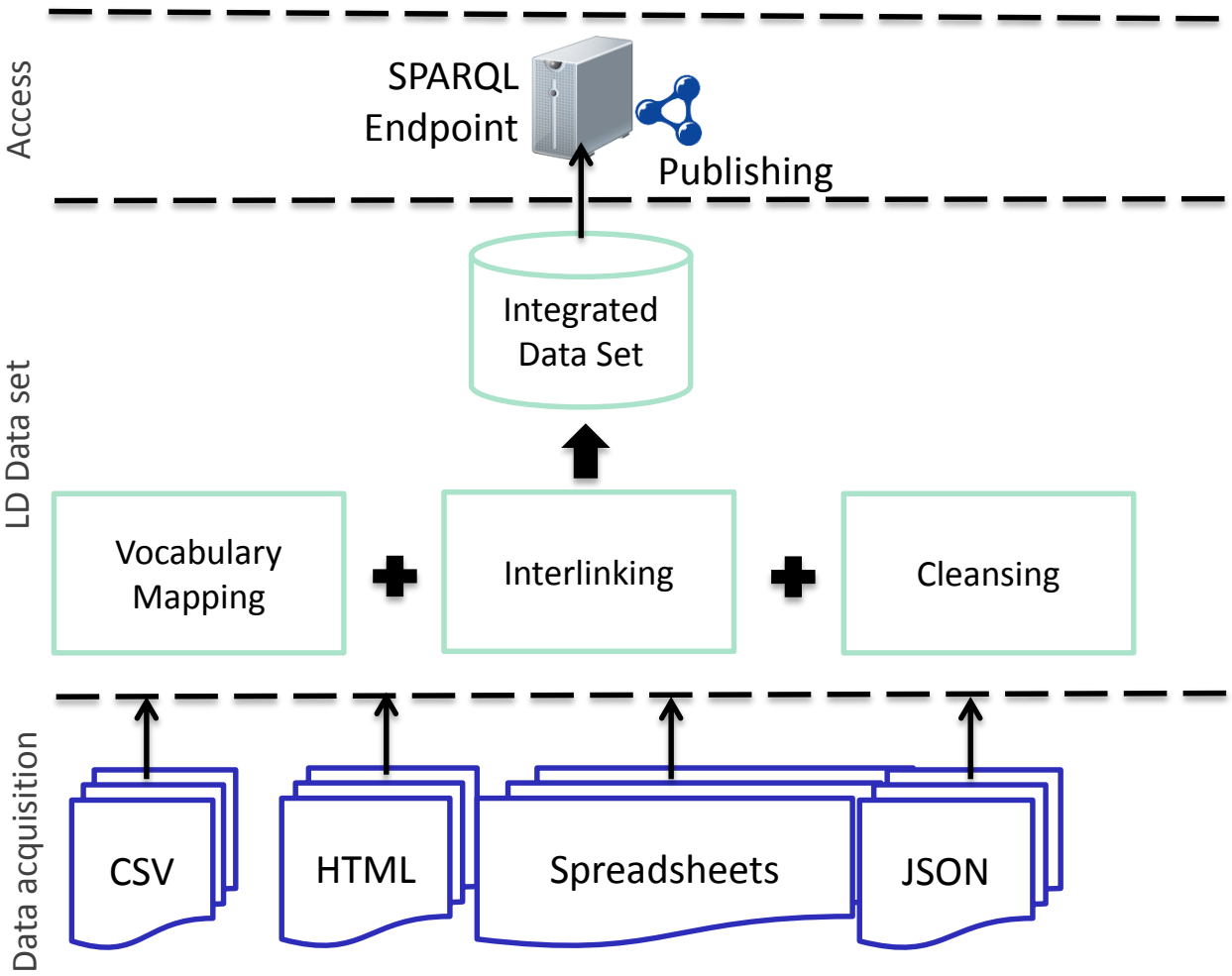


Linked Data Principles

Set the best practices for publishing structured data on the web

1. Use IRIs as **names** for things
2. Use HTTP IRIs so that users can **look up** those names
3. When someone looks up an IRI, **provide useful information**, using the standards (RDF, SPARQL)
4. Include links to other IRIs, so that users can **discover** more things (e.g., though RDF statements that link to other IRIs)

Typical architecture



- **Task**: Integrate information into an RDF database
- Data may be available in non-RDF formats:
  - Plain text
  - CSV, TSV files
  - HTML tables
  - Spreadsheets
  - XML
  - JSON

Kinds of data

CSV

```
The Beatles, 250 million
Elvis Presley, 203.3 million
Michael Jackson, 157.4 million
Madonna, 160.1 million
Led Zeppelin, 135.5 million
Queen, 90.5 million
```

JSON

```
{
  "artist": {
    "class": "artist",
    "name": "The Beatles"
  },
  "rank": 1,
  "value": 250 million
},
…
```

HTML tables

Artist	Country of origin	Period active	Release-year of first charted record	Total certified units (from available markets)[Notes]
The Beatles	United Kingdom	1960–1970[4]	1962[4]	**Total available certified units: 250 million**[show]
Elvis Presley	United States	1954–1977[28]	1954[28]	**Total available certified units: 203.3 million**[show]
Michael Jackson[Note 2]	United States	1964–2009[32]	1971[32]	**Total available certified units: 157.4 million**[show]
Madonna	United States	1979–present[44]	1982[44]	**Total available certified units: 160.1 million**[show]
Led Zeppelin	United Kingdom	1968–1980[50]	1969[50]	**Total available certified units: 135.5 million**[show]
Queen	United Kingdom	1971–present[53]	1973[53]	**Total available certified units: 90.5 million**[show]

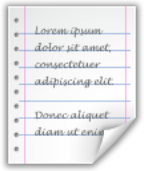
http://en.wikipedia.org/wiki/List_of_best-selling_music_artists

Extracting the Data

- Data may be stored in many **formats**

Spreadsheets
or tabular data

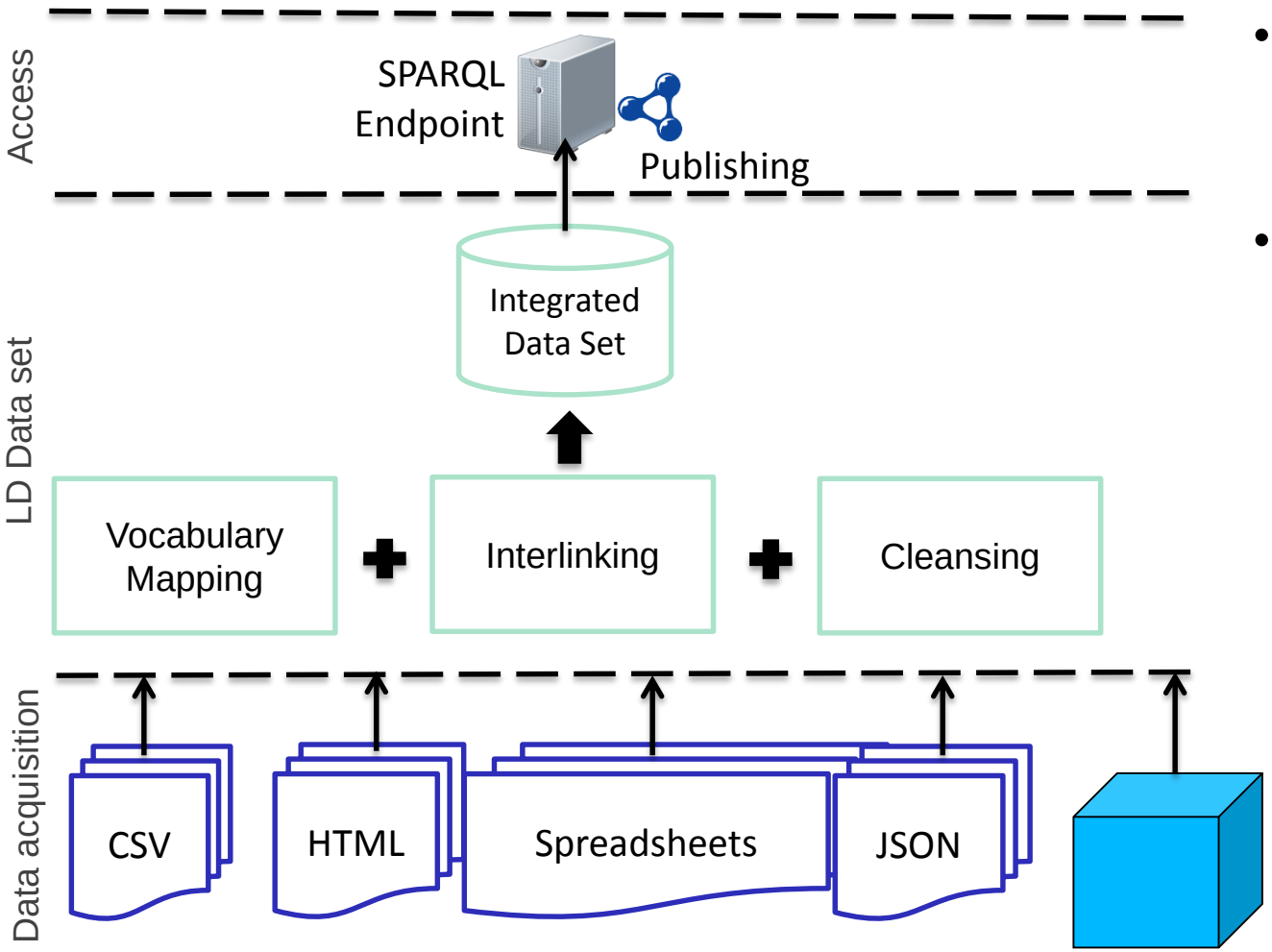
Databases



text

R2RML
Direct Mapping

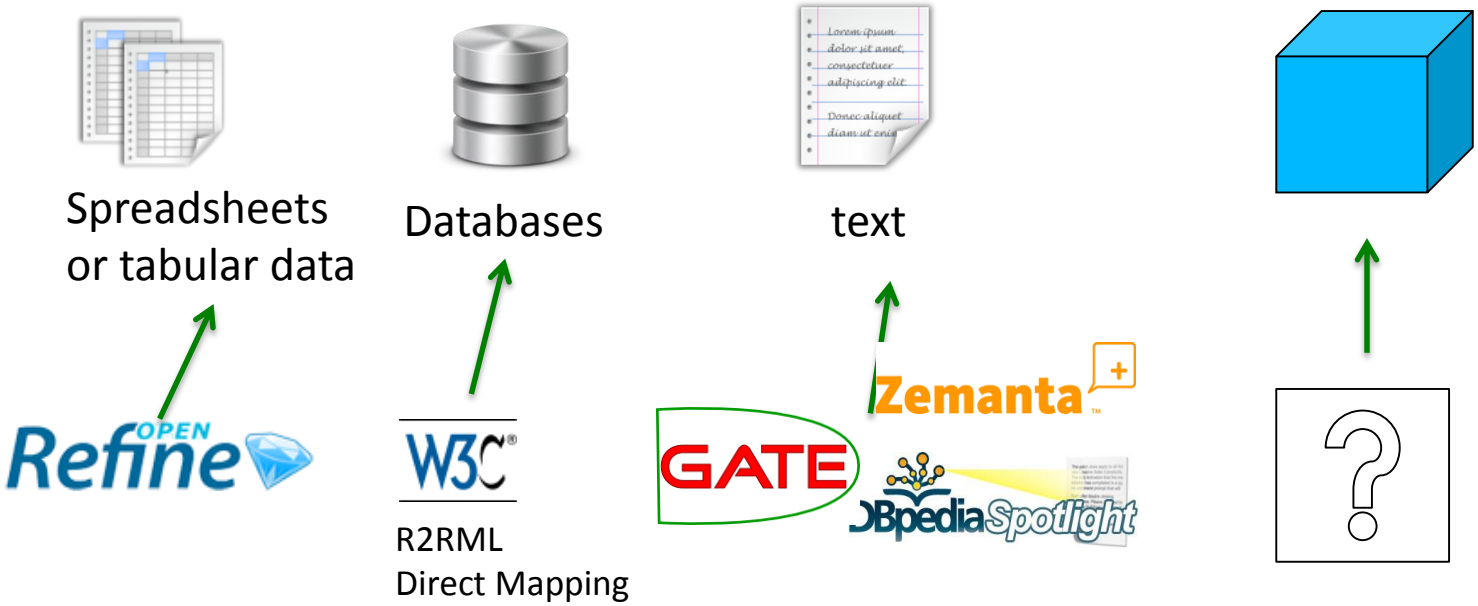
Typical architecture



- **Task**: Integrate latest information into an RDF database.
- Data may be available in non-RDF formats:
  - Plain text
  - CSV, TSV files
  - HTML tables
  - Spreadsheets
  - XML
  - JSON
  - **MD data?**

Extracting the Data

- Data may be stored in many **formats**



Publishing Linked Data: Vocabularies

- Collections of defined **relationships** and **classes** of resources.
  - Classes group together similar resources.
- Terms from **well-known vocabularies** should be **reused** wherever possible.
- New terms should be defined only if you can not find required terms in existing vocabularies.
  - This is the case of QB4OLAP

Publishing Linked Data: Vocabularies

Some well-known vocabularies

Vocabulary	Description	Classes and Relationships
Friend-of-a-Friend **(FOAF)**	Vocabulary for describing people.	foaf:Person, foaf:Agent, foaf:name, foaf:knows, foaf:member
Dublin Core **(DC)**	Defines general metadata attributes.	dc:FileFormat, dc:MediaType, dc:creator, dc:description
Semantically-Interlinked Online Communities **(SIOC)**	Vocabulary for representing online communities.	sioc:Community, sioc:Forum, sioc:Post, sioc:follows, sioc:topic
Music Ontology **(MO)**	Provides terms for describing artists, albums and tracks.	mo:MusicArtist, mo:MusicGroup, mo:Signal, mo:member, mo:record
Simple Knowledge Organization System **(SKOS)**	Vocabulary for representing taxonomies and loosely structured knowledge.	skos:Concept, skos:inScheme, skos:definition, skos:example

Publishing Linked Data: Vocabularies

SKOS

- Simple Knowledge Organization System
  - http://www.w3.org/TR/skos-reference/
- Data model for knowledge organization systems (thesauri, classification scheme, taxonomies)
- SKOS data expressed as RDF triples
- Allows the creation of RDF links between different data sets with the usage of **mapping properties**

Publishing Linked Data: Vocabularies

SKOS: Mapping properties

- Used to link SKOS concepts (particularly instances) in different schemes:
  - skos:closeMatch: Links two concepts that are sufficiently similar (sometimes can be used interchangeably)
  - skos:exactMatch: Indicates that the two concepts can be used interchangeably. Axiom: It is a **transitive** property
  - skos:relatedMatch: States an associative mapping link between two concepts

Linking the dataset

- owl:sameAs
  - Creates links between individuals
  - States that two URIs refer to the same individuals
- rdfs:seeAlso
  - States that a resource may provide additional information about the subject resource
- Example: Links in **MusicBrainz**:
  - owl:sameAs is used for music artists
  - rdfs:seeAlso is used for albums

Publishing MD Linked data

- OLAP (On-line Analytical Processing) allows analyzing huge amounts of data for decision-making
- Multidimensional data are seen as data cubes (DC).
- Many situations where it would be useful to publish multi-dimensional data on the web
  - Publish existing ROLAP/MOLAP cubes on the web
  - Create cubes on self-service basis from web and/or other data
  - Modify public open data (e.g., World Bank, Eurostat) for OLAP analysis
  - Link MD data to related datasets and concepts (e.g., Dbpedia, Open Street Maps, etc.)

Publishing MD Linked data

- Represent multidimensional data using Semantic Web standards (eg: RDF, RDF-S)
  - Publish and share multidimensional schemas and data instances (e.g. open data, government data)
- Requirements:
  - Alignment with Linked Data and Linked Open Data principles
- Standards:
  - The Data Cube vocabulary (QB) provides means to do this using the RDF standard.
  - The model underlying the Data Cube vocabulary compatible with (Statistical Data and Metadata eXchange) – SDMX

Publishing MD Linked data

The Data Cube Vocabulary - QB [Cyganiak et al. 2012]

- Oriented to statistical data analysis
- Used to publish statistical data
- W3C Recommendation January 2014

- However, QB…
  - Does not directly support the classical multidimensional model for OLAP
  - Does not represent dimension structure
  - Does not bind measures to aggregate functions
  - Dimension hierarchies not accounted for directly
  - Consequence: OLAP operators are difficult to define over QB (see Kämpgen et al., ILD, ESWC 2012).

Publishing MD Linked data

QB4OLAP: A vocabulary that enhances QB

- QB4OLAP: An RDF vocabulary that fully represents the classical multidimensional model
- On top: A set of OLAP operators implemented as SPARQL queries
- To be usable we need:
  - Build QB4OLAP cubes from QB cubes, allowing reusing published data without rewriting the cube
  - Methodology and algorithms that (semi-) automatically build the SPARQL queries that implement OLAP operators
  - Query optimization

Publishing MD Linked data

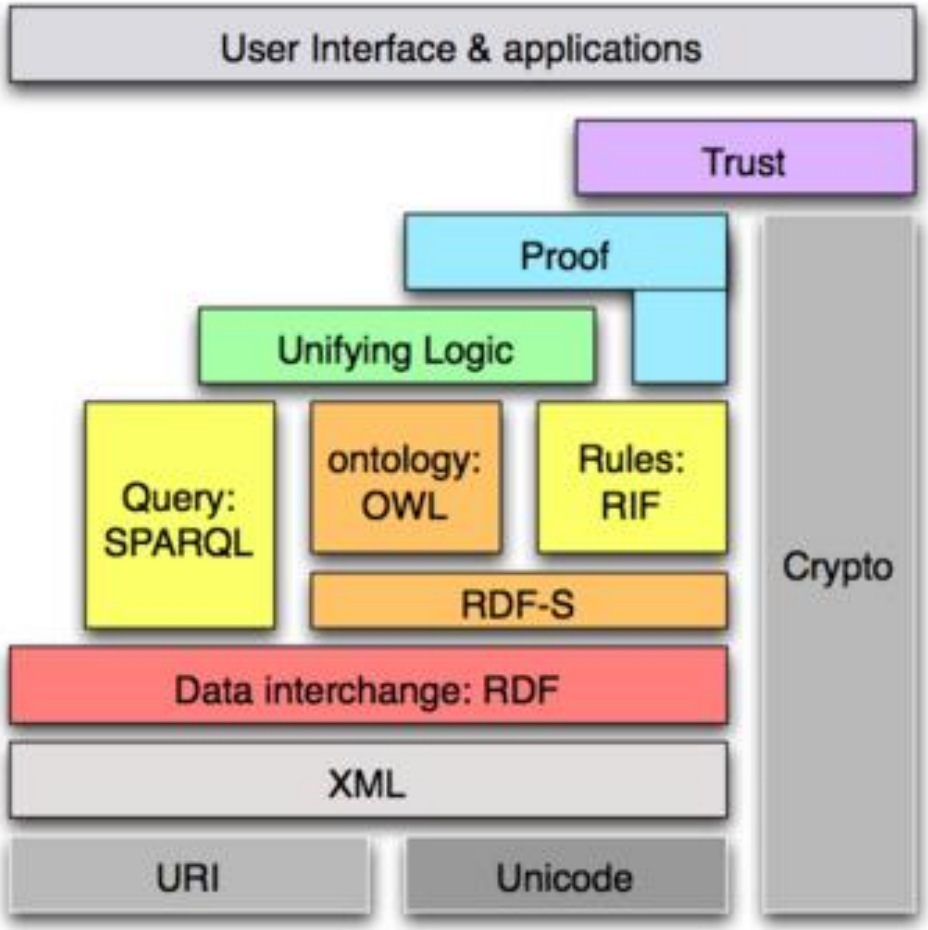
In this talk we will ...

- Explain the basics of MD data publication on the web
- Describe QB
- Show QB's limitations when it comes to support MD data
- Discuss QB4OLAP as an alternative to QB
- Explain how to build OLAP cubes with QB4OLAP
- Query QB4OLAP cubes

Outline

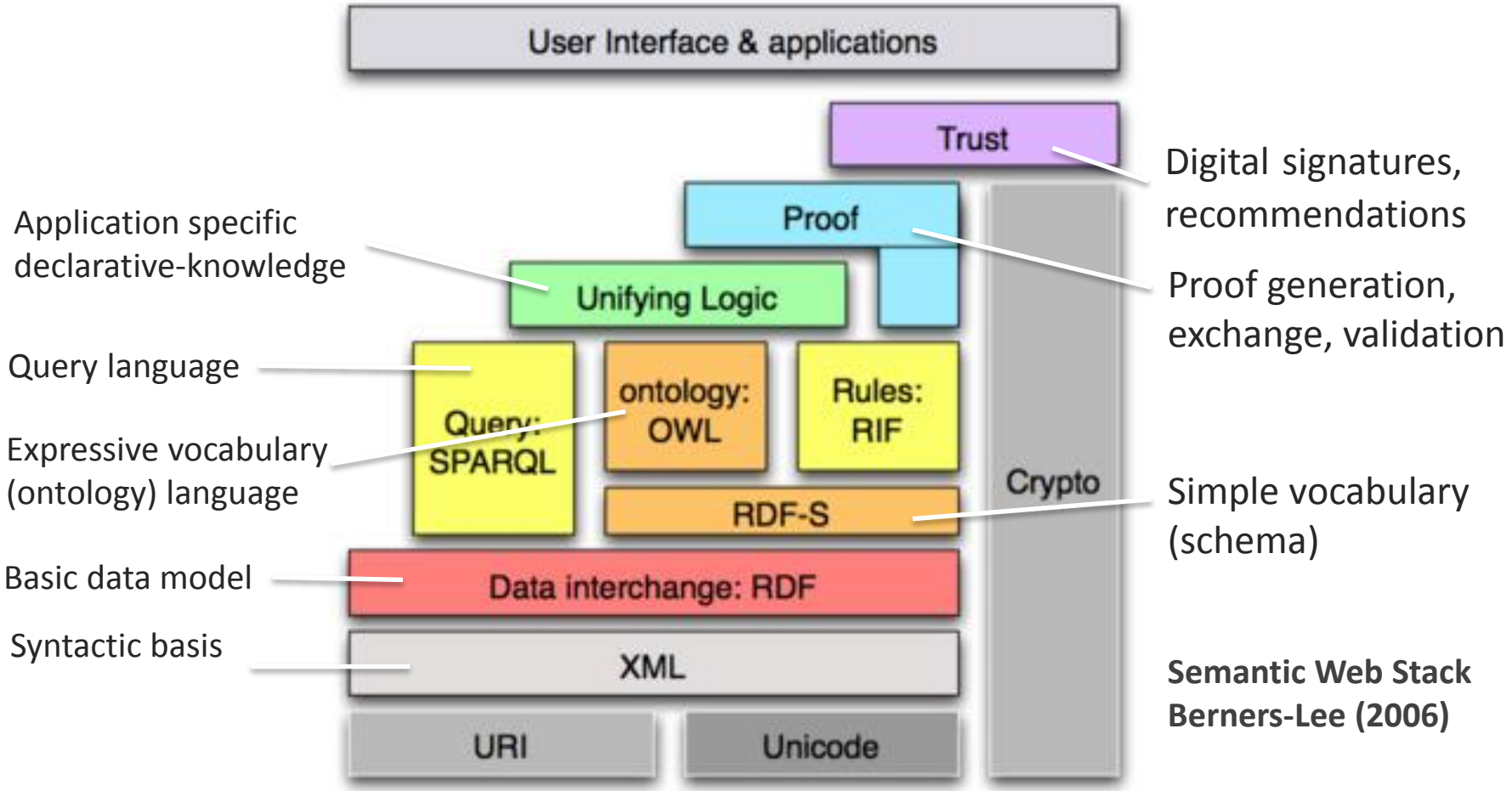
- The Semantic Web
- **RDF and SPARQL Basics**
- Vocabularies for OLAP on the SW: QB and QB4OLAP
- Modeling Data Cubes on the Semantic Web using QB4OLAP
- Querying Data Cubes on the Semantic Web
- Summary

Semantics on the Web



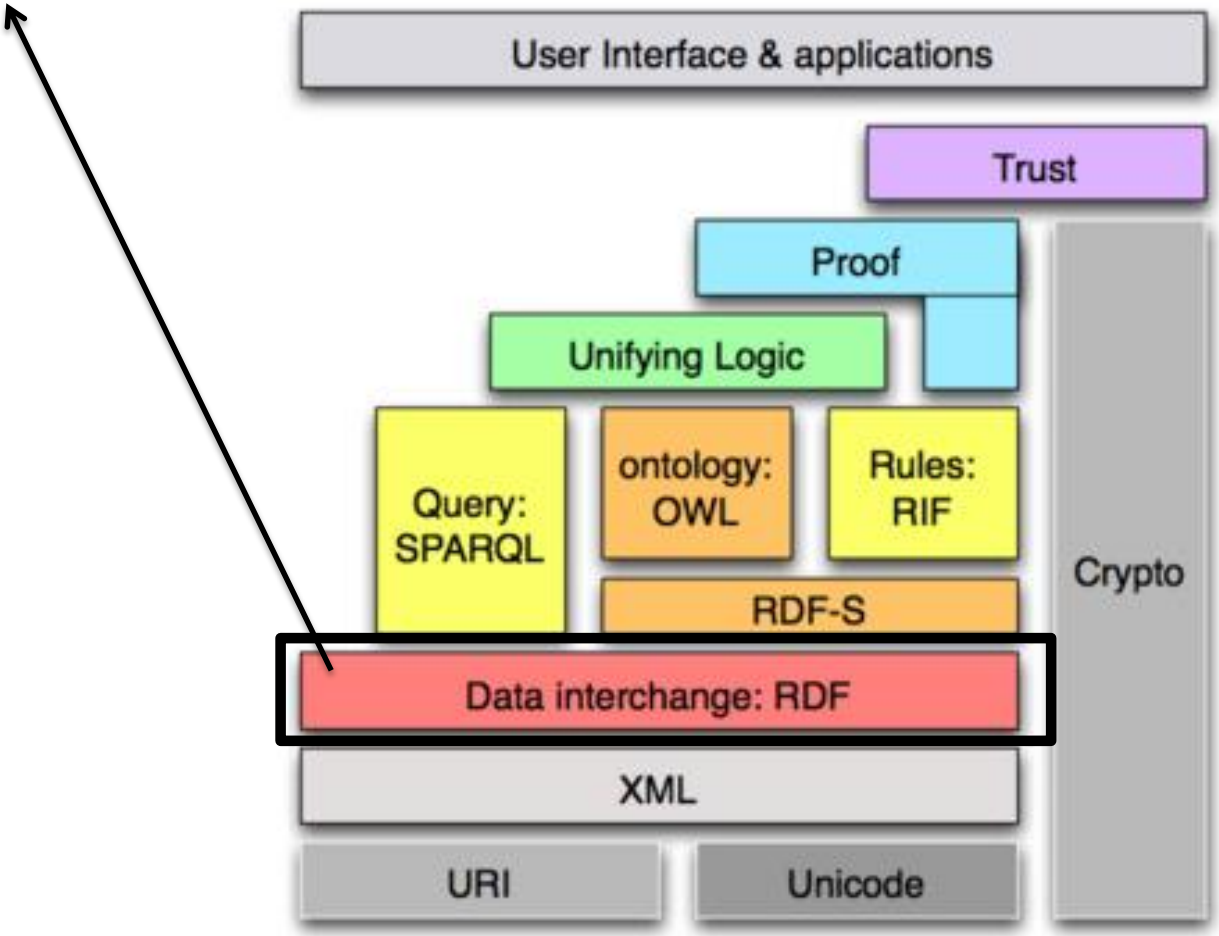
Semantic Web Stack
Berners-Lee (2006)

Semantics on the Web



Semantics on the Web

RDF – Resource Description Framework



Semantic Web Stack
Berners-Lee (2006)

RDF: Resource Description Framework

- RDF is the basic layer of the Semantic Web stack
- A data model based on graphs, that allows to describe resources and represent relationships between them
- Basic building block: RDF triple
  - **Subject** – a resource, which may be identified with an IRI
  - **Predicate** – an IRI-identified specification of a relationship
  - **Object** – a resource or literal to which the subject is related

RDF: Resource Description Framework

Example

<http://musicbrainz.org/artist/b10bbbfc-cf9e-42e0-be17-e2c3e1d2600d#_>
    <http://www.w3.org/2002/07/owl#sameAs>
        <http://dbpedia.org/resource/The_Beatles>.

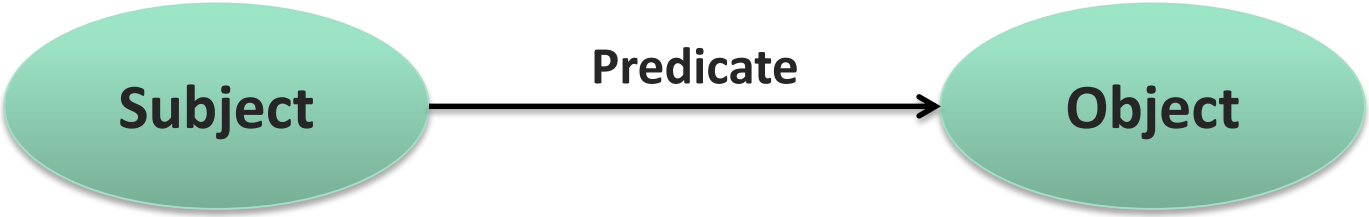
IRI - resource

<http://musicbrainz.org/artist/b10bbbfccf9e-42e0-be17-e2c3e1d2600d#_>
    <http://xmlns.com/foaf/0.1/name>
        "The Beatles" .

Literals

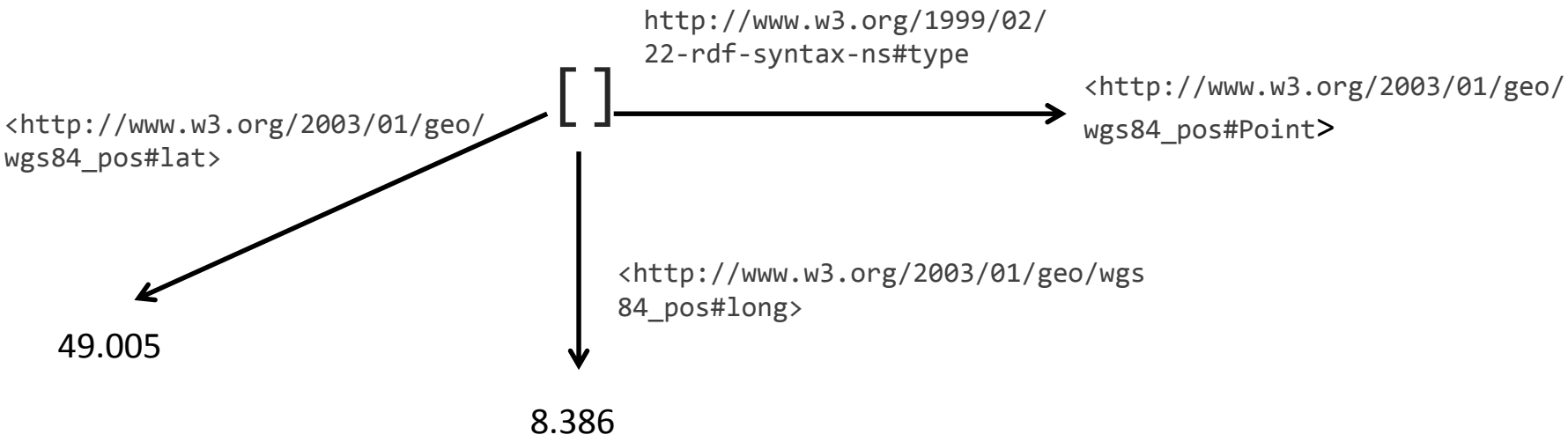
RDF Graphs

- Set of RDF assertions: Labeled directed graph
  - **Resources** – Subjects and objects are nodes of the graph
  - **Predicates** – A label for an arc, connecting subject to object



RDF Blank Nodes

- RDF graphs can also contain unidentified resources, called *blank nodes*:



RDF Turtle

- Turtle: Syntax to make RDF more readable
- Basically a textual representation of an RDF graph
- Many IRIs share same basis, then we use **prefixes**:

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

  @prefix rdfs:<http://www.w3.org/2000/01/rdf-schema#>.

  @prefix owl:<http://www.w3.org/2002/07/owl#>.

  @prefix mo:<http://purl.org/ontology/mo/>.

  @prefix dbpedia:<http://dbpedia.org/resource/>.

  We can use a unique base:

  @base <http://musicbrainz.org/>.

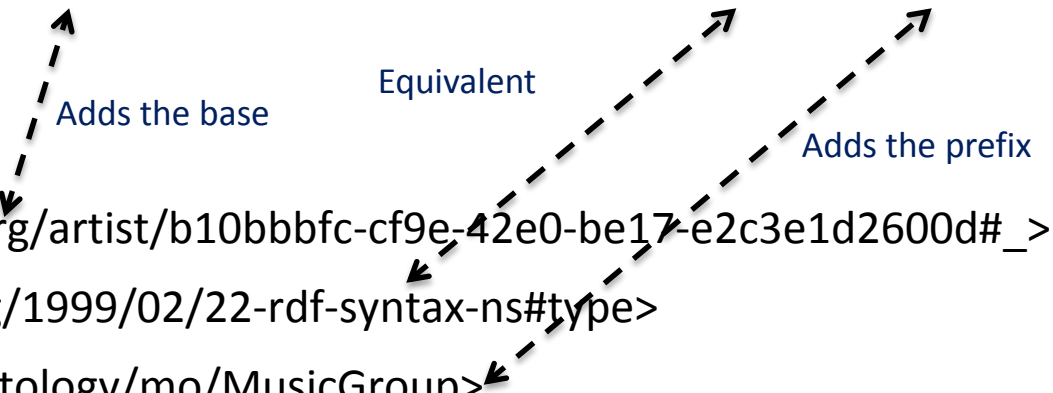
RDF Turtle

- Shorthands:

```
@base <http://musicbrainz.org/>.
@prefix mo:<http://purl.org/ontology/mo/>.

<artist/b10bbbfc-cf9e-42e0-be17-e2c3e1d2600d#_> a  mo:MusicGroup.
```

Is equivalent to:

```
<http://musicbrainz.org/artist/b10bbbfc-cf9e-42e0-be17-e2c3e1d2600d#_>
  <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
    <http://purl.org/ontology/mo/MusicGroup>.
```



RDF Turtle

- When multiple statements apply to **same subject** they can be abbreviated:

```
<artist/b10bbbfc-cf9e-42e0-be17-e2c3e1d2600d#_>
    rdfs:label "The Beatles";
    owl:sameAs dbpedia:The_Beatles ,
            <http://www.bbc.co.uk/music/artists/b10bbbfc-
                    cf9e-42e0-be17-e2c3e1d2600d#artist> .
```

Same subject

Same subject & predicate

RDF Turtle

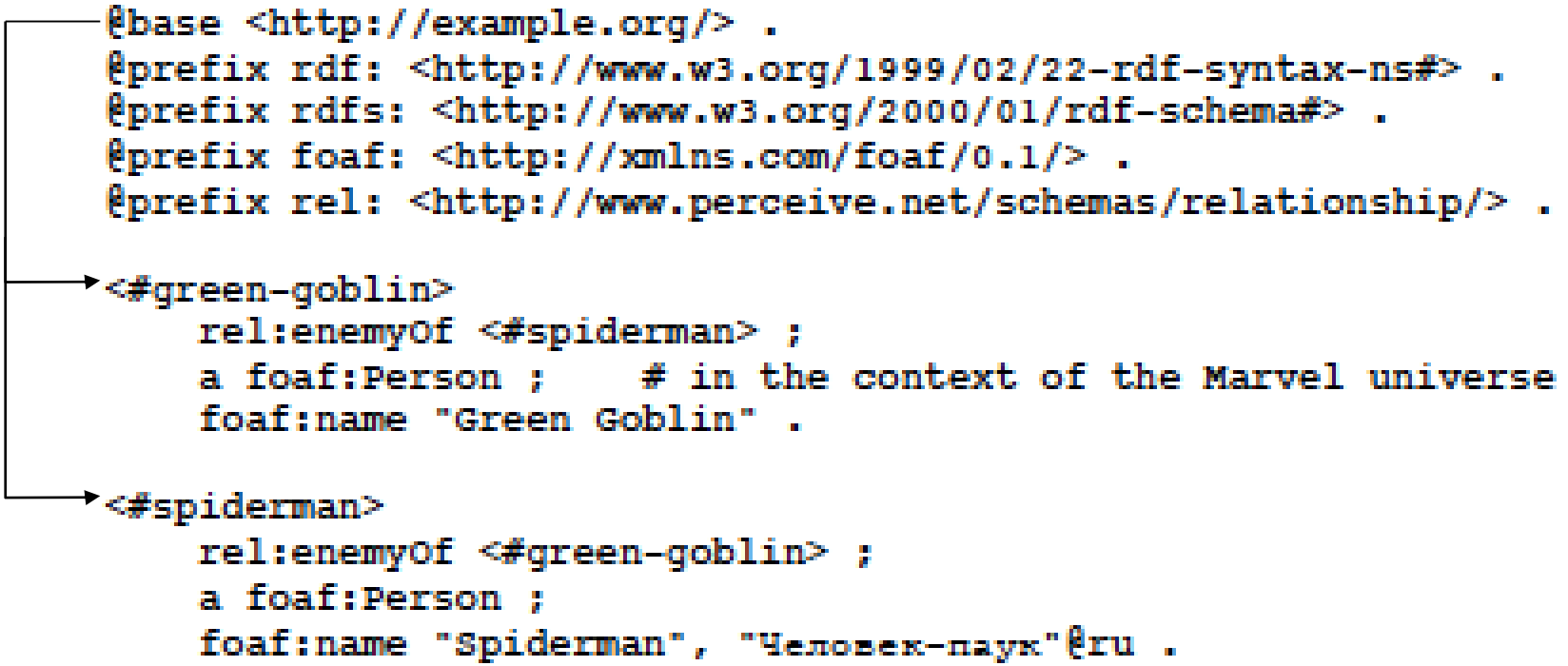
- Data types and language tags for literals:

```
<recording/5098d0a8-d3c3-424e-9367-1f2610724410#_> a  mo:Signal;
        rdfs:label "All You Need Is Love" ;
        mo:duration "PT3M48S"^^xsd:duration .

dbpedia:The_Beatles
        dbpedia-owl:abstract
              "The Beatles were an English rock band formed (…) "@en,
              "The Beatles waren eine britische Rockband in den (…) "@de .
```

- Also xsd:integer, xsd:decimal, xsd:double, xsd:Boolean, etc.

Example

- Turtle document describing the relationship between Green Goblin and Spiderman (from http://www.w3.org/TR/turtle/)



```
@base <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rel: <http://www.perceive.net/schemas/relationship/> .

<#green-goblin>
    rel:enemyOf <#spiderman> ;
    a foaf:Person ;    # in the context of the Marvel universe
    foaf:name "Green Goblin" .

<#spiderman>
    rel:enemyOf <#green-goblin> ;
    a foaf:Person ;
    foaf:name "Spiderman", "Человек-паук"@ru .
```

IRIs

- Writing IRIs in Turtle

```
# A triple with all absolute IRIs
<http://one.example/subject1> <http://one.example/predicate1> <http://one.example/object1> .

@base <http://one.example/> .
<subject2> <predicate2> <object2> .     # relative IRIs, e.g. http://one.example/subject2

BASE <http://one.example/>
<subject2> <predicate2> <object2> .     # relative IRIs, e.g. http://one.example/subject2

@prefix p: <http://two.example/> .
p:subject3 p:predicate3 p:object3 .     # prefixed name, e.g. http://two.example/subject3

PREFIX p: <http://two.example/>
p:subject3 p:predicate3 p:object3 .     # prefixed name, e.g. http://two.example/subject3

@prefix p: <path/> .                    # prefix p: now stands for http://one.example/path/
p:subject4 p:predicate4 p:object4 .     # prefixed name, e.g. http://one.example/path/subject4

@prefix : <http://another.example/> .    # empty prefix
:subject5 :predicate5 :object5 .        # prefixed name, e.g. http://another.example/subject5

:subject6 a :subject7 .                 # same as :subject6 <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> :subject7 .
```

Turtle: Blank Nodes

- Expressed as "_:" followed by a blank node label (a series of name characters)

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

_:alice foaf:knows _:bob .
_:bob foaf:knows _:alice .
```

- Also:

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

# Someone knows someone else, who has the name "Bob".
[] foaf:knows [ foaf:name "Bob" ] .
```

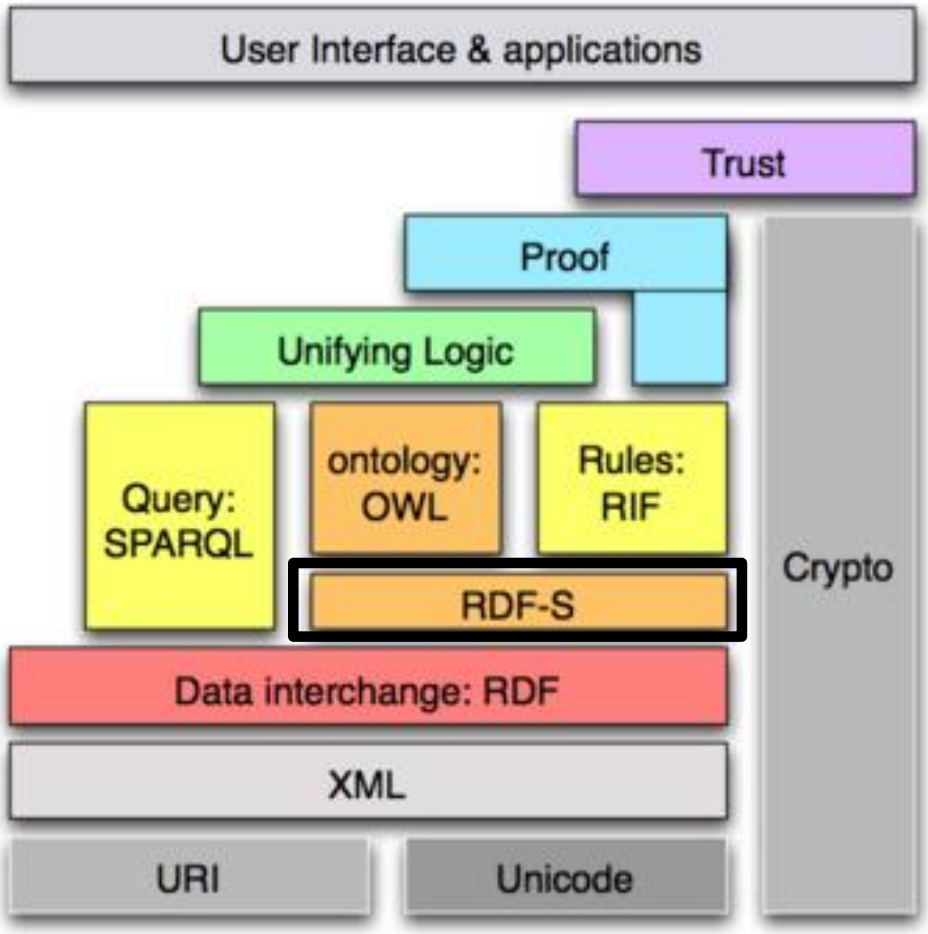
Turtle: Blank Nodes

- The use of predicateObjectList within a blankNodePropertyList is common for representing a series of properties of a node.

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

[ foaf:name "Alice" ] foaf:knows [
    foaf:name "Bob" ;
    foaf:knows [
        foaf:name "Eve" ] ;
    foaf:mbox <bob@example.com> ] .
```

```
_:a <http://xmlns.com/foaf/0.1/name> "Alice" .
_:a <http://xmlns.com/foaf/0.1/knows> _:b .
_:b <http://xmlns.com/foaf/0.1/name> "Bob" .
_:b <http://xmlns.com/foaf/0.1/knows> _:c .
_:c <http://xmlns.com/foaf/0.1/name> "Eve" .
_:b <http://xmlns.com/foaf/0.1/mbox> <bob@example.com> .
```

Semantics on the Web



Semantic Web Stack
Berners-Lee (2006)

RDF Schema (RDFS)

- Language for two tasks w.r.t. the RDF data model:
  - **Expectation** – nominate:
    - the 'types', i.e., *classes*, of things we might make assertions about
    - the *properties* we might apply, as predicates in these assertions, to capture their relationships.
  - **Inference** – given a set of assertions, using these classes and properties, specify what should be inferred about assertions that are *implicitly* made.

RDF Schema (RDFS)

- rdf:Property **-** Class of RDF properties. Example:

  mo:member - Indicates a member of a musical group.

- rdfs:domain **-** States that any resource that has a given property is an instance of one or more classes.

  mo:member rdfs:domain mo:MusicGroup .

- rdfs:range **-** States that the values of a property are instances of one or more classes.

  mo:member rdfs:range foaf:Agent .

RDF Schema (RDFS)

Schema

```
mo:MusicGroup rdfs:subClassOf
foaf:Group .
```

We *expect* to use this vocabulary to make assertions about music groups.

Existing fact

```
<artist/b10bbbfc-cf9e-42e0-be17-
e2c3e1d2600d#_> rdf:type
mo:MusicGroup .
```

Having made such an assertion...

Inferred **fact**

```
<artist/b10bbbfc-cf9e-42e0-be17-
e2c3e1d2600d#_> rdf:type
foaf:Group .
```

Inferences can be drawn that we did not explicitly make

RDF Schema (RDFS)

Resources and predicates with (limited) **inferences**:

rdfs:Resource

rdfs:Literal, rdfs:Datatype

rdfs:Class, rdfs:subClassOf

rdfs:subPropertyOf

rdfs:range, rdfs:domain

Some predicates with **NO inferences**:

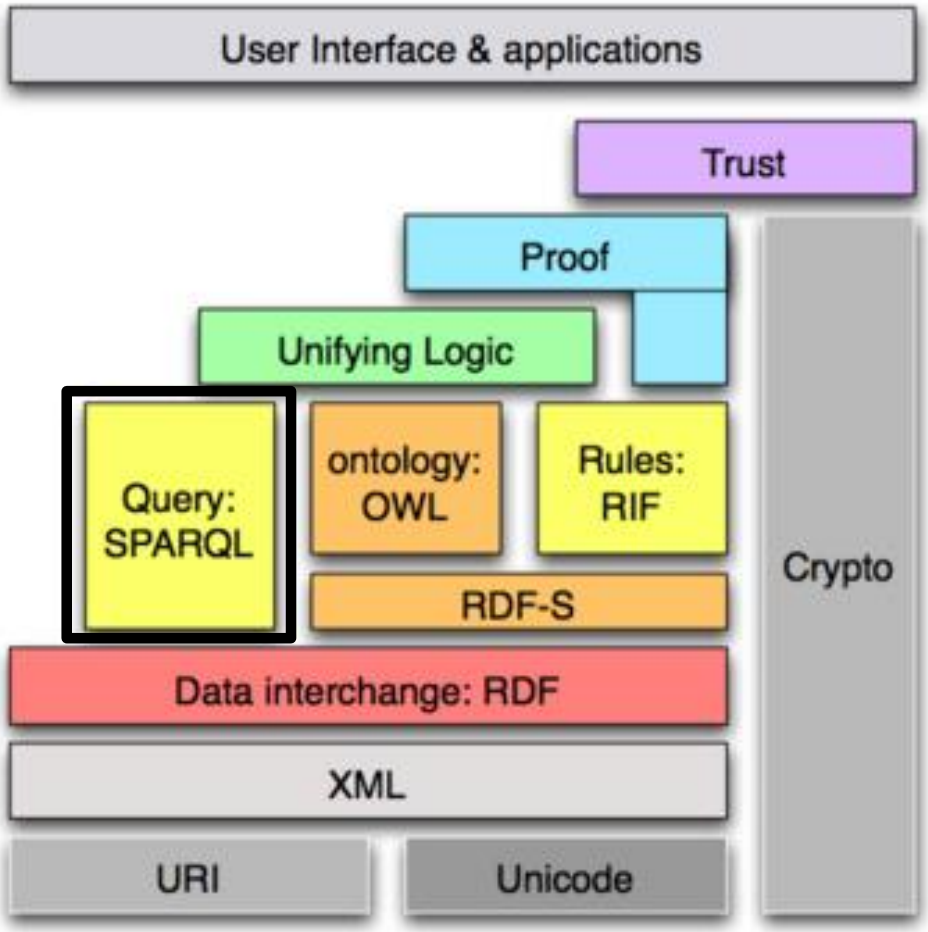
rdfs:comment

rdfs:label

rdfs:seeAlso

rdfs:isDefinedBy

Semantics on the Web



Semantic Web Stack
Berners-Lee (2006)

SPARQL

- SPARQL Query
  - Declarative query language for RDF data
  - http://www.w3.org/TR/rdf-sparql-query/
- SPARQL Algebra
  - The semantics of a SPARQL query execution
  - http://www.w3.org/2001/sw/DataAccess/rq23/rq24-algebra.html
- SPARQL Update
  - Declarative manipulation language for RDF data
  - http://www.w3.org/TR/sparql11-update/
- SPARQL Protocol
  - Standard for communication between SPARQL services and clients
  - http://www.w3.org/TR/sparql11-protocol/

SPARQL

- SPARQL: Simple Protocol and RDF Query Language
- Query language designed to use a syntax similar to SQL
- Four query forms:

  - **SELECT** returns variables and their bindings
  - **CONSTRUCT** returns an RDF graph specified by a graph template
  - **ASK** tests whether or not a query pattern has a solution, returns yes/no
  - **DESCRIBE** returns an RDF graph containing RDF data about resources

SPARQL

- The syntax of a **SELECT** query is as follows:

  - **SELECT** tells which components of the matches should be returned
  - **FROM** (optional) indicates the sources for the data
  - **WHERE** defines patterns to match against the data
  - **ORDER BY** defines how to sort the selected matches

SPARQL

- **RDF triple:** Basic building block, of the form: subject, predicate, object
  Example:

```
dbpedia:The_Beatles foaf:name "The Beatles" .
```

- **RDF triple pattern:** Contains one or more variables.

  Examples:

```
dbpedia:The_Beatles foaf:made ?album .
?album mo:track ?track .
?album ?p ?o .
```

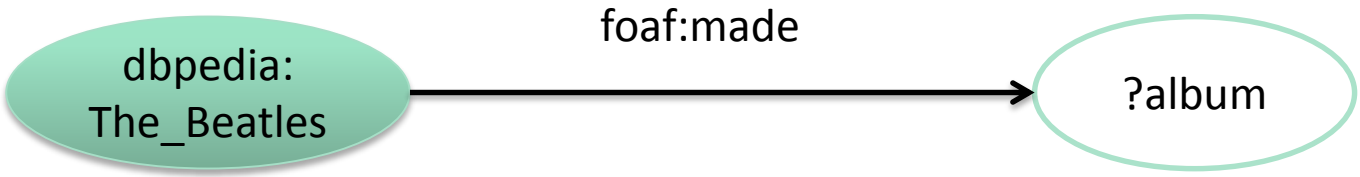
- **RDF quad pattern:** Contains the graph name: IRI or variable
- Examples:

```
GRAPH <:g> {:s :p :o .}
GRAPH ?g {dbpedia:The_Beatles foaf:name ?o.}
```

SPARQL Query

Main idea: **Pattern matching**

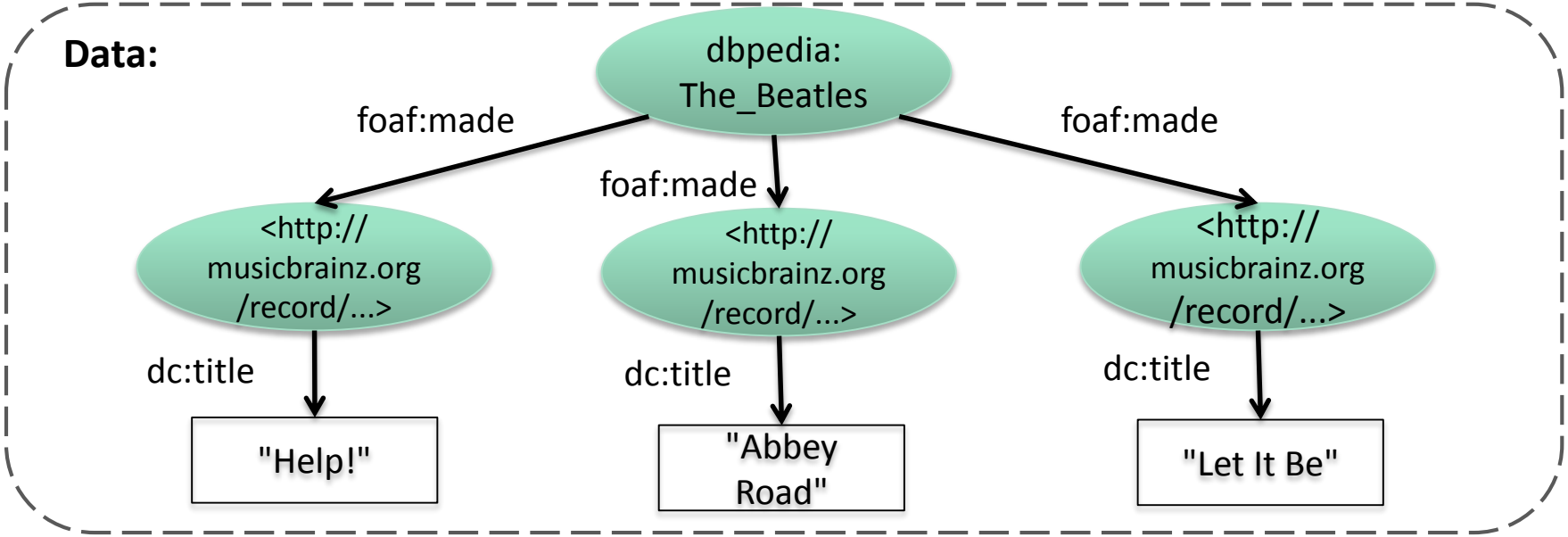
- Queries describe sub-graphs of the queried graph
- **Graph patterns** are RDF graphs specified in Turtle syntax, which contain variables (prefixed by either “?” or “$”)



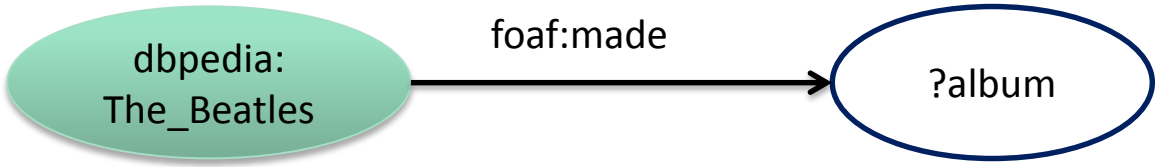
- Sub-graphs that match the graph patterns yield a **result**

SPARQL Query

Query: *names of the albums recorded by The Beatles*



Graph pattern



Results

?album
<http://musicbrainz.org...>
<http://musicbrainz.org...>
<http://musicbrainz.org...>

SPARQL Query

```
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX mo: <http://purl.org/ontology/mo/>

SELECT ?title
FROM <http://musicbrainz.org/20130302>
WHERE {
   dbpedia:The_Beatles foaf:made ?album .
   ?album a mo:Record ; dc:title ?title .
}
ORDER BY ?title
```

- **Prefix definitions**
- Unlike in Turtle, final period not used

SPARQL Query

```
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX mo: <http://purl.org/ontology/mo/>

SELECT ?title
FROM <http://musicbrainz.org/20130302>
WHERE {
   dbpedia:The_Beatles foaf:made ?album .
   ?album a mo:Record ; dc:title ?title
}
ORDER BY ?title
```

- **Data set specification:**
- FROM or FROM NAMED (optional)
- Indicates the sources for the data against which to find matches

SPARQL Query

```
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX mo: <http://purl.org/ontology/mo/>

SELECT ?title
FROM <http://musicbrainz.org/20130302>
WHERE {
   dbpedia:The_Beatles foaf:made ?album .
   ?album a mo:Record ; dc:title ?title
}
ORDER BY ?title
```

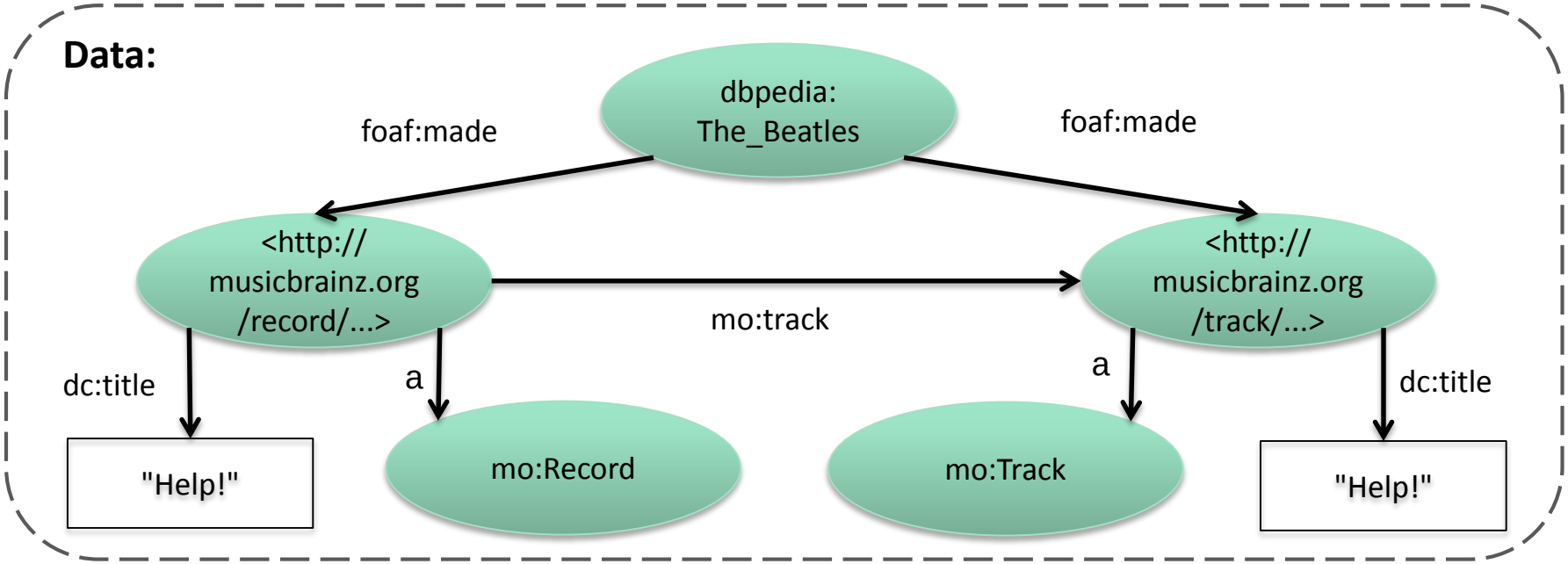
- **Solution modifier:**
- Modifies the result set: ORDER BY, LIMIT, OFFSET

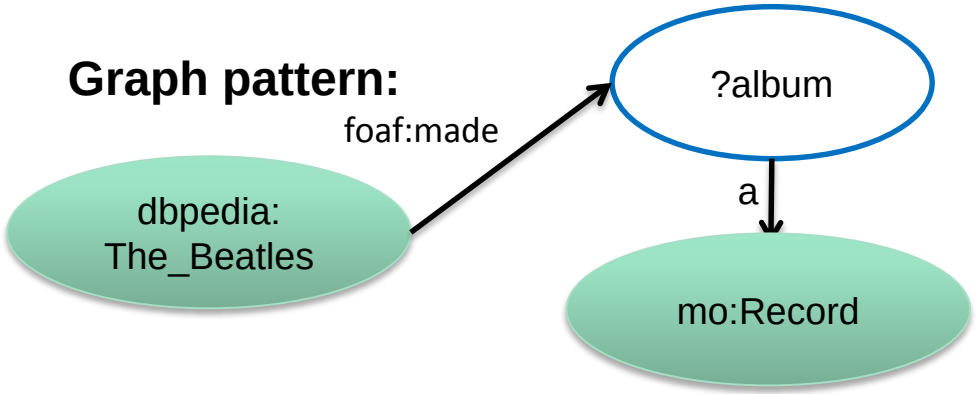
SPARQL

- **Another example:** *names of the albums and tracks recorded by The Beatles*

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX mo: <http://purl.org/ontology/mo/>

SELECT ?album_name ?track_title
WHERE {
 <http://musicbrainz.org/artist/b10bbbfc-cf9e-42e0-be17-e2c3e1d2600d#_>
          foaf:made ?album .
 ?album dc:title ?album_name ;
          mo:track ?track .
 ?track dc:title ?track_title . }
```

SPARQL Query





Results:

?album
<http://musicbrainz.org...>

SPARQL

SQL	SPARQL
Based on relations (tables)	Based on labeled directed graphs
The relations (tables) to be matched over should be indicated	Assumes a default graph (The FROM clause populates this with specific identified subgraphs)
(Retrieval) queries produce a relation from a relation	SELECT queries produce a relation from a graph CONSTRUCT queries produce a graph from a graph ASK tests whether or not a query pattern has a solution. Returns true/false. DESCRIBE returns a single RDF graph containing RDF data about resource

Query Form: SELECT

- Different types of filters and functions may be used

Filter: Comparison and logical operators

Query: *Albums and tracks recorded by 'The Beatles', where the duration of the song is more than 300 secs. and no longer than 400 secs.*

```
SELECT ?album_name ?track_title ?date ?duration
WHERE {
   dbpedia:The_Beatles foaf:made ?album .
   ?album  dc:title ?album_name ;
           mo:track ?track .
   ?track   dc:title ?track_title ;
           mo:duration ?duration.
   FILTER (?duration > 300000 && ?duration < 400000) }
```

Query Form: SELECT

- Aggregates
  - Calculate aggregate values: COUNT, SUM, MIN, MAX, AVG,…
  - GROUP BY operator
  - Prune at group level using HAVING

Query: *Retrieve the duration of the albums recorded by 'The Beatles' if its is greater than 1 hour.*

```
SELECT ?album (SUM(?track_duration) AS ?album_duration)
WHERE {
   dbpedia:The_Beatles foaf:made ?album .
   ?album mo:track ?track .
   ?track mo:duration ?track_duration .}
GROUP BY ?album
HAVING (SUM(?track_duration) > 3600000)
```

Query Form: SELECT

- Duplicate elimination

Query: *Retrieve the name of the albums recorded by 'The Beatles' which have at least two different songs.*

```
SELECT DISTINCT ?album_name
WHERE {
   dbpedia:The_Beatles foaf:made ?album .
   ?album dc:title ?album_name ;
              mo:track ?track1 ;
              mo:track ?track2 .
    FILTER (?track1 != ?track2) }
```

Result:

?album
"Revolver"
"Sessions"
"Abbey Road"

Query Form: SELECT

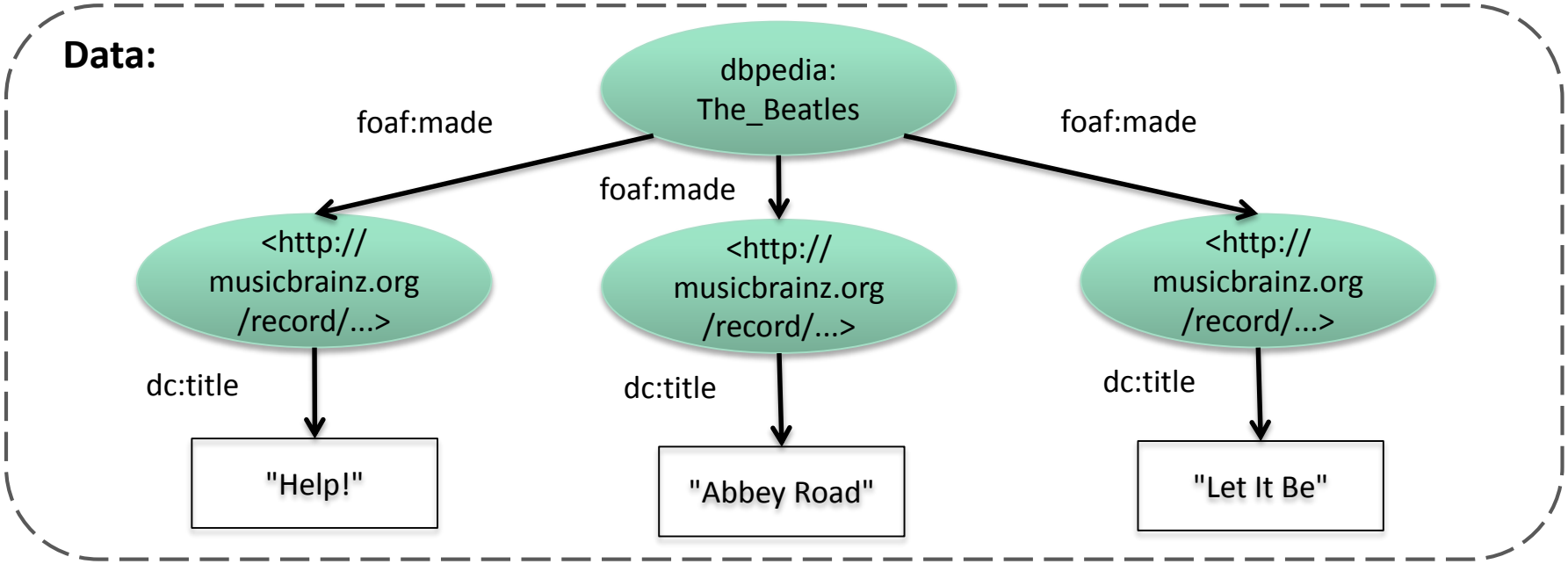
- Optional Graph Pattern
- OPTIONAL clause encloses the optional parts
- Addresses incompleteness in RDF graphs
- If the optional part does not match, it creates no bindings but does not eliminate the solution

Query: *List artists who made an album, and their pictures, if they exist*

```
SELECT {
   ?artist  ?picture }
WHERE {
   ?artist foaf:made ?album .
   OPTIONAL {?artist foaf:depiction ?picture .}}
```

- If there is no picture, only the artist is listed

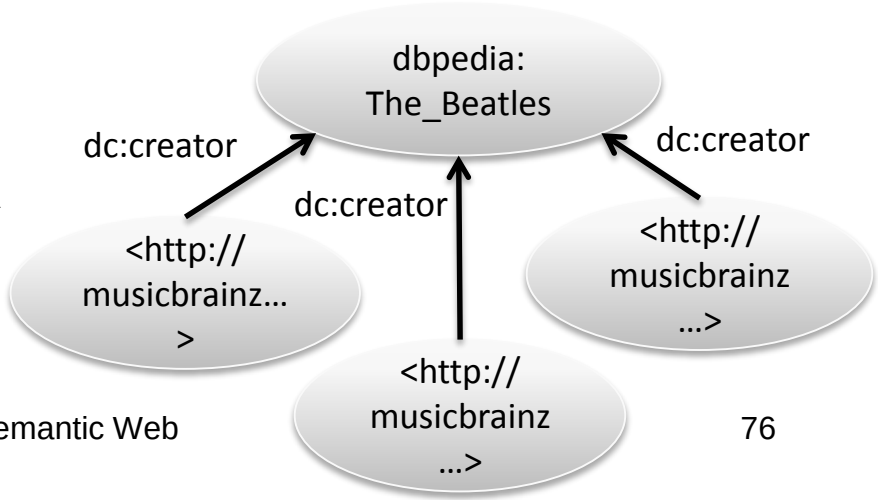
Query Form: CONSTRUCT



Query:

```
CONSTRUCT {
  ?album dc:creator dbpedia:The_Beatles .}
WHERE {
  dbpedia:The_Beatles foaf:made ?album .}
```

Result



Query Form: CONSTRUCT

- Subsets of results
- Can combine the query with **solution modifiers** (ORDER BY, LIMIT, OFFSET)

Query: *Produce the dc:creator descriptions for the 10 most recent albums and their tracks recorded by 'The Beatles'.*

```
CONSTRUCT {
  ?album dc:creator dbpedia:The_Beatles .
  ?track   dc:creator dbpedia:The_Beatles . }
WHERE {
  dbpedia:The_Beatles foaf:made ?album .
  ?album mo:track ?track ;
                dc:date   ?date . }
  ORDER BY DESC(?date)
  LIMIT 10
```

Query Form: CONSTRUCT

- Regular expression filters over strings

Query: *Create the dc:creator descriptions of the albums recorded by 'The Beatles' whose title contains the word 'love'.*

```
CONSTRUCT {?album dc:creator dbpedia:The_Beatles .}
WHERE {
   dbpedia:The_Beatles foaf:made ?album .
   ?album dc:title ?album_name .
   FILTER (REGEX(?album_name, ".*love.*", i)) }
```

i: case insensitive indicator (optional)

Query Form: CONSTRUCT

- Optional Graph Pattern
- OPTIONAL clause encloses the optional parts
- Addresses incompleteness in RDF graphs
- When the optional part does not match, no triple is created

Query: *Create the dc:creator and dc:depicts descriptions of artists.*

```
CONSTRUCT {
   ?album  dc:creator ?artist .
   ?picture dc:depicts ?artist . }
WHERE {
   ?artist foaf:made ?album .
   OPTIONAL {?artist foaf:depiction ?picture .}}
```

Query Form: CONSTRUCT

- Sub-queries and Aggregate Values
- To combine the CONSTRUCT query form with aggregate values, a sub-query should be created inside the WHERE clause

Query: *Materialize the duration of the albums recorded by 'The Beatles'.*

```
CONSTRUCT {?album mo:duration ?album_duration . }
WHERE {
  SELECT ?album (SUM(?track_duration) AS ?album_duration)
  WHERE {
            dbpedia:The_Beatles foaf:made ?album .
            ?album mo:track ?track .
            ?track mo:duration ?track_duration . }
            GROUP BY ?album
            HAVING (SUM(?track_duration) > 3600000)}
```

Query Form: ASK

- Namespaces are added with the ‘PREFIX’ directive
- Statement patterns that make up the graph are specified between brackets (“{}”)

Query: *Is Paul McCartney member of ‘The Beatles’?*

```
ASK WHERE { dbpedia:The_Beatles mo:member
                dbpedia: Paul_McCartney.}
```

Result:

true

Query: *Is Elvis Presley member of ‘The Beatles’?*

```
ASK WHERE {dbpedia:The_Beatles  mo:member
                dbpedia:Elvis_Presley.}
```

Result:

false

Query Form: DESCRIBE

Takes the resources in the solution, and provides information about them as RDF statements. They can be identified by:

- Specifying explicit IRIs

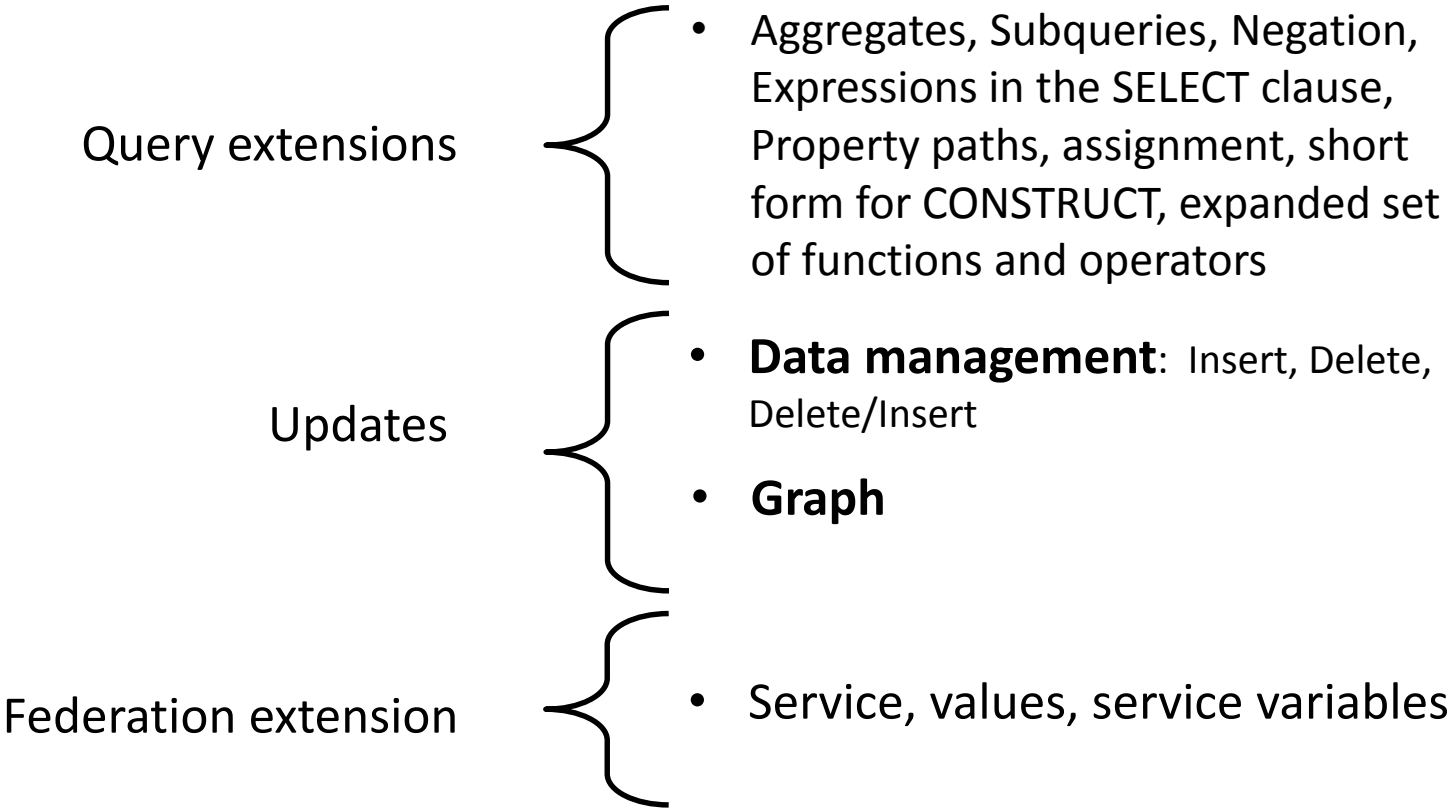
```
PREFIX dbpedia: <http://dbpedia.org/resource/>
DESCRIBE dbpedia:Paul_McCartney
```

- Bindings of variables in the WHERE clause

```
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX mo: <http://purl.org/ontology/mo/>
DESCRIBE ?member
WHERE {
dbpedia:The_Beatles mo:member ?member .}
```

SPARQL 1.1

- **SPARQL 1.0** only allows accessing the data (i.e., **querying**)
- **SPARQL 1.1** introduces:



SPARQL: Updates

- SPARQL 1.1 provides **graph update operations**:
  - **INSERT DATA:** adds explicit triples
  - **DELETE DATA:** removes explicit triples
  - **DELETE/INSERT WHERE:** updates based on triples specified in the WHERE clause (as in SELECT and CONSTRUCT).
  - **LOAD:** reads the content of a document into a graph
  - **COPY/MOVE/APPEND:** manipulates at named graph level
  - **CLEAR/DROP:** removes all triples in one or more graph

SPARQL: Updates

Insert *the following albums recorded by The Beatles into the graph* <http://myFavGroups/The_Beatles>

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX foaf: < http: //xmlns.com/foaf/0.1/>
INSERT DATA { GRAPH < http://myFavGroups/The_Beatles> {
<http://musicbrainz.org/artist/b10bbbfc-cf9e-42e0-be17-e2c3e1d2600d#_>
foaf:made
<http://musicbrainz.org/release/3a685770-7326-34fc-9f18-e5f5626f3dc5#_> ,
<http://musicbrainz.org/release/cb6f8798-d51e-4fa5-a4d1-2c0602bfe1b6#_>  .
<http://musicbrainz.org/release/3a685770-7326-34fc-9f18-e5f5626f3dc5#_>
   dc:title "Please Please Me”  .
< http://musicbrainz.org/release/cb6f8798-d51e-4fa5-a4d1-2c0602bfe1b6#_ >
   dc:title "Something New” . }
```

SPARQL: Updates

Delete *all the information about the album Casualities of The Beatles.*

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
DELETE { ?album ?predicate ?object . }
WHERE {
<http://musicbrainz.org/artist/b10bbbfc-cf9e-42e0-be17-e2c3e1d2600d#_>
      foaf:made   ?album .
?album dc:title "Casualities";
      ?predicate ?object . }
```

Outline

- The Semantic Web
- RDF and SPARQL Basics
- **Vocabularies for OLAP on the SW: QB and QB4OLAP**
- Modeling Data Cubes on the Semantic Web using QB4OLAP
- Querying Data Cubes on the Semantic Web
- Summary

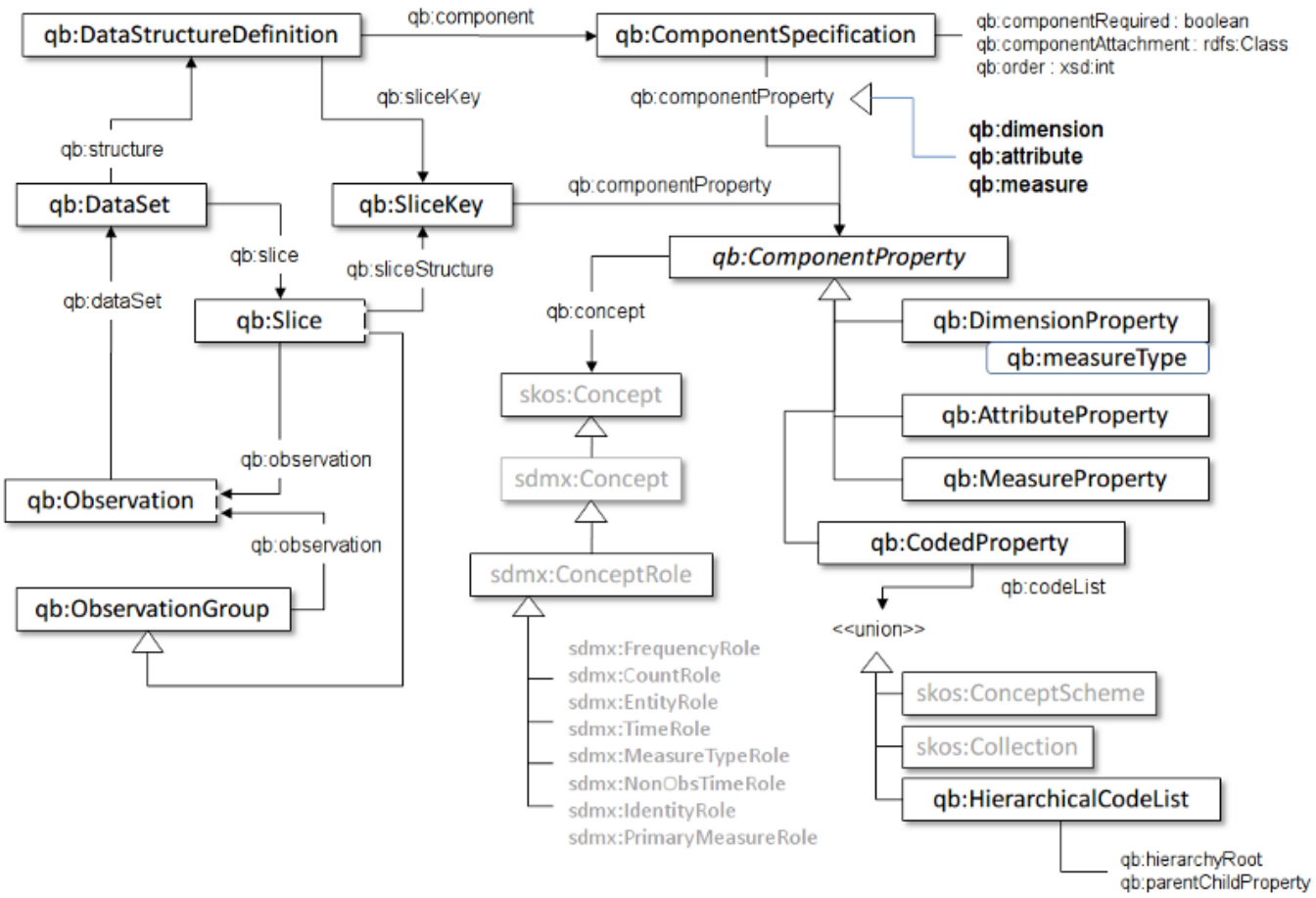
Let us recall the problem...

- Publish MD Linked data
- Represent multidimensional data using SW standards
  - Publish and share multidimensional schemas and data instances
- Requirements:
  - Alignment with Linked Data and Linked Open Data principles
  - Reuse existing vocabularies
  - Represent the main features of the MD model
- Standards:
  - The Data Cube vocabulary (QB) is the current standard

The RDF Data Cube (QB) vocabulary

- W3C Recommendation since 2014
- Oriented to publish statistical data
- Based on the **SDMX model** (http://sdmx.org/)
  - Statistical Data and Metadata Exchange
  - An initiative to foster standards for exchanging statistical information
  - Data set: a set of observations sharing dimensionality
  - Observation: A point in a multidimensional space mapped to a measure
  - A dimension classifies statistical series
  - Primary Measure: the phenomenon to be measured in the dataset
  - Structural metadata: Data Structure Definition

QB: RDF Data Cube vocabulary



The RDF Data Cube vocabulary

- Schema of a data set: specified by the **Data Structure Definition** (DSD)
  - Class: `qb:DataStructureDefinition`
  - Dataset: A set of Observations with the same dimensionality
  - Components of the dataset defined in the DSD
- Defined by **component properties**
  - Instances of the class: `qb:ComponentProperty`
- Basic element: an **observation**
  - Instances of the class: `qb:Observation`
  - Compose the instances of `qb:DataSet`
  - Associated with a data set through the property `qb:dataSet`
  - Linked to dimensions and measures through instances of
    - `qb: DimensionProperty, qb:MeasureProperty, qb: AttributeProperty`
- To specify additional attributes
  - `qb:ComponentSpecification, e.g., qb:componentRequired, qb:order.`

Example: World Bank linked data

- Example: World Bank Linked Data (http://data.worldbank.org/indicator)
- Collection of indicators for countries, regions, etc.
  - Available in many formats
  - VoID file containing metadata for the data sets
  - SPARQL endpoint available
  - Example indicator: “Market capitalization of listed companies”
    - We will use this indicator

QB: Data structure definition for WBLD

```
@prefix qb: <http://purl.org/linked-data/cube#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
@prefix sdmx-measure: <http://purl.org/linked-data/sdmx/2009/measure#>.

<http://worldbank.270a.info/dataset/world-bank-indicators/structure>
        a qb:DataStructureDefinition ;
        qb:component  [ a qb:ComponentSpecification ;
                        qb:dimension  <http://worldbank.270a.info/property/indicator> ;
                        qb:order "1"^^xsd:int ],
                       [ a qb:ComponentSpecification ;
                        qb:dimension sdmx-dimension:refArea ;
                        qb:order "2"^^xsd:int ],
                      [ a qb:ComponentSpecification ;
                        qb:dimension sdmx-dimension:refPeriod ;
                        qb:order "3"^^xsd:int ],
                      [ a qb:ComponentSpecification ;
                        qb:measure sdmx-measure:obsValue ;
                        qb:order "4"^^xsd:int ] .
```

QB: Observations

@prefix qb: <http://purl.org/linked-data/cube#> .
@prefix property: <http://worldbank.270a.info/property/> .
@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
@prefix sdmx-measure: <http://purl.org/linked-data/sdmx/2009/measure#> .
@prefix indicator: <http://worldbank.270a.info/classification/indicator/>.
@prefix country: <http://worldbank.270a.info/classification/country/> .
@prefix year: <http://reference.data.gov.uk/id/year/> .

<http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/RS/2012>
a qb:Observation ;
qb:dataSet <http://worldbank.270a.info/dataset/CM.MKT.LCAP.CD> ;
property: indicator indicator:CM.MKT.LCAP.CD ;
sdmx-dimension: refArea country:RS ;
sdmx-dimension: refPeriod year:2012 ;
sdmx-measure: obsValue 7450560827.04874 ;
property:decimal 0 .

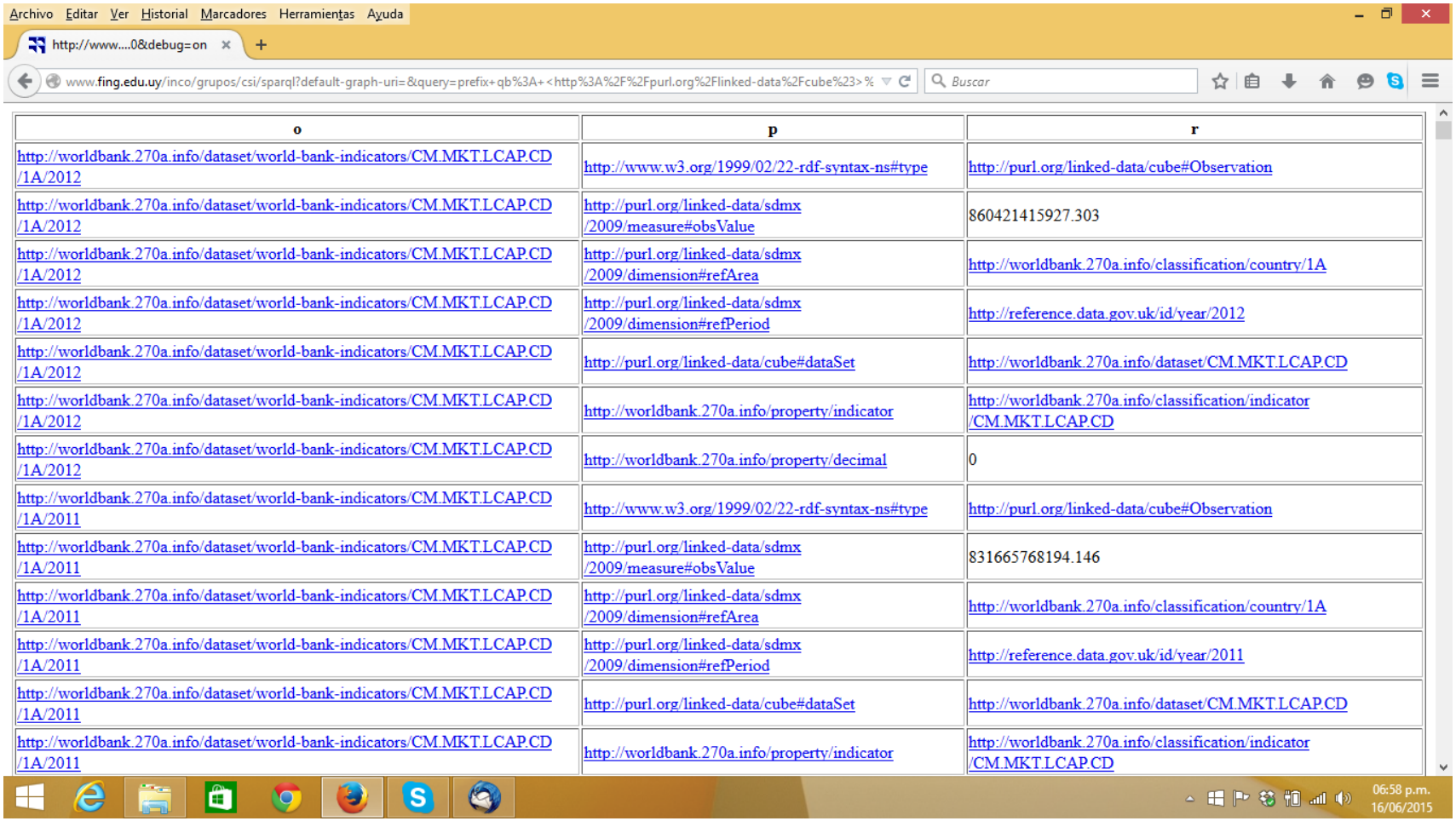
Observations

Endpoint:

http://www.fing.edu.uy/inco/grupos/csi/sparql

```
prefix qb: <http://purl.org/linked-data/cube#>

select *
FROM  <http://www.fing.edu.uy/inco/cubes/instances/wbld>
WHERE
 { ?o a qb:Observation;
       ?p ?r }
```

WBLD observations



Archivo Editar Ver Historial Marcadores Herramientas Ayuda

http://www....0&debug=on

www.fing.edu.uy/inco/grupos/csi/sparql?default-graph-uri=&query=prefix+qb%3A+<http%3A%2F%2Fpurl.org%2Flinked-data%2Fcube%23>%

Buscar

o	p	r
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2012	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://purl.org/linked-data/cube#Observation
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2012	http://purl.org/linked-data/sdmx/2009/measure#obsValue	860421415927.303
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2012	http://purl.org/linked-data/sdmx/2009/dimension#refArea	http://worldbank.270a.info/classification/country/1A
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2012	http://purl.org/linked-data/sdmx/2009/dimension#refPeriod	http://reference.data.gov.uk/id/year/2012
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2012	http://purl.org/linked-data/cube#dataSet	http://worldbank.270a.info/dataset/CM.MKT.LCAP.CD
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2012	http://worldbank.270a.info/property/indicator	http://worldbank.270a.info/classification/indicator/CM.MKT.LCAP.CD
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2012	http://worldbank.270a.info/property/decimal	0
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2011	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://purl.org/linked-data/cube#Observation
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2011	http://purl.org/linked-data/sdmx/2009/measure#obsValue	831665768194.146
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2011	http://purl.org/linked-data/sdmx/2009/dimension#refArea	http://worldbank.270a.info/classification/country/1A
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2011	http://purl.org/linked-data/sdmx/2009/dimension#refPeriod	http://reference.data.gov.uk/id/year/2011
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2011	http://purl.org/linked-data/cube#dataSet	http://worldbank.270a.info/dataset/CM.MKT.LCAP.CD
http://worldbank.270a.info/dataset/world-bank-indicators/CM.MKT.LCAP.CD/1A/2011	http://worldbank.270a.info/property/indicator	http://worldbank.270a.info/classification/indicator/CM.MKT.LCAP.CD

06:58 p.m.
16/06/2015

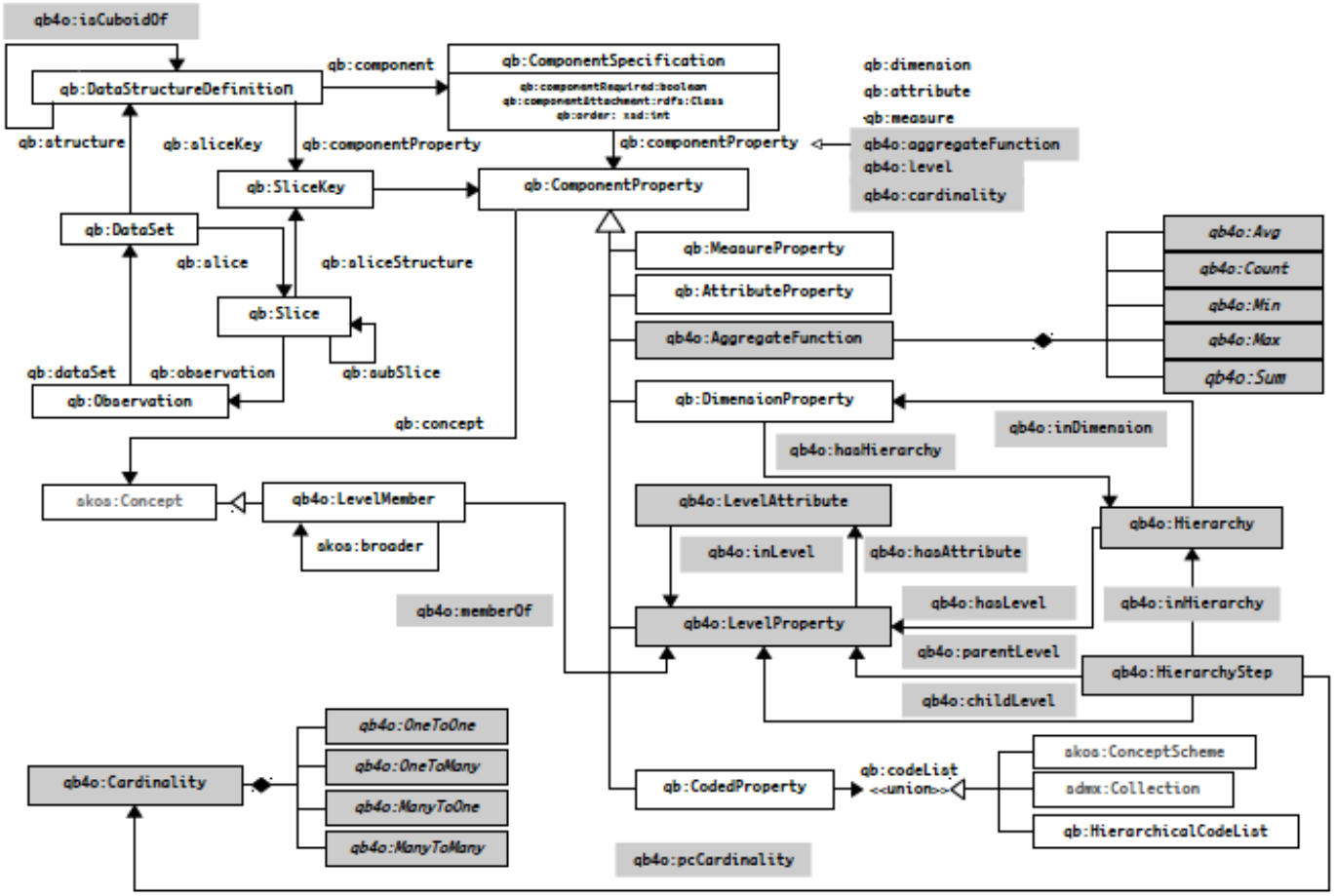
Limitations of QB

- No native support of dimensional hierarchies
  - Hierarchies are first-class citizens in OLAP
  - QB defines hierarchies at the *instance* level
  - No structural metadata for hierarchies, cardinalities, etc.
  - Can express “Argentina aggregates over South America”, but not that “Argentina is a country, South America is a continent, and countries aggregate over continents”
- No metadata to represent aggregate functions
  - Usual in OLAP tolos: Aassociate a measure with an aggregate function
- No support for descriptive attributes in levels
  - Useful for dicing using condition over attribute values

Limitations of QB (cont.)

- However, QB became a W3C recommendation in 2014
- Also, considerable amount of data already published using QB
- We need to:
  - Represent most common features of the MD model
  - Add metadata to implement OLAP using SPARQL
  - Provide compatibility with QB
    - **Extend** published data in QB, at the minimum cost
- We proposed the QB4OLAP vocabulary

QB4OLAP [1]



1. Joint work with Lorena Etcheverry, Universidad de la República, Uruguay

QB4OLAP: Data cube structure

- QB prefix: “qb:”, QB4OLAP prefix: “qb4o:”
- Characteristics:
  - In QB, observations are linked to **dimensions**
  - In QB4OLAP**,** observations (facts) are linked to **dimension levels**
  - Dimension levels are instances of the class **`qb4o:LevelProperty`**
  - Dimensions are still represented as instances of **`qb:DimensionProperty`**
  - **`qb4o:LevelProperty`** subclass of **`qb:ComponentProperty`** => can define the schema of the cube using **`qb:DataStructureDefinition`**
  - Instance of class **`qb4o:AggregateFunction`**: SUM, AVG, etc.
  - Measures and aggregate functions, linked through **`qb4o:aggregateFunction`**

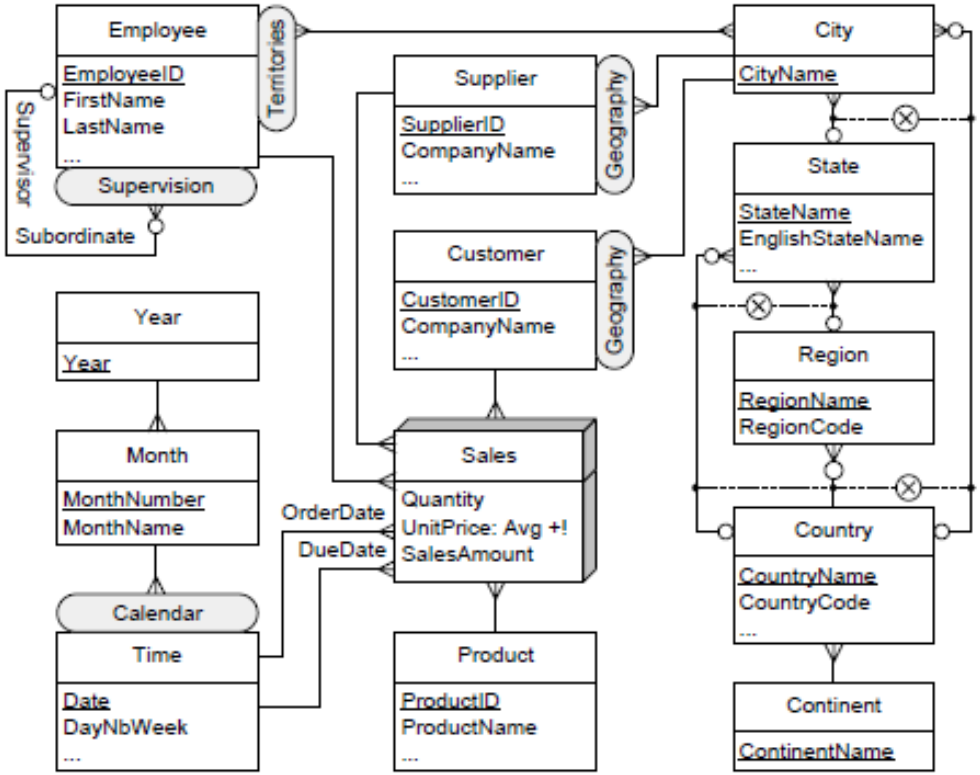
QB4OLAP: Dimension hierarchies

- Represented by the class **`qb4o:Hierarchy`**
- Associated with dimensions through properties:
  - **`qb4o:hasHierarchy and qb4o:inDimension`**
- To indicate that a level belongs to a hierarchy: **`qb4o:inHierarchy`**
- A level may be in many hierarchies, may have many parent levels
  - The class **`qb4o:HierarchyStep`** supports this concept
  - Hierarchy step defined by **`qb4o:parentLevel, qb4o:childLevel`**
  - Cardinality: **`qb4o:pcCardinality`** (default **`qb4o:ManyToOne`**)
- Next, we define the hierarchy schema for the WBLD dimensions

Outline

- The Semantic Web
- RDF and SPARQL Basics
- Vocabularies for OLAP on the SW: QB and QB4OLAP
- **Modeling Data Cubes on the Semantic Web using QB4OLAP**
- Querying Data Cubes on the Semantic Web
- Summary

What can we represent using QB4OLAP?



Northwind DW

How can we use QB4OLAP ?

1. Enriching a QB data set
   - Given a QB data set, enrich it with QB4OLAP metadata
   - Methodology to semi-automatically do this, in progress [1]
   - Examples: WBLD, Eurostat
2. Exporting from a DW [2]
   - Given a traditional cube, export it using QB4OLAP
   - Example: Northwind DW
3. Building a QB4OLAP cube from scratch

1. Joint work with Jovan Varga, O. Romero (UPC), T.B. Pedersen, C.Thomsen (Äalborg University)

2. Joint work with E. Zimányi (ULB), M. Bouza, B. Elliot (Universidad de la República, Uruguay)

How can we use QB4OLAP ?

.

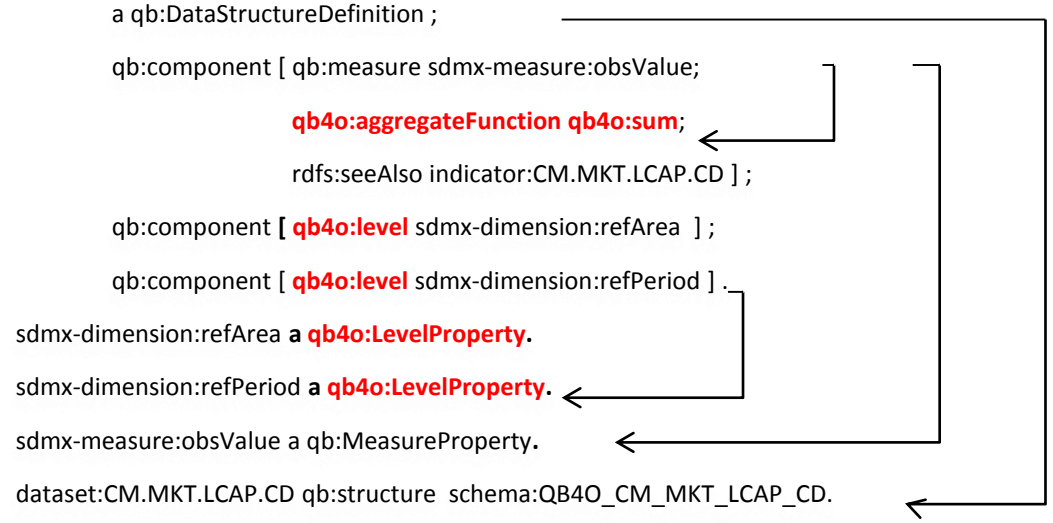
(1) Enriching a QB data set with QB4OLAP

WBLD: Data cube structure in QB4OLAP

```
@prefix qb: <http://purl.org/linked-data/cube#> .
@prefix qb4o: <http://purl.org/qb4olap/cubes#>
@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
@prefix sdmx-measure: <http://purl.org/linked-data/sdmx/2009/measure#>.
@prefix schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>
@prefix instances: <http://www.fing.edu.uy/inco/cubes/instances/world-bank-indicators#>
 schema:QB4O_CM_MKT_LCAP_CD
        a qb:DataStructureDefinition ;
        qb:component [ qb:measure sdmx-measure:obsValue;
                        qb4o:aggregateFunction qb4o:sum;
                        rdfs:seeAlso indicator:CM.MKT.LCAP.CD ] ;
        qb:component [ qb4o:level sdmx-dimension:refArea  ] ;
        qb:component [ qb4o:level sdmx-dimension:refPeriod ] .

dataset:CM.MKT.LCAP.CD qb:structure  schema:QB4O_CM_MKT_LCAP_CD.
```

WBLD: Data cube structure in QB4OLAP

```
@prefix qb: <http://purl.org/linked-data/cube#> .
@prefix qb4o: <http://purl.org/qb4olap/cubes#>
@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
@prefix sdmx-measure: <http://purl.org/linked-data/sdmx/2009/measure#>.
@prefix schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>
@prefix instances: <http://www.fing.edu.uy/inco/cubes/instances/world-bank-indicators#>
schema:QB4O_CM_MKT_LCAP_CD
        a qb:DataStructureDefinition ;
        qb:component [ qb:measure sdmx-measure:obsValue;
                qb4o:aggregateFunction qb4o:sum;
                rdfs:seeAlso indicator:CM.MKT.LCAP.CD ] ;
        qb:component [ qb4o:level sdmx-dimension:refArea  ] ;
        qb:component [ qb4o:level sdmx-dimension:refPeriod ] .
sdmx-dimension:refArea a qb4o:LevelProperty.
sdmx-dimension:refPeriod a qb4o:LevelProperty.
sdmx-measure:obsValue a qb:MeasureProperty.
dataset:CM.MKT.LCAP.CD qb:structure  schema:QB4O_CM_MKT_LCAP_CD.
```

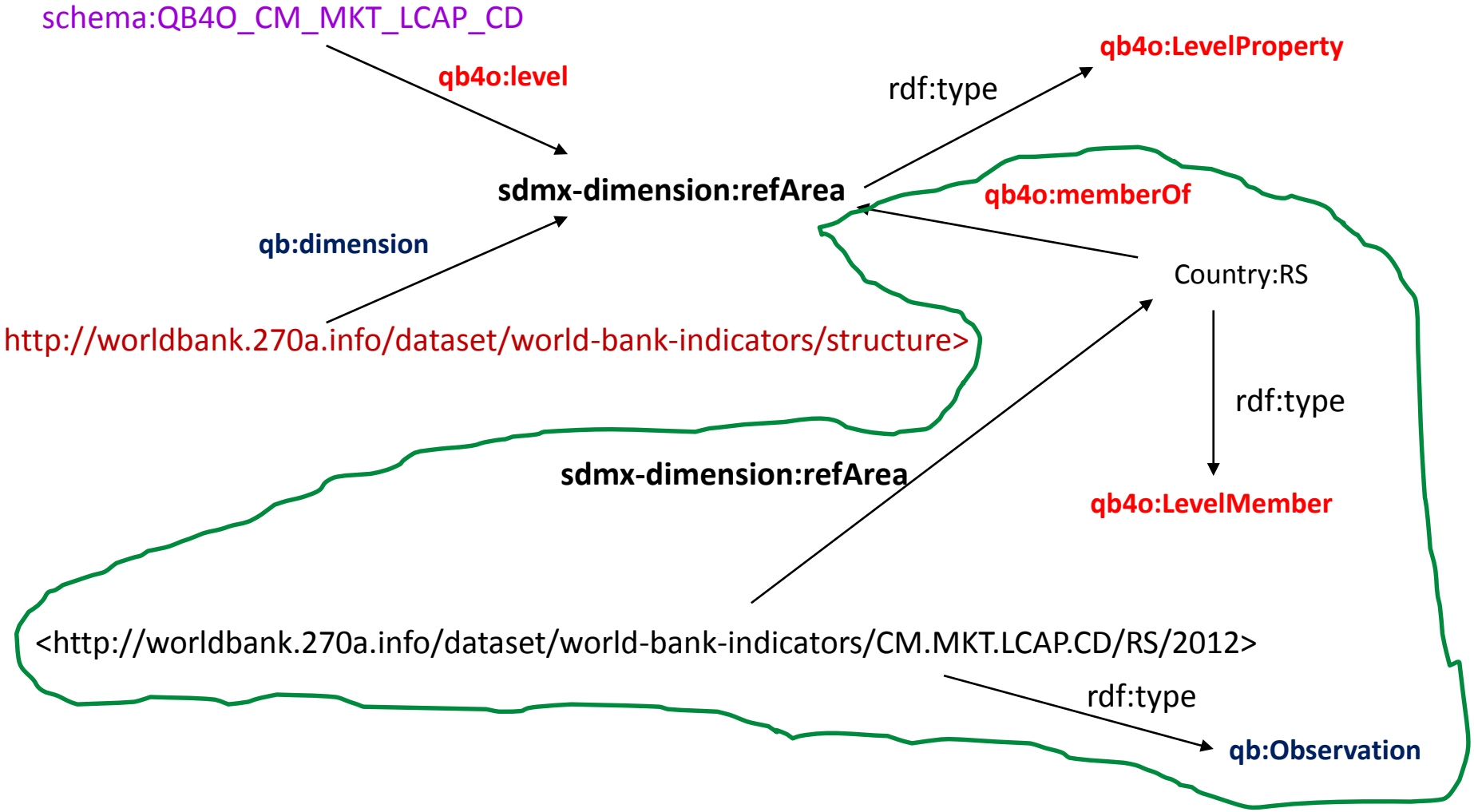


Schema: QB vs QB4OLAP

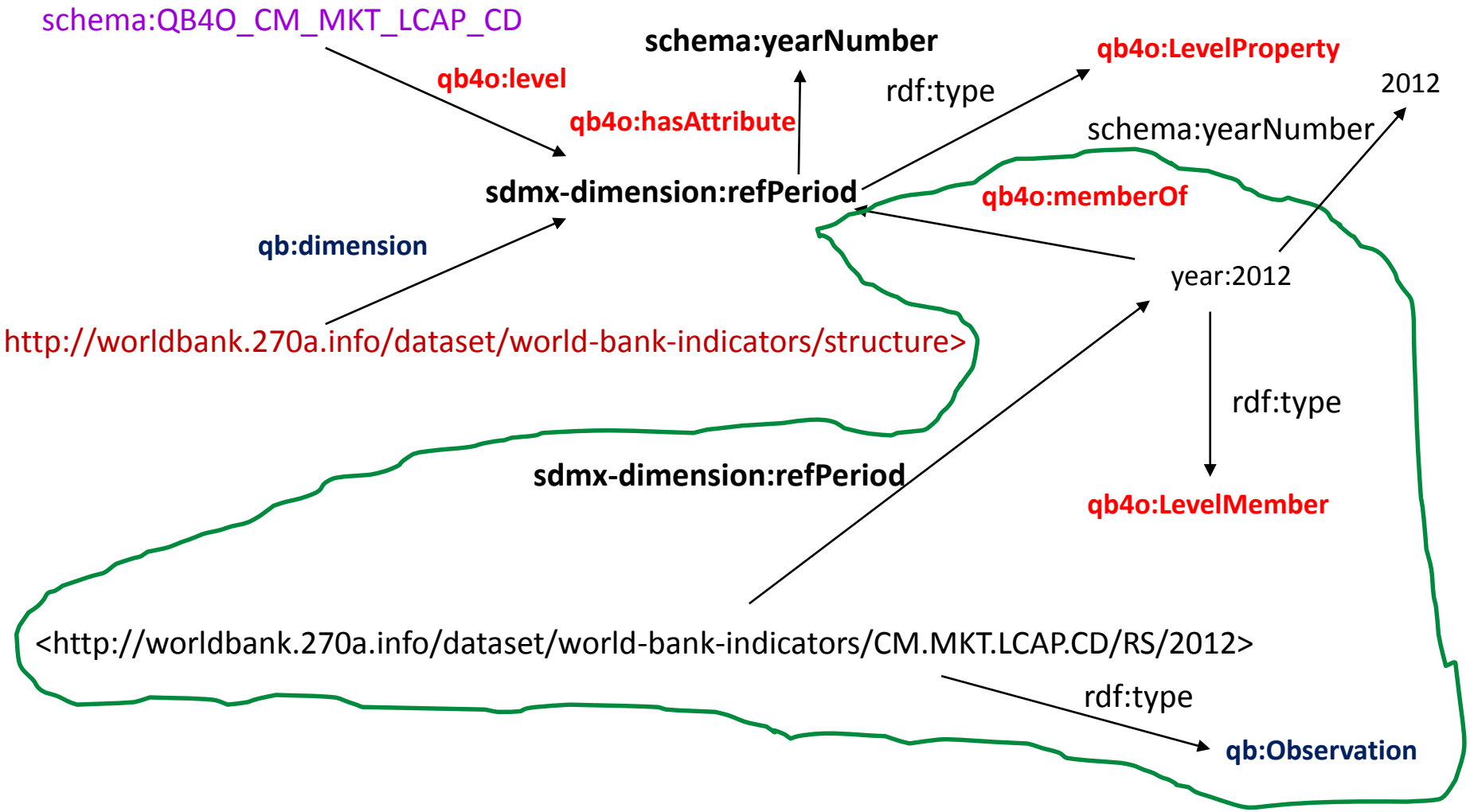
```
@prefix schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>
schema:QB4O_CM_MKT_LCAP_CD
        a qb:DataStructureDefinition ;
        qb:component [ qb:measure sdmx-measure:obsValue;
                        qb4o:aggregateFunction qb4o:sum;
                        rdfs:seeAlso indicator:CM.MKT.LCAP.CD ] ;
        qb:component [ qb4o:level sdmx-dimension:refArea  ] ;
        qb:component [ qb4o:level sdmx-dimension:refPeriod ] .
 sdmx-dimension:refArea a qb4o:LevelProperty.
 sdmx-dimension:refPeriod a qb4o:LevelProperty.
 sdmx-measure:obsValue a qb:MeasureProperty.
 dataset:CM.MKT.LCAP.CD qb:structure  schema:QB4O_CM_MKT_LCAP_CD.
```

```
<http://worldbank.270a.info/dataset/world-bank-indicators/structure>
    a qb:DataStructureDefinition ;
    qb:component  [ a qb:ComponentSpecification ;
                    qb:dimension  <http://worldbank.270a.info/property/indicator> ;
                    qb:order "1"^^xsd:int ],
                  [ a qb:ComponentSpecification ;
                    qb:measure sdmx-measure:obsValue ;
                    qb:order "4"^^xsd:int ]
                  [ a qb:ComponentSpecification ;
                    qb:dimension sdmx-dimension:refArea ;
                    qb:order "2"^^xsd:int ],
                  [ a qb:ComponentSpecification ;
                    qb:dimension sdmx-dimension:refPeriod ;
                    qb:order "3"^^xsd:int ] .
.
```

QB vs. QB4OLAP

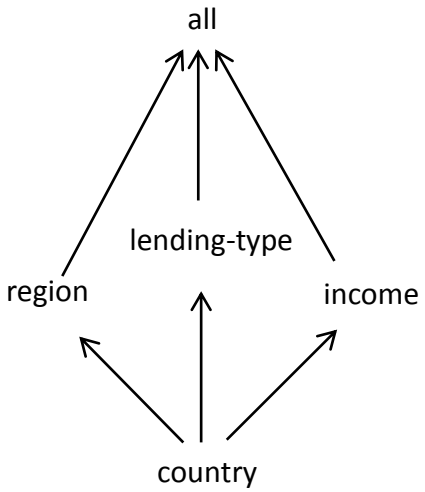


QB vs. QB4OLAP



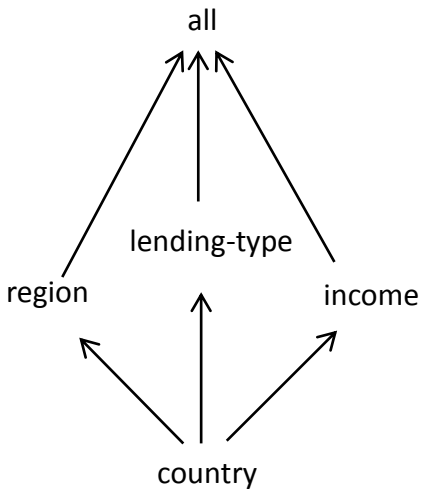
QB4OLAP: Dimension hierarchies

```
@prefix qb: <http://purl.org/linked-data/cube#> .
@prefix qb4o: <http://purl.org/qb4olap/cubes#>
@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
@prefix schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>

# Geographical dimension  and its hierarchies

schema:geoDim a qb:DimensionProperty ;
    rdfs:label "Geographical dimension"@en;
    qb4o:hasHierarchy schema:geoHier,
                      schema:lendingHier,
                      schema:incomeHier.

# Base level for the cube

sdmx-dimension:refArea a qb4o:LevelProperty;
                  rdfs:label "country level"@en.

#Upper hierarchy levels for the geographical dimension

schema:region a qb4o:LevelProperty;
                  rdfs:label "Geographical regions"@en .
schema:lendingtype a qb4o:LevelProperty;
                  rdfs:label "Lending type level"@en .
schema:income a qb4o:LevelProperty;
                  rdfs:label "Income level"@en.
schema:geoAll a qb4o:LevelProperty;
                  rdfs:label "All reference areas"@en.
```



QB4OLAP: Dimension hierarchies

```
@prefix qb: <http://purl.org/linked-data/cube#> .
@prefix qb4o: <http://purl.org/qb4olap/cubes#>
@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
@prefix schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>

# Geographical dimension  and its hierarchies

schema:geoDim a qb:DimensionProperty ;
    rdfs:label "Geographical dimension"@en;
    qb4o:hasHierarchy  schema:geoHier,
                       schema:lendingHier,
                       schema:incomeHier.

schema:geoHier a qb4o:Hierarchy ;
   rdfs:label "Geographical hierarchy"@en;
   qb4o:inDimension schema:geoDim ;
   qb4o:hasLevel sdmx-dimension:refArea, schema:region, schema:geoAll  .
.....
# Hierarchy steps for the schema:geoHier hierarchy

_:hs1  a qb4o:HierarchyStep;
       qb4o:inHierarchy schema:geoHier;
       qb4o:childLevel schema:region;
       qb4o:parentLevel schema:geoAll;
       qb4o:pcCardinality qb4o:ManyToOne.

 _:hs2  a qb4o:HierarchyStep;
      qb4o:inHierarchy schema:geoHier;
      qb4o:childLevel sdmx-dimension:refArea;
      qb4o:parentLevel schema:region;
      qb4o:pcCardinality qb4o:ManyToOne.
```



QB4OLAP: Dimension instances

```
@prefix schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>
@prefix instance: <http://www.fing.edu.uy/inco/cubes/instances/world-bank-indicators#>

 country:RS a qb4o:LevelMember;
        qb4o:memberOf sdmx-dimension:refArea;
        skos:broader lending:IBD; skos:broader income:UMC;
        skos:broader region:ECS;
        skos:prefLabel "Serbia"@en .

lending:IBD a qb4o:LevelMember;
        qb4o:memberOf schema:lending;
        skos:broader instance:geoAll;
        skos:prefLabel "IBRD"@en .

income:UMC a qb4o:LevelMember;
        qb4o:memberOf schema:income;
        skos:broader instance:geoAll;
        skos:prefLabel "Upper middle income"@en .

region:ECS a qb4o:LevelMember;
        qb4o:memberOf schema:region;
        skos:broader instance:geoAll;
        skos:prefLabel "Europe & Central Asia (all income levels)"@en .

instance:geoAll a qb4o:LevelMember;
        qb4o:memberOf schema:
        skos:prefLabel "Geo ALL"@en
.
```

QB4OLAP: Level attributes

```
@prefix qb: <http://purl.org/linked-data/cube#> .
@prefix qb4o: <http://purl.org/qb4olap/cubes#>
@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
@prefix schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>


# Time dimension

schema:timeDim a qb:DimensionProperty ;
    rdfs:label "Time dimension"@en;
    qb4o:hasHierarchy schema:timeHier.

schema:timeHier a qb4o:Hierarchy ;
   rdfs:label “Time hierarchy"@en;
   qb4o:inDimension schema:timeDim ;
   qb4o:hasLevel sdmx-dimension:refPeriod, schema:timeAll  .



sdmx-dimension:refPeriod a qb4o:LevelProperty;
    rdfs:label "year level"@en.

sdmx-dimension:refPeriod  qb4o:hasAttribute   schema:yearNumber

schema:yearNumber  a qb4o:LevelAttribute;
    rdfs:label "year number"@en.
```

QB4OLAP: Attribute instances

@prefix qb: <http://purl.org/linked-data/cube#> .
@prefix qb4o: <http://purl.org/qb4olap/cubes#>
@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
@prefix schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>
@prefix instance: <http://www.fing.edu.uy/inco/cubes/instances/world-bank-indicators#>
@prefix year: <http://reference.data.gov.uk/id/year>

 # Time dimension

schema:timeDim a **qb:DimensionProperty** ;
rdfs:label "Time dimension"@en;
qb4o:hasHierarchy schema:timeHier.

sdmx-dimension:refPeriod a **qb4o:LevelProperty**;
rdfs:label "year level"@en.

sdmx-dimension:refPeriod **qb4o:hasAttribute** schema:yearNumber

schema:yearNumber a **qb4o:LevelAttribute**;
rdfs:label "year number"@en.

……
year:2012 a **qb4o:LevelMember**;
qb4o:memberOf sdmx-dimension:refPeriod ;
skos:broader instance:timeAll;
skos:prefLabel ”2012"@en .

year:2012 schema:yearNumber “2012” ^^xsd:integer

Another example: Eurostat

- Statistics about asylum appplications to the EU
- Applications by month, age, gender, citizenship, destination
- Published using QB
- Again, we use strategy of enrichment of the QB data set
- Observations in tabular form:

Sex	Age	Time	Application_type	Citizenship	Destination	Measures
Sex	*Age*	*Month*	*Application_type*	*Country*	*Country*	*#applications*
M	14 to 17	201301, January 2013	new applicant	CM, Cameroon	BE, Belgium	5
F	less than 14	201303, March 2013	new applicant	CM, Cameroon	FR, France	5
M	18 to 34	201301, January 2013	new applicant	CM, Cameroon	FR, France	10
F	18 to 34	201301, January 2013	new applicant	CD, Democratic Republic of the Congo	BE, Belgium	25
F	18 to 34	201303, March 2013	new applicant	CD, Democratic Republic of the Congo	BE, Belgium	30

Eurostat observations

prefix qb:http://purl.org/linked-data/cube#

SELECT <http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11> ?p ?o

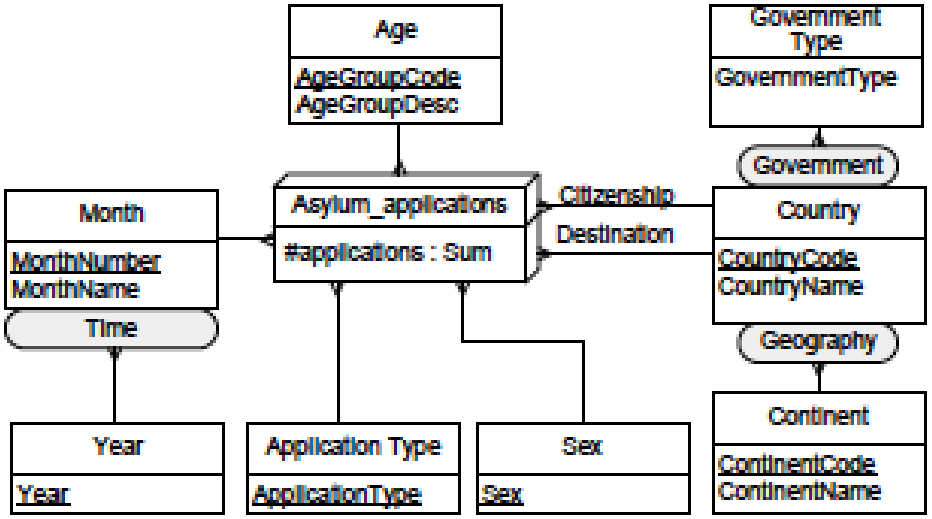
FROM <http://www.fing.edu.uy/inco/cubes/instances/migr_asyapp>

WHERE{<http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11> ?p ?o . }

callret-0	p	o
http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://purl.org/linked-data/cube#Observation
http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11	http://purl.org/linked-data/cube#dataSet	http://eurostat.linked-statistics.org/data/migr_asyappctzm
http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11	http://purl.org/linked-data/sdmx/2009/dimension#freq	http://purl.org/linked-data/sdmx/2009/code#freq-M
http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11	http://eurostat.linked-statistics.org/property#citizen	http://eurostat.linked-statistics.org/dic/citizen#AD
http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11	http://eurostat.linked-statistics.org/property#sex	http://eurostat.linked-statistics.org/dic/sex#F
http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11	http://eurostat.linked-statistics.org/property#age	http://eurostat.linked-statistics.org/dic/age#TOTAL
http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11	http://eurostat.linked-statistics.org/property#asyl_app	http://eurostat.linked-statistics.org/dic/asyl_app#ASY_APP
http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11	http://eurostat.linked-statistics.org/property#geo	http://eurostat.linked-statistics.org/dic/geo#AT
http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11	http://purl.org/linked-data/sdmx/2009/dimension#timePeriod	2013-11-01
http://eurostat.linked-statistics.org/data/migr_asyappctzm#M,AD,F,TOTAL,ASY_APP,AT,2013M11	http://purl.org/linked-data/sdmx/2009/measure#obsValue	0

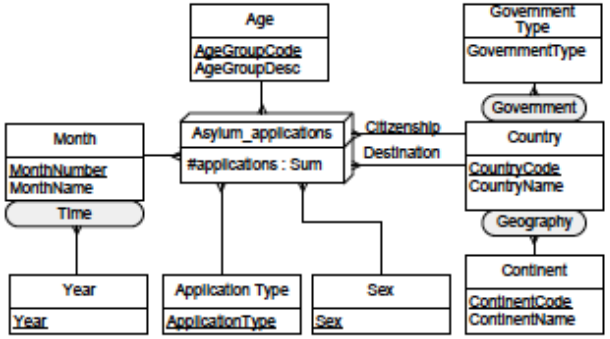
Eurostat: Model

- Multidimensional conceptual model



Eurostat QB4OLAP schema

```
@prefix property: <http://eurostat.linked-statistics.org/property#>  .
@prefix sdmx-measure:     <http://purl.org/linked-data/sdmx/2009/measure#> .
@prefix sdmx-dimension: <http://purl.org/linked-data/sdmx/2009/dimension#> .
@prefix schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#> .
@prefix qb4o: <http://purl.org/qb4olap/cubes#> .

schema:migr_asyappctzmQB4O    rdf:type  qb:DataStructureDefinition;
        qb:component [ qb:measure sdmx-measure:obsValue; qb4o:aggregateFunction qb4o:sum ] ;
        qb:component [ qb4o:level property:age ; qb4o:cardinality qb4o:ManyToOne  ] ;
        qb:component [ qb4o:level sdmx-dimension:refPeriod ; qb4o:cardinality qb4o:ManyToOne ]
        qb:component [ qb4o:level property:sex ; qb4o:cardinality qb4o:ManyToOne ] ;
        qb:component [ qb4o:level property:geo ; qb4o:cardinality qb4o:ManyToOne ] ;
        qb:component [ qb4o:level property:citizen  qb4o:cardinality qb4o:ManyToOne  ] ;
        qb:component [ qb4o:level property:asyl_app ; qb4o:cardinality qb4o:ManyToOne  ] ;
        skos:notation "migr_asyappctzm_DSD" .

#states that eurostat instances are described by the schema defined in QB4OLAP
<http://eurostat.linked-statistics.org/data/migr_asyappctzm> qb:structure schema:migr_asyappctzmQB4O.
sdmx-measure:obsValue a qb:MeasureProperty;
      rdfs:label "Number of applications"@en;   rdfs:range xsd:integer .
```



Age dimension

```
schema:ageDim a qb:DimensionProperty ;
  rdfs:label ”Age range dimension"@en ;
  qb4o:hasHierarchy schema:ageHier .

schema:ageHier a qb4o:Hierarchy ;
  rdfs:label ”Age Hierarchy"@en ;
  qb4o:inDimension schema:ageDim ;
  qb4o:hasLevel property:age, schema:ageAll .

property:age  a qb4o:LevelProperty ;
  rdfs:label ”Age range"@en .

schema:ageAll a qb4o:LevelProperty ;
   rdfs:label "All ages"@en .

_:ih11 a qb4o:HierarchyStep ;
   qb4o:inHierarchy schema:ageHier ;
   qb4o:childLevel property:age;
   qb4o:parentLevel schema:ageAll;
   qb4o:pcCardinality qb4o:ManyToOne .
```

Age dimension instances

age:Y_LT14
    qb4o:memberOf property:age ;
    skos:broader age:TOTAL ;
    skos:prefLabel "Moins de 14 ans"@fr , "Less than 14 years"@en , "Weniger als 14 Jahre"@de .

age:Y14-17
    qb4o:memberOf property:age ;
    skos:broader age:TOTAL ;
    skos:prefLabel "14 bis 17 Jahre"@de , "De 14 à 17 ans"@fr , "From 14 to 17 years"@en .
…..
……
age:UNK
    qb4o:memberOf property:age **;**
    skos:broader age:TOTAL ;
    skos:prefLabel "Unbekannt"@de , "Inconnu"@fr , "Unknown"@en .

age:TOTAL
   qb4o:memberOf schema:ageAll;
   rdf:type skos:Concept ;
   skos:prefLabel "Insgesamt"@de , "Total"@fr , "Total"@en .

*** Note that observations can be at different granularity levels (schema:ageAll, property:age)

Time dimension

```
schema:timeDim a qb:DimensionProperty  ;
  rdfs:label "Time dimension"@en  ;
  qb4o:hasHierarchy schema:timeHier .

schema:timeHier a qb4o:Hierarchy  ;
  rdfs:label "Time Hierarchy"@en  ;
  qb4o:inDimension schema:timeDim ;
  qb4o:hasLevel sdmx-dimension:refPeriod, schema:year , schema:timeAll .

sdmx-dimension:refPeriod a qb4o:LevelProperty ;
  rdfs:label "Month level"@en  .
 -- sdmx-refPeriod may have any granularity;  in Eurostat: month or year; manual identification of  granularity

schema:year a qb4o:LevelProperty ;
  rdfs:label "Year"@en  .

schema:timeAll a qb4o:LevelProperty ;
   rdfs:label "All dates"@en .

_:ih21 a qb4o:HierarchyStep ;
   qb4o:inHierarchy schema:timeHier ;
   qb4o:childLevel sdmx-dimension:refPeriod ;
   qb4o:parentLevel schema:year;  qb4o:pcCardinality qb4o:ManyToOne .

_:ih22 a qb4o:HierarchyStep;
   qb4o:inHierarchy schema:timeHier ;
   qb4o:childLevel schema:year ;
   qb4o:parentLevel schema:timeAll ;  qb4o:OneToManyToOne .
```

Time dimension instances

```
@prefix time:<http://purl.org/qb4olap/dimensions/time#> .

time:TOTAL
     qb4o:memberOf schema:timeAll .

time:2008
    qb4o:memberOf schema:year;
    skos:broader time:TOTAL .

time:2009
    qb4o:memberOf schema:year;
    skos:broader time:TOTAL .

time:2014
    qb4o:memberOf schema:year;
    skos:broader time:TOTAL .

time:200801
    qb4o:memberOf sdmx-dimension:refPeriod;
    skos:broader time:2008 .
….
time:201401
     qb4o:memberOf sdmx-dimension:refPeriod;
     skos:broader time:2014 .

…..
```

Sex dimension

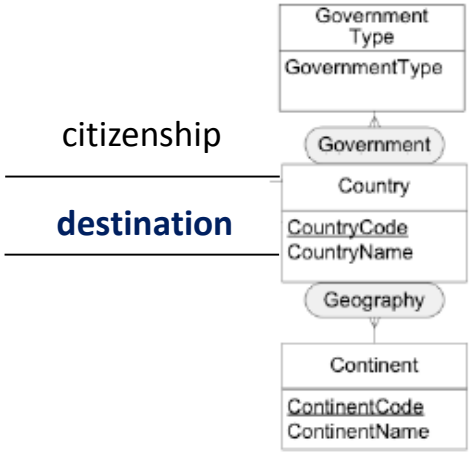
```
schema:sexDim a qb:DimensionProperty  ;
  rdfs:label ”Sex dimension"@en  ;
  qb4o:hasHierarchy schema:sexHier .

schema:sexHier a qb4o:Hierarchy  ;
  rdfs:label ”Sex Hierarchy"@en  ;
  qb4o:inDimension schema:sexDim ;
  qb4o:hasLevel property:sex, schema:sexAll .

Property:sex  a qb4o:LevelProperty ;
  rdfs:label ”Sex"@en  .

Schema:sexAll a qb4o:LevelProperty ;
   rdfs:label "All sexes"@en .

_:ih31 a qb4o:HierarchyStep ;
   qb4o:inHierarchy schema:sexHier ;
   qb4o:childLevel property:sex;
   qb4o:parentLevel schema:sexAll;
   qb4o:cardinality qb4o:ManyToOne .
```

Sex dimension instances

```
@prefix sex:<http://eurostat.linked-statistics.org/dic/sex#> .

sex:F
    qb4o:memberOf property:sex ;
    skos:broader sex:T ;
    skos:notation "F" ;
    skos:prefLabel "Frauen"@de , "Femmes"@fr , "Females"@en .

sex:M
    qb4o:memberOf property:sex ;
    skos:broader sex:T ;
    skos:notation "M" ;
    skos:prefLabel "Hommes"@fr , "Males"@en , "Männer"@de .

sex:UNK
    qb4o:memberOf property:sex ;
    skos:broader sex:T ;
    skos:notation "UNK" ;
    skos:prefLabel "Unbekannt"@de , "Inconnu"@fr , "Unknown"@en .

sex:T
    qb4o:memberOf schema:sexAll;
    skos:notation "T" ;
    skos:prefLabel "Insgesamt"@de , "Total"@fr , "Total"@en .
```

Destination dimension

```
schema:destinationDim a qb:DimensionProperty ;
  rdfs:label "Asylum geographical destination dimension"@en;
  qb4o:hasHierarchy schema:destinationGeoHier, schema:destinationGovHier.

schema:destinationGeoHier a qb4o:Hierarchy ;
   rdfs:label "Asylum destination  Geographical Hierarchy"@en ;
   qb4o:inDimension schema:destinationDim;
   qb4o:hasLevel property:geo, schema:continent, schema:destAll.

schema:destinationGovHier a qb4o:Hierarchy ;
   rdfs:label "Asylum destination Government Hierarchy"@en ;
   qb4o:inDimension schema:destinationDim;
   qb4o:hasLevel property:geo, schema:governmentType, schema:destAll

property:geo a qb4o:LevelProperty;
  rdfs:label "Country of asylum application"@en .

schema:continent a qb4o:LevelProperty;
  rdfs:label "Continent"@en .

schema:governmentType a qb4o:LevelProperty;
   rdfs:label "Government Type"@en .

schema:destAll a qb4o:LevelProperty;
   rdfs:label "All destinations"@en .
```



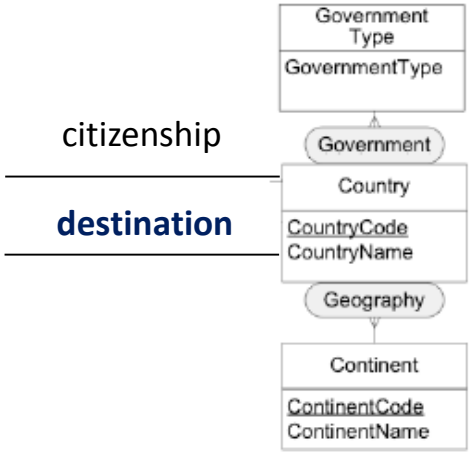
We need two different dimensions because instances are contained in levels, and bottom levels must differ from each other. In Eurostat geo instances are different from citizen instances, even though they refer to the same conuntry => represented by different IRIs

Destination dimension

schema:destinationDim a **qb:DimensionProperty** ;
rdfs:label "Asylum geographical destination dimension"@en;
qb4o:hasHierarchy schema:destinationGeoHier, schema:destinationGovHier.

schema:destinationGeoHier **a qb4o:Hierarchy** ;
rdfs:label "Asylum destination Geographical Hierarchy"@en ;
qb4o:inDimension schema:destinationDim;
qb4o:hasLevel property:geo, schema:continent, schema:destAll.

schema:destinationGovHier a **qb4o:Hierarchy** ;
rdfs:label "Asylum destination Government Hierarchy"@en ;
qb4o:inDimension schema:destinationDim;
qb4o:hasLevel property:geo, schema:governmentType, schema:destAll



_:ih41 a **qb4o:HierarchyStep** ;
qb4o:inHierarchy *schema:destinationGeoHier* ;
qb4o:ChildLevel property:geo ; **qb4o:parentLevel** schema:continent ; **qb4o:pcCardinality** qb4o : ManyToOne .

_:ih42 a **qb4o:HierarchyStep** ;
qb4o:inHierarchy *schema:destinationGeoHier* ;
qb4o:ChildLevel schema:continent ; **qb4o:parentLevel** schema:destAll; **qb4o:pcCardinality** qb4o : ManyToOne .

_:ih43 a **qb4o:HierarchyStep** ;
qb4o:inHierarchy *schema:destinationGovHier* ;
qb4o:ChildLevel property:geo ; **qb4o:parentLevel** schema:governmentType ; **qb4o:pcCardinality** qb4o : ManyToOne .

_:ih44 a **qb4o:HierarchyStep** ;
qb4o:inHierarchy *schema:destinationGovHier* ;
qb4o:ChildLevel schema:governmentType ; **qb4o:parentLevel** schema:destAll ; **qb4o:pcCardinality** qb4o : ManyToOne .

Destination dimension instances

@prefix desDim: <http://www.fing.edu.uy/inco/cubes/dims/migr_asyapp/destination#> . **- prefix of the instances created manually**

<http://eurostat.linked-statistics.org/dic/geo#CZ>
rdf:type skos:Concept ;
qb4o:memberOf property:geo ;
skos:broader desDim:EU ; -- corresponds to continent level
skos:broader dbpedia:Parliamentary_republic ; --- this was added manually – corresponds to government type
skos:inScheme dic:geo ; skos:notation "CZ" ;
skos:prefLabel "Tschechische Republik"@de , "République tchèque"@fr , "Czech Republic"@en .

<http://eurostat.linked-statistics.org/dic/geo#SK>
rdf:type skos:Concept ;
qb4o:memberOf property:geo ;
skos:broader desDim:EU ;
skos:broader dbpedia:Parliamentary_republic ; --- this was added manually
skos:inScheme dic:geo ; skos:notation "SK" ;
skos:prefLabel "Slovaquie"@fr , "Slowakei"@de , "Slovakia"@en .

<http://eurostat.linked-statistics.org/dic/geo#NO>
rdf:type skos:Concept ;
qb4o:memberOf property:geo ;
skos:broader desDim:EU ;
skos:broader dbpedia:Unitary State ; --- this was added manually
skos:inScheme dic:geo ; skos:notation "NO" ;
skos:prefLabel "Norvège"@fr , "Norway"@en , "Norwegen"@de .
……
desDim:EU
qb4o:memberOf schema: continent ;
skos:broader desDim:TOTAL; skos:prefLabel "Europe"@en .

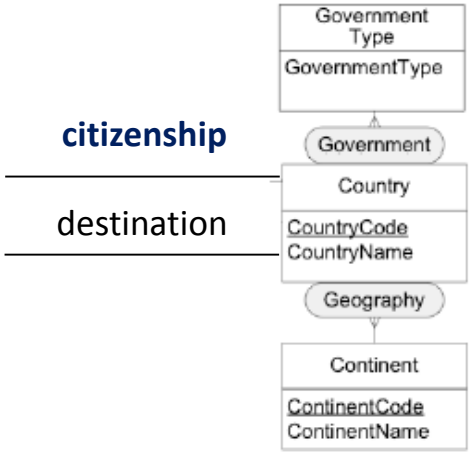
desDim:TOTAL
qb4o:memberOf schema:destAll; skos:prefLabel "All destinations”@en .

Destination dimension instances (cont.)

```
dbpedia:Parliamentary_republic  qb4o:memberOf  schema:governmentType ;
    skos:broader    desDim:TOTAL ;
    skos:prefLabel"Parliamentary republic"@en ;
    skos:notation    "Parliamentary republic"@en .

dbpedia:Unitary_state  qb4o:memberOf   schema:governmentType ;
    skos:broader      desDim:TOTAL ;
    skos:prefLabel"Unitary state"@en ;
   skos:notation  "Unitary state"@en .

dbpedia:Federalism   qb4o:memberOf        schema:governmentType ;
     skos:broader   desDim:TOTAL ;
     skos:prefLabel           "Federalism"@en ;
     skos:notation    "Federalism"@en .

dbpedia:Federal_Government_of_Somalia     qb4o:memberOf    schema:governmentType ;
     skos:broader      desDim:TOTAL ;
     skos:prefLabel           "Federal Government of Somalia"@en ;
     skos:notation    "Federal Government of Somalia"@en .

dbpedia:Republic qb4o:memberOf schema:governmentType ;
      skos:broader  desDim:TOTAL ;
      skos:prefLabel           "Republic"@en ;
      skos:notation   "Republic"@en ; .

 …..
```

Citizenship dimension

```
schema:citizenshipDim a qb:DimensionProperty ;
    rdfs:label "Applicant citizenship dimension"@en;
    qb4o:hasHierarchy schema:citizenshipGeoHier, schema:citizenshipGovHier.

schema:citizenshipGeoHier a qb4o:Hierarchy ;
    rdfs:label "Applicant citizenship Geo Hierarchy"@en ;
    qb4o:inDimension   schema:citizenshipDim;
    qb4o:hasLevel   property:citizen, schema:continent, schema:citAll.

schema:citizenshipGovHier a qb4o:Hierarchy ;
    rdfs:label "Applicant citizenship Government Hierarchy"@en ;
    qb4o:inDimension schema:citizenshipDim;
    qb4o:hasLevel property:citizen, schema:governmentType, schema:citAll.

Property:citizen a qb4o:LevelProperty;
    rdfs:label "Country of asylum application"@en .

Schema:citAll a qb4o:LevelProperty;
     rdfs:label "All citizenships"@en .
```

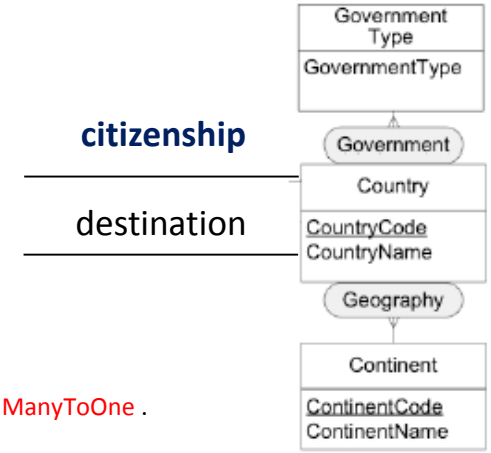


Citizenship dimension

schema:citizenshipDim a **qb:DimensionProperty** ;
rdfs:label “Citizenship dimension"@en ;
qb4o:**hasHierarchy** schema:citizenshipGeoHier, schema:citizenshipGovHier.

schema:citizenshipGeoHier a **qb4o:Hierarchy** ;
qb4o:inDimension schema:citizenshipDim ;
qb4o:hasLevel property:country, *schema:continent*, schema:citAll .

schema:citizenshipGovHier **a** **qb4o:Hierarchy** ;
qb4o:inDimension schema:citizenshipDim ;
qb4o:hasLevel property:country, *schema:governmentType*, schema:citAll .



_:ih45 a **qb4o:HierarchyStep** ;
qb4o:inHierarchy *schema::citizenshipGeoHier* ;
qb4o:ChildLevel property:citizen ; **qb4o:parentLevel** schema:continent ; **qb4o:pcCardinality** qb4o : ManyToOne .

_:ih46 a **qb4o:HierarchyStep** ;
qb4o:inHierarchy *schema:citizenshipGeoHier* ;
qb4o:ChildLevel schema:continent ; **qb4o:parentLevel** schema:citAll; **qb4o:pcCardinality** qb4o: ManyToOne .

_:ih47 a **qb4o:HierarchyStep** ;
qb4o:inHierarchy *schema:destinationGovHier* ;
qb4o:ChildLevel property:citizen ; **qb4o:parentLevel** schema:governmentType ; **qb4o:pcCardinality** qb4o : ManyToOne .

_:ih48 a **qb4o:HierarchyStep** ;
qb4o:inHierarchy *schema:destinationGovHier* ;
qb4o:ChildLevel schema:governmentType ; **qb4o:parentLevel** schema:citAll ; **qb4o:pcCardinality** qb4o : ManyToOne .

.

Citizenship dimension instances

@prefix desDim: <http://www.fing.edu.uy/inco/cubes/dims/migr_asyapp/destination#> . **- prefix of the instances created manually**

<http://eurostat.linked-statistics.org/dic/citizen#AR>
rdf:type skos:Concept ;
qb4o:memberOf property:citizen ;
skos:broader citDim:SA ;
skos:broader dbpedia:Federal_republic ;
skos:inScheme dic:citizen ; skos:notation "AR" ;
skos:prefLabel "Argentine"@fr , "Argentina"@en , "Argentinien"@de .

<http://eurostat.linked-statistics.org/dic/citizen#CF>
rdf:type skos:Concept ;
qb4o:memberOf property:citizen ;
skos:broader citDim:AF ;
skos:broader dbpedia:Provisional_government;
skos:inScheme dic:citizen ; skos:notation "CF" ;
skos:prefLabel "Central African Republic"@en , "Zentralafrikanische Republik"@de , "République centrafricaine"@fr .

<http://eurostat.linked-statistics.org/dic/citizen#NL>
rdf:type skos:Concept ;
qb4o:memberOf property:citizen **;**
skos:broader citDim:EU ;
skos:broader dbpedia:Unitary state;
skos:inScheme dic:citizen ; skos:notation "NL" ;
skos:prefLabel "Netherlands"@en , "Niederlande"@de , "Pays-Bas"@fr .

<http://eurostat.linked-statistics.org/dic/citizen#AZ>
rdf:type skos:Concept ;
qb4o:memberOf property:citizen ;
skos:broader citDim:EU ;
skos:broader dbpedia:Unitary state;
…..

Citizenship dimension instances (cont.)

```
dbpedia:Parliamentary_republic  qb4o:memberOf  schema:governmentType ;
    skos:broader   citDim:TOTAL ;
    skos:prefLabel "Parliamentary republic"@en ;
    skos:notation "Parliamentary republic"@en .

dbpedia:Unitary_state  qb4o:memberOf   schema:governmentType ;
    skos:broader     citDim:TOTAL ;
    skos:prefLabel "Unitary state"@en ;
    skos:notation  "Unitary state"@en .

dbpedia:Federalism   qb4o:memberOf        schema:governmentType ;
     skos:broader   citDim:TOTAL ;
     skos:prefLabel "Federalism"@en ;
    skos:notation  "Federalism"@en .

dbpedia:Federal_Government_of_Somalia     qb4o:memberOf    schema:governmentType ;
     skos:broader  citDim:TOTAL ;
     skos:prefLabel "Federal Government of Somalia"@en ;
     skos:notation   "Federal Government of Somalia"@en .

dbpedia:Republic qb4o:memberOf schema:governmentType ;
      skos:broader  citDim:TOTAL ;
      skos:prefLabel  "Republic"@en ;
      skos:notation    Republic"@en .

 …..
```

Citizenship dimension instances (cont.)

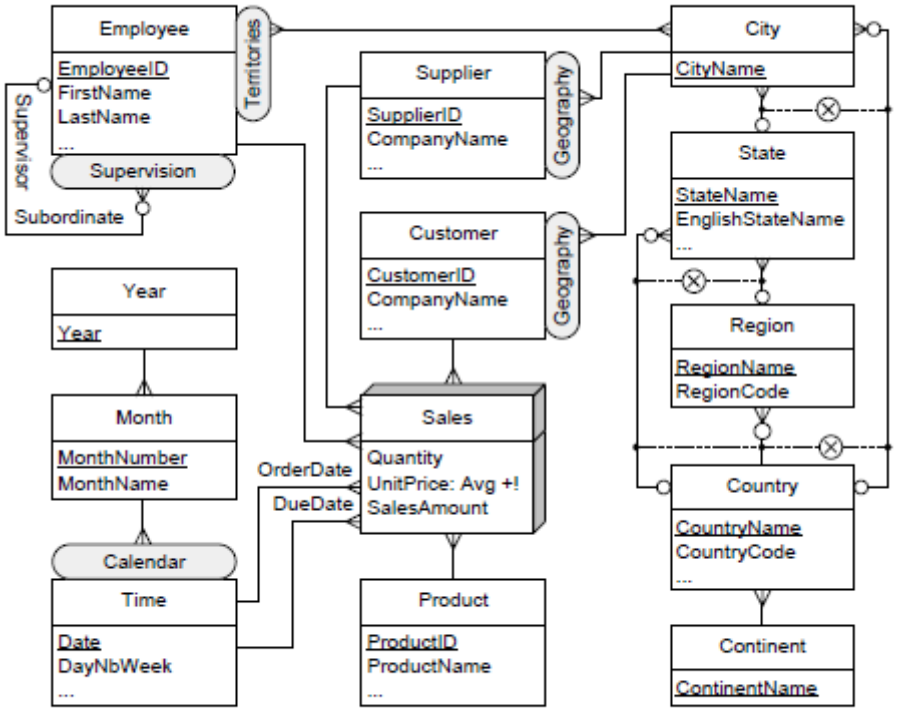
```
citDim:TOTAL
   qb4o:memberOf schema:citAll;
   skos:notation "All citizenships" ;
   skos:prefLabel "All citizenships"@en .

citDim:AF
     qb4o:memberOf schema:continent ;
     skos:broader citDim:TOTAL;
     skos:notation "AF" ;
     skos:prefLabel "Africa"@en .

citDim:EU
     qb4o:memberOf schema:continent ;
     skos:broader citDim:TOTAL;
     skos:notation "EU" ;
     skos:prefLabel "Europe"@en .

citDim:SA
     qb4o:memberOf schema:continent ;
     skos:broader citDim:TOTAL;
     skos:notation "SA" ;
     skos:prefLabel "South America"@en .

citDim:NA
     qb4o:memberOf schema:continent ;
     skos:broader citDim:TOTAL;
     skos:notation "NA" ;
     skos:prefLabel "Northern America"@en .
….
```

How can we use QB4OLAP ?

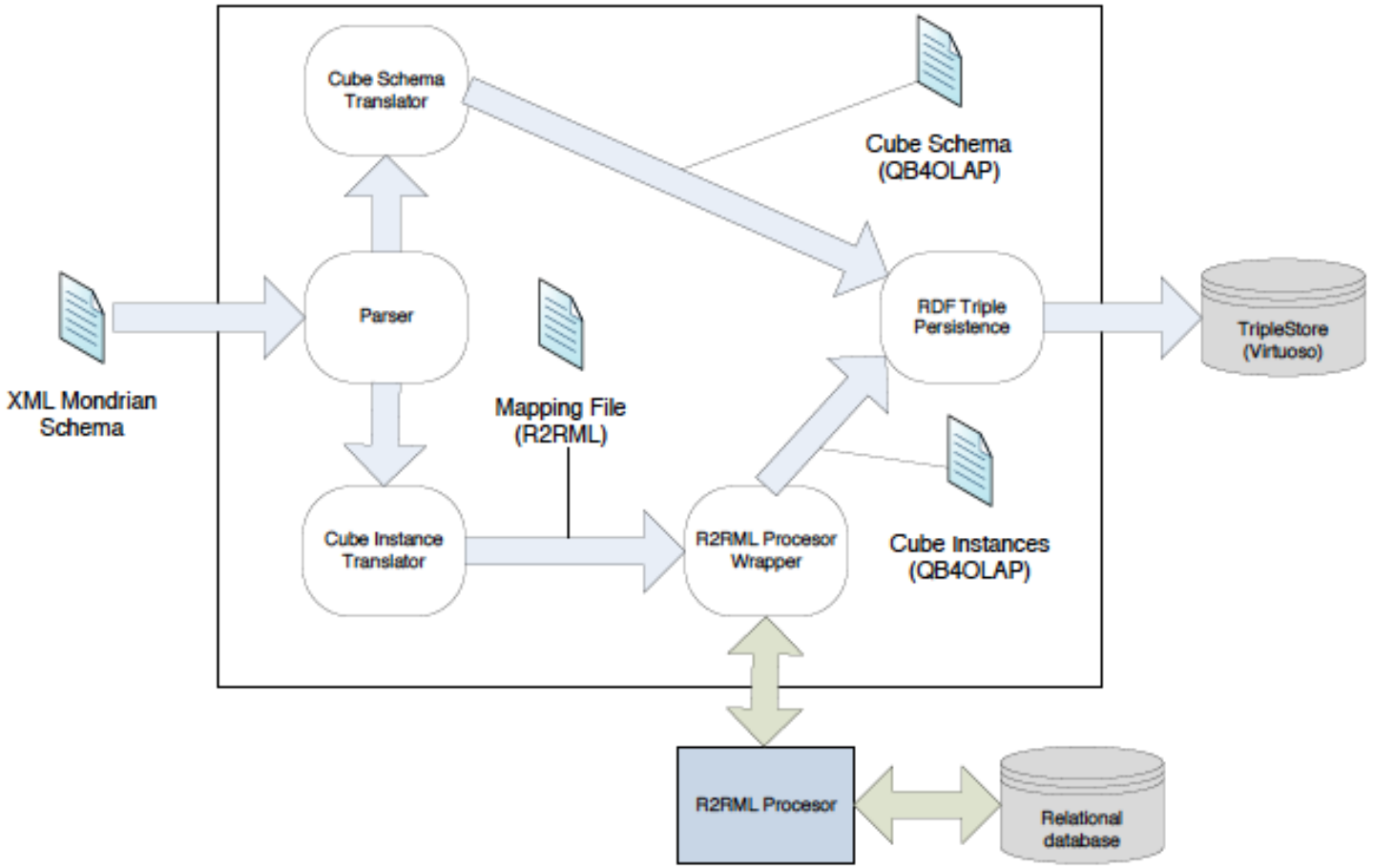
.

(2) Exporting from an existing DW

Publishing a DW using QB4OLAP



Publishing a DW using QB4OLAP

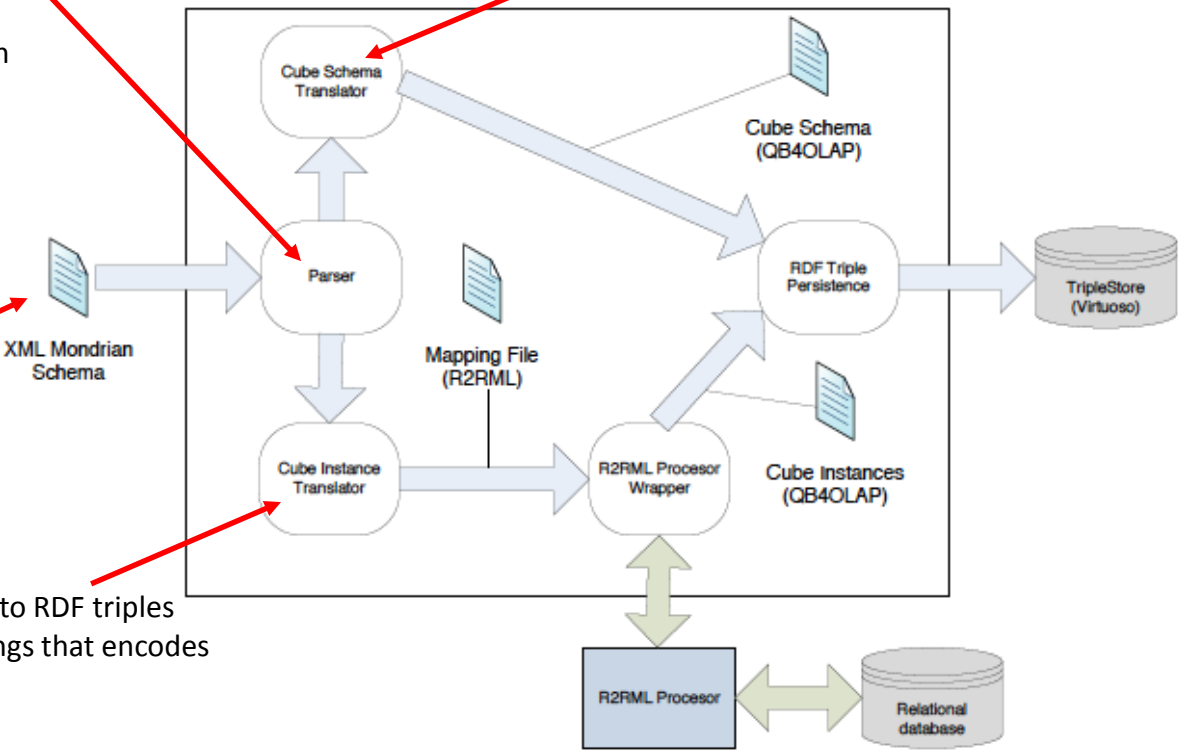


Publishing a DW using QB4OLAP

- Validates the input XML file (Mondrian schema) against DTD
- Extracts the logical model of the cube
- Extracts the mappings to the physical model
- Creates an in-memory representation of this information

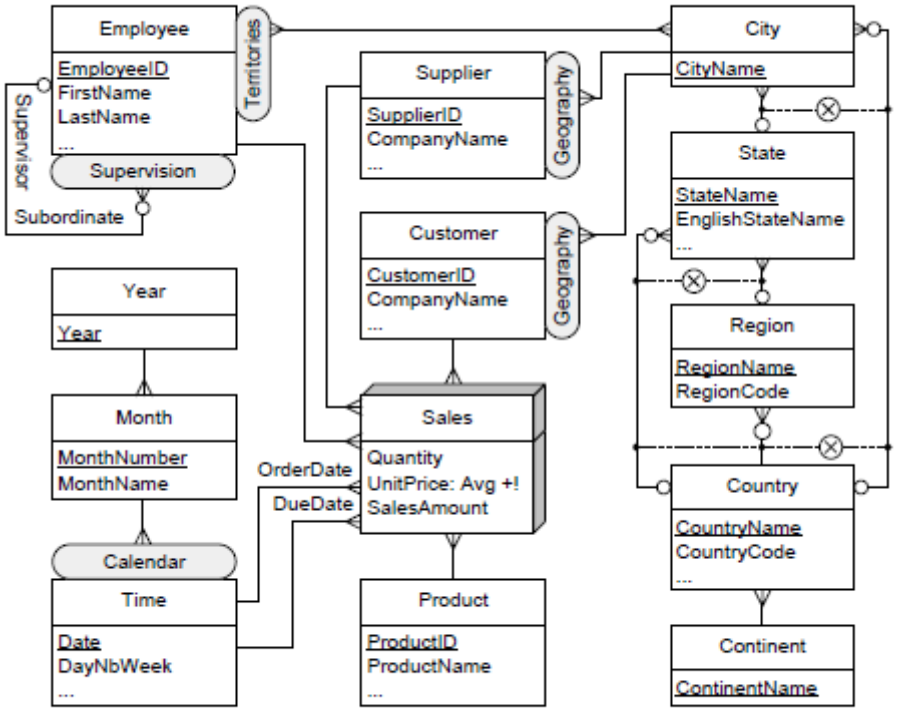
- Translates the logical model of the data cube into RDF,
- Output: QB4OLAP cube schema

- Mondrian schema: XML document
- Defines an MD database
- Contains cubes, dimensions, hierarchies, and members, and a mapping to a physical model.



- Translates the data in the RDB into RDF triples
- Produces a set of R2RML mappings that encodes these instances

QB4OLAP for Northwind DW



QB4OLAP for Northwind DW - Schema

```
@prefix nw: <http://dwbook.org/cubes/schemas/northwind#> .

 # Cube definition

 nw:Northwind a qb:DataStructureDefinition ;

 # Lowest level for each dimension in the cube
 qb:component [qb4o:level nw:employee ; qb4o:cardinality qb4o:ManyToOne ] ;
 qb:component [qb4o:level nw:orderDate ; qb4o:cardinality qb4o:ManyToOne ] ;
 qb:component [qb4o:level nw:dueDate ; qb4o:cardinality qb4o:ManyToOne ] ;
 qb:component [qb4o:level nw:shippedDate ; qb4o:cardinality qb4o:ManyToOne ] ;
 qb:component [qb4o:level nw:product ; qb4o:cardinality qb4o:ManyToOne ] ;
 qb:component [qb4o:level nw:order ; qb4o:cardinality qb4o:OneToOne ] ;
 qb:component [qb4o:level nw:shipper ; qb4o:cardinality qb4o:ManyToOne ] ;
 qb:component [qb4o:level nw:supplier ; qb4o:cardinality qb4o:ManyToOne ] ;
 qb:component [qb4o:level nw:customer ; qb4o:cardinality qb4o:ManyToOne ] ;

# Measures in the cube
 qb:component [qb:measure nw:quantity ; qb4o:hasAggregateFunction qb4o:sum] ;
 qb:component [qb:measure nw:unitPrice ; qb4o:hasAggregateFunction qb4o:avg] ;
 qb:component [qb:measure nw:discount ; qb4o:hasAggregateFunction qb4o:avg] ;
 qb:component [qb:measure nw:salesAmount ; qb4o:hasAggregateFunction qb4o:sum] ;
 qb:component [qb:measure nw:freight ; qb4o:hasAggregateFunction qb4o:sum] ;
 qb:component [qb:measure nw:netAmount ; qb4o:hasAggregateFunction qb4o:sum] .
```

QB4OLAP for Northwind DW - Schema

```
@prefix nw: <http://dwbook.org/cubes/schemas/northwind#> .

 # -- Measure definition

nw:quantity a rdf:Property , qb:MeasureProperty ;
        rdfs:label "Quantity"@en ;
        rdfs:subPropertyOf sdmx-measure:obsValue ;
        rdfs:range xsd:positiveInteger .

nw:unitPrice a rdf:Property , qb:MeasureProperty ;
        rdfs:label "Unit Price"@en ;
        rdfs:subPropertyOf sdmx-measure:obsValue ;
        rdfs:range xsd:decimal .

.....

nw:freight a rdf:Property , qb:MeasureProperty ;
        rdfs:label "Freight"@en ;
        rdfs:subPropertyOf sdmx-measure:obsValue ;
        rdfs:range xsd:decimal .

nw:netAmount a rdf:Property , qb:MeasureProperty ;
        rdfs:label "Net Amount"@en ;
        rdfs:subPropertyOf sdmx-measure:obsValue ;
        rdfs:range xsd:decimal .
```

Employee dimension schema

```
# -- Employee dimension definition

nw:employeeDim a rdf:Property , qb:DimensionProperty ;
        rdfs:label "Employee Dimension"@en ;
        qb4o:hasHierarchy nw:supervision , nw:territories .

# -- Supervision hierarchy

nw:supervision a qb4o:HierarchyProperty ; rdfs:label "Supervision Hierarchy"@en ;
        qb4o:inDimension nw:employeeDim ;
        qb4o:hasLevel nw:employee .

_:supervision_hs1 a qb4o:HierarchyStep ; qb4o:inHierarchy nw:supervision ;
        qb4o:childLevel nw:employee ; qb4o:parentLevel nw:employee ;
        qb4o:pcCardinality qb4o:ManyToOne .

# -- Territories hierarchy

nw:territories a qb4o:HierarchyProperty ; rdfs:label "Territories Hierarchy"@en ;
        qb4o:inDimension nw:employeeDim ;
        qb4o:hasLevel nw:employee , nw:city , nw:state , nw:region , nw:country , nw:continent .
```

Employee dimension schema

```
# -- Territories hierarchy

_:territories_hs1 a qb4o:HierarchyStep ; qb4o:inHierarchy nw:territories ;
   qb4o:childLevel nw:employee ; qb4o:parentLevel nw:city ;
   qb4o:pcCardinality qb4o:ManyToMany .
_:territories_hs2 a qb4o:HierarchyStep ; qb4o:inHierarchy nw:territories ;
   qb4o:childLevel nw:city ; qb4o:parentLevel nw:state ;
   qb4o:pcCardinality qb4o:ManyToOne .
_:territories_hs3 a qb4o:HierarchyStep ; qb4o:inHierarchy nw:territories ;
   qb4o:childLevel nw:city ; qb4o:parentLevel nw:country ;
   qb4o:pcCardinality qb4o:ManyToOne .
_:territories_hs4 a qb4o:HierarchyStep ; qb4o:inHierarchy nw:territories ;
   qb4o:childLevel nw:state ; qb4o:parentLevel nw:region ;
   qb4o:pcCardinality qb4o:ManyToOne .
_:territories_hs5 a qb4o:HierarchyStep ; qb4o:inHierarchy nw:territories ;
   qb4o:childLevel nw:state ; qb4o:parentLevel nw:country ;
   qb4o:pcCardinality qb4o:ManyToOne .
_:territories_hs6 a qb4o:HierarchyStep ; qb4o:inHierarchy nw:territories ;
   qb4o:childLevel nw:region ; qb4o:parentLevel nw:country ;
   qb4o:pcCardinality qb4o:ManyToOne .
_:territories_hs7 a qb4o:HierarchyStep ; qb4o:inHierarchy nw:territories ;
   qb4o:childLevel nw:country ; qb4o:parentLevel nw:continent ;
   qb4o:pcCardinality qb4o:ManyToOne .
```

Employee dimension schema

```
# -- Employee level

nw:employee a qb4o:LevelProperty ;
        rdfs:label "Employee Level"@en ;
        qb4o:hasAttribute nw:employeeID ;
        qb4o:hasAttribute nw:firstName ;
        qb4o:hasAttribute nw:lastName ;
        qb4o:hasAttribute nw:title ;
        qb4o:hasAttribute nw:birthDate ;
        qb4o:hasAttribute nw:hireDate ;
        qb4o:hasAttribute nw:addressStreet ;
        qb4o:hasAttribute nw:addressCity ;
        qb4o:hasAttribute nw:addressRegion ;
        qb4o:hasAttribute nw:addressPostalCode ;
        qb4o:hasAttribute nw:addressCountry ;
        skos:broader nw:city , nw:employee .

nw:employeeID a qb4po:LevelAttribute ; rdfs:label "Employee ID"@en ;  rdfs:range xsd:positiveInteger .
nw:firstName a qb4o:LevelAttribute ;  rdfs:label "First Name"@en ; rdfs:range xsd:string .
nw:lastName a qb4o:LevelAttribute ;  rdfs:label "Last Name"@en ;  rdfs:range xsd:string .
nw:title a qb4o:LevelAttribute ;    rdfs:label "Title"@en ;  rdfs:range xsd:string .
nw:birthDate a qb4o:LevelAttribute ;  rdfs:label "Birth Date"@en ; rdfs:range xsd:date .
……
```

Employee dimension schema

```
# -- City level

nw:city a qb4o:LevelProperty ;
        rdfs:label "City Level"@en ;
        qb4o:hasAttribute nw:cityName ;
        skos:broader nw:state, nw:country .

# -- State level

nw:cityName a qb:LevelAttribute ;
        rdfs:label "City Name"@en ;  rdfs:range xsd:string .

nw:state a qb4o:LevelProperty ;
        rdfs:label "State Level"@en ;
        qb4o:hasAttribute nw:stateName ;
        qb4o:hasAttribute nw:englishStateName ;
        qb4o:hasAttribute nw:stateType ;
        qb4o:hasAttribute nw:stateCode ;
        qb4o:hasAttribute nw:stateCapital ;
        skos:broader nw:region, nw:country .
```

Employee dimension schema

```
nw:stateName a qb4o:LevelAttribute ;
        rdfs:label "State Name"@en ;  rdfs:range xsd:string .
nw:englishStateName a qb4o:LevelAttribute ;
        rdfs:label "English State Name"@en ;  rdfs:range xsd:string .
nw:stateType a qb:4oLevelAttribute ;
        rdfs:label "State Type"@en ; rdfs:range xsd:string .
nw:stateCode a qb4o:LevelAttribute ;
        rdfs:label "State Code"@en ;  rdfs:range xsd:string .
nw:stateCapital a qb4o:LevelAttribute ;
        rdfs:label "State Capital"@en ;   rdfs:range xsd:string .

# -- Region level

nw:region a qb4o:LevelProperty ;
        rdfs:label "Region Level"@en ;
        qb4o:hasAttribute nw:regionCode ;
        qb4o:hasAttribute nw:regionName ;
        skos:broader nw:country .

nw:regionCode a qb4o:LevelAttribute ;
        rdfs:label "Region Code"@en ;  rdfs:range xsd:string .
nw:regionName a qb4o:LevelAttribute ;   rdfs:label "Region Name"@en ;
        rdfs:range xsd:string .
```

Employee dimension schema

```
# -- Country level

nw:country a qb4o:LevelProperty ;
        rdfs:label "Country Level"@en ;
        qb4o:hasAttribute nw:countryName ;
        qb4o:hasAttribute nw:countryCode ;
        qb4o:hasAttribute nw:countryCapital ;
        qb4o:hasAttribute nw:population ;
        qb4o:hasAttribute nw:subdivision ;
        skos:broader nw:continent .

nw:countryName a qb:LevelAttribute ;
        rdfs:label "Country Name"@en ; rdfs:range xsd:string .
nw:countryCode a qb4o:LevelAttribute ;  rdfs:label "Country Code"@en ;
        rdfs:range xsd:string .
nw:countryCapital a qb4o:LevelAttribute ;
        rdfs:label "Country Capital"@en ;
        rdfs:range xsd:string .
nw:population a qb4o:LevelAttribute ;  rdfs:label "Population"@en ;
        rdfs:range xsd:positiveInteger .
nw:subdivision a qb4o:LevelAttribute ;   rdfs:label "Subdivision"@en ;
        rdfs:range xsd:string .
```

Employee dimension schema

```
# -- Continent level

nw:continent a qb4o:LevelProperty ;
   rdfs:label "Continent Level"@en ;
   qb4o:hasAttribute nw:continentName .

nw:continentName a qb4o:LevelAttribute ;
   rdfs:label "Continent Name"@en ;
   rdfs:range xsd:string .
```

Northwind DW: Instances

- Instances can be produced through R2RML, importing from a Relational DW
  - Input:
    - A mapping file
    - A ROLAP DW
  - Output:
    - Triples representing instances
- Triples produced by an engine (in this case, Spyder)

Instances: Employee dimension - mapping

```
<#TriplesMapEmployee> a rr:TriplesMap ;
rr:logicalTable [ rr:tableName "Employee" ] ;
rr:subjectMap [
   rr:termType rr:IRI ;
   rr:template "http://dwbook.org/cubes/instances/northwind/Employee#{EmployeeKey}" ;
] ;
rr:predicateObjectMap [
   rr:predicate qb4o:memberOf ;
   rr:object nw:employee ;
] ;
rr:predicateObjectMap [
   rr:predicate nw:employeeID ;
   rr:objectMap [ rr:column "EmployeeKey" ] ;
] ;
rr:predicateObjectMap [
   rr:predicate nw:firstName ;
   rr:objectMap [ rr:column "FirstName" ] ;
] ;
rr:predicateObjectMap [
   rr:predicate nw:lastName ;
   rr:objectMap [ rr:column "LastName" ] ;
] ;
```

Instances: Employee dimension - mapping

```
rr:predicateObjectMap [
   rr:predicate nw:title ;
   rr:objectMap [ rr:column "Title" ] ; ] ;
……
rr:predicateObjectMap [
   rr:predicate nw:addressCity ;
   rr:objectMap [ rr:column "City" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate  nw:addressRegion ;
   rr:objectMap [ rr:column "Region" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:addressPostalCode ;
   rr:objectMap [ rr:column "PostalCode" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:addressCountry ;
   rr:objectMap [ rr:column "Country" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate skos:broader ;
   rr:objectMap [
   rr:termType rr:IRI ;
   rr:template "http://dwbook.org/cubes/instances/northwind/Employee#{SupervisorKey}" ] ;
] .
```

Instances: Employee dimension - triples

<http://dwbook.org/cubes/**instances**/northwind/Employee#8>
<http://dwbook.org/cubes/**schemas**/northwind#birthDate> "1958-01-09" ;
<http://dwbook.org/cubes/schemas/northwind#addressPostalCode> "98105" ;
<http://dwbook.org/cubes/schemas/northwind#employeeID> 8 ;
<http://dwbook.org/cubes/schemas/northwind#addressRegion> "WA" ;
<http://www.w3.org/2004/02/skos/core#broader>
<http://dwbook.org/cubes/**instances**/northwind/City#125>,
<http://dwbook.org/cubes/instances/northwind/City#102>,
<http://dwbook.org/cubes/instances/northwind/City#120>,
<http://dwbook.org/cubes/instances/northwind/Employee#2>,
<http://dwbook.org/cubes/instances/northwind/City#79> ;
<http://dwbook.org/cubes/schemas/northwind#addressCountry> "USA" ;
<http://dwbook.org/cubes/schemas/northwind#hireDate> "1994-03-05" ;
<http://dwbook.org/cubes/schemas/northwind#lastName> "Callahan" ;
<http://dwbook.org/cubes/schemas/northwind#addressCity> "Seattle" ;
<http://dwbook.org/cubes/schemas/northwind#firstName> "Laura" ;
<http://dwbook.org/cubes/schemas/northwind#title> "Inside Sales Coordinator" ;
<http://dwbook.org/cubes/schemas/northwind#addressStreet> "4726 - 11th Ave. N.E." ;
<http://purl.org/qb4olap/cubes#memberOf> <http://dwbook.org/cubes/schemas/northwind#employee> .

…….

Northwind DW: Observations – mapping

```
<#TriplesMapSales> a rr:TriplesMap ;
# rr:logicalTable [ rr:tableName "Sales" ] ;
rr:logicalTable [
   rr:sqlQuery """
   SELECT CustomerKey, EmployeeKey, OrderDateKey, DueDateKey, ShippedDateKey, ShipperKey, ProductKey, SupplierKey,
         CAST(OrderNo AS NVARCHAR) AS OrderNo, CAST(OrderLineNo AS NVARCHAR) AS OrderLineNo, UnitPrice, Quantity, Discount,
         SalesAmount, Freight, SalesAmount-Freight AS NetAmount
   FROM Sales """ ;
   rr:sqlVersion rr:MSSQLServer ;
   rr:projectedColumns "CUSTOMERKEY EMPLOYEEKEY ORDERDATEKEY DUEDATEKEY SHIPPEDDATEKEY SHIPPERKEY PRODUCTKEY
   SUPPLIERKEY ORDERNO ORDERLINENO UNITPRICE QUANTITY DISCOUNT SALESAMOUNT FREIGHT NETAMOUNT" ; ] ;

rr:subjectMap [
   rr:termType rr:IRI ;
   rr:template "http://dwbook.org/cubes/instances/northwind/Sales#{OrderNo}:{OrderLineNo}" ;
   rr:class qb:Observation ; ] ;

rr:predicateObjectMap [
   rr:predicate qb:dataSet ;
   rr:object nw:dataset1 ; ] ;

rr:predicateObjectMap [
   rr:predicate nw:customer ;
   rr:objectMap [
   rr:termType rr:IRI ;
   rr:template "http://dwbook.org/cubes/instances/northwind/Customer#{CustomerKey}" ] ;
] ;
```

Northwind DW: Observations – mapping

```
rr:predicateObjectMap [
   rr:predicate nw:employee ;
   rr:objectMap [
   rr:termType rr:IRI ;
   rr:template "http://dwbook.org/cubes/instances/northwind/Employee#{EmployeeKey}" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:orderDate ;
   rr:objectMap [
   rr:termType rr:IRI ;
   rr:template "http://dwbook.org/cubes/instances/northwind/Time#{OrderDateKey}" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:dueDate ;
   rr:objectMap [
   rr:termType rr:IRI ;
   rr:template "http://dwbook.org/cubes/instances/northwind/Time#{DueDateKey}" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:shippedDate ;
   rr:objectMap [
   rr:termType rr:IRI ;
   rr:template "http://dwbook.org/cubes/instances/northwind/Time#{ShippedDateKey}" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:shipper ;
   rr:objectMap [
   rr:termType rr:IRI ;
   rr:template "http://dwbook.org/cubes/instances/northwind/Shipper#{ShipperKey}" ] ; ] ;
```

Northwind DW: Observations – mapping

```
rr:predicateObjectMap [
   rr:predicate nw:product ;
   rr:objectMap [
   rr:termType rr:IRI ;
   rr:template "http://dwbook.org/cubes/instances/northwind/Product#{ProductKey}" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:supplier ;
   …
   rr:template "http://dwbook.org/cubes/instances/northwind/Supplier#{SupplierKey}" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:order ;
    …
   rr:template "http://dwbook.org/cubes/instances/northwind/Order#{OrderNo}:{OrderLineNo}" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:quantity ;
   rr:objectMap [ rr:column "Quantity" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:unitPrice ;
   rr:objectMap [ rr:column "UnitPrice" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:discount ;
   rr:objectMap [ rr:column "Discount" ] ; ] ;
rr:predicateObjectMap [
   rr:predicate nw:salesAmount ;
   rr:objectMap [ rr:column "SalesAmount" ] ; ] ;
….
] .
```

Northwind DW: Observations - triples

<http://dwbook.org/cubes/instances/northwind/Sales#10619:1>
a <http://purl.org/linked-data/cube#Observation> ;
<http://purl.org/linked-data/cube#dataSet> <http://dwbook.org/cubes/instances/northwind#dataset1> ;
<http://dwbook.org/cubes/schemas/northwind#product>
<http://dwbook.org/cubes/instances/northwind/Product#21> ;
<http://dwbook.org/cubes/schemas/northwind#order>
<http://dwbook.org/cubes/instances/northwind/Order#10619:1> ;
<http://dwbook.org/cubes/schemas/northwind#employee>
<http://dwbook.org/cubes/instances/northwind/Employee#3>
<http://dwbook.org/cubes/schemas/northwind#supplier>
<http://dwbook.org/cubes/instances/northwind/Supplier#8> ;
<http://dwbook.org/cubes/schemas/northwind#dueDate>
<http://dwbook.org/cubes/instances/northwind/Time#428> ;
<http://dwbook.org/cubes/schemas/northwind#shipper>
<http://dwbook.org/cubes/instances/northwind/Shipper#3> ;
<http://dwbook.org/cubes/schemas/northwind#shippedDate>
<http://dwbook.org/cubes/instances/northwind/Time#403> ;
<http://dwbook.org/cubes/schemas/northwind#customer>
<http://dwbook.org/cubes/instances/northwind/Customer#315> ;
<http://dwbook.org/cubes/schemas/northwind#orderDate>
<http://dwbook.org/cubes/instances/northwind/Time#400> ;
<http://dwbook.org/cubes/schemas/northwind#quantity> 42 ;
<http://dwbook.org/cubes/schemas/northwind#unitPrice> "10.0000"^^xsd:decimal ;
<http://dwbook.org/cubes/schemas/northwind#salesAmount> "420.0000"^^xsd:decimal ;
<http://dwbook.org/cubes/schemas/northwind#freight> "45.5250"^^xsd:decimal ;
<http://dwbook.org/cubes/schemas/northwind#discount> "0.0"^^xsd:float ;
<http://dwbook.org/cubes/schemas/northwind#netAmount> "374.4750"^^xsd:decimal ;

How can we use QB4OLAP ?

.

(3) Building a QB4OLAP cube from scratch

How can we use QB4OLAP ?

.

(3) Building a QB4OLAP cube from scratch

The simplest case: Same techniques applied...

Outline

- The Semantic Web
- RDF and SPARQL Basics
- Vocabularies for OLAP on the SW: QB and QB4OLAP
- Modeling Data Cubes on the Semantic Web using QB4OLAP
- **Querying Data Cubes on the Semantic Web**
- Summary

Querying Data Cubes: Cube Algebra

- Ultimate goal: a user-oriented model for OLAP: A data model based on cubes
  - High-level OLAP query language needed
  - Not possible without dimension structure
  - QB only represents dimension instances, only “manual” SPARQL queries

- Algebra based on Ciferri et al., *Cube Algebra: A Generic User-Centric Model and Query Language for OLAP Cubes*. IJDWM 9(2): 39-65 (2013)

- Cube Algebra queries automatically translated to SPARQL (after simplification)

- Example: *Market capitalization by geographical region*

```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>;
PREFIX data: <http://worldbank.270a.info/dataset/>;
$C1 := SLICE (data:CM.MKT.LCAP.CD, schema:timeDim);
$C2 := ROLLUP ($C1, schema:geoDim, schema:region);
```

Cube Algebra simplification

- Elimination of redundancy
- Equivalence rules defined
- Produces a query, equivalent to the original one
- Rules (If query does not contain DICE operators)

  - **Rule 1**: Replace all the operations that use ROLLUP and DRILLDOWN over the same dimension, with a single ROLLUP from the bottom level of the dimension
  - **Rule 2**: When there is a SLICE, and a sequence of ROLLUP and DRILLDOWN operations over a certain dimension, delete the sequence of ROLLUPs and DRILLDOWNs and keep only the SLICE over this dimension
  - **Rule 3**: Perform SLICE operations as soon as posible

Cube Algebra simplification

- If query contains DICE
  - We assume queries are of the form
    (ROLLUP | SLICE |DRILLDOWN)* (DICE)*
- DICE always at the end
- We can split the query:
  - One part does not contain DICE statement, the other one does
  - We apply rules over the first part

Cube Algebra simplification

- Example (Eurostat): Asylum applications by year and continent - **Query 1**

```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
QUERY
$C1 := ROLLUP (data:migr_asyappctzm, schema:citizenshipDim, schema:continent);
$C2 := ROLLUP ($C1, schema:citizenshipDim, schema:citAll);
$C3 := ROLLUP ($C2, schema:timeDim, schema:year);
$C4 := DRILLDOWN ($C3, schema:citizenshipDim, schema:continent);
```

Redundant query – Simplified using Rule 1

```
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
QUERY
$C1 := ROLLUP (data:migr_asyappctzm, schema:citizenshipDim, schema:continent);
$C2 := ROLLUP ($C1, schema:timeDim, schema:year);
```

Cube Algebra simplification

- Another example (Eurostat): Asylum applications by year – **Query 2**

```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
QUERY
$C1 := ROLLUP (data:migr_asyappctzm, schema:citizenshipDim, schema:continent);
$C2 := SLICE ($C1, schema:citizenshipDim);
$C3 := ROLLUP ($C2, schema:timeDim, schema:year);
```

Rule 2, avoids the ROLLUP over schema:citizenshipDim dimension

```
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
QUERY
$C1 := SLICE(data:migr_asyappctzm, schema:citizenshipDim);
$C2 := ROLLUP($C1,schema:timeDim,schema:year);
```

Cube Algebra simplification

- Another example (Eurostat): Example with the WBLD – Yearly Market capitalization by income level - **Query 3**

```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>;
PREFIX data: <http://worldbank.270a.info/dataset/>;
QUERY
$C1 := ROLLUP (<http://worldbank.270a.info/dataset/CM.MKT.LCAP.CD>,
                    schema:geoDim,schema:geoAll);
$C2 := DRILLDOWN ($C1, schema:geoDim, schema:income);
```

The simplification gives

```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>;
PREFIX data: <http://worldbank.270a.info/dataset/>;
QUERY
$C1 := ROLLUP (data:CM.MKT.LCAP.CD, schema:geoDim, schema:income) ;
```

Cube Algebra simplification

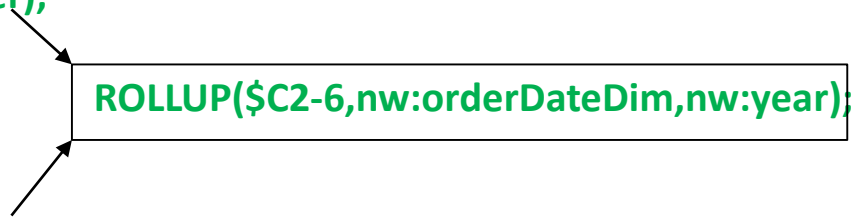
- A more complex query, over the Northwind DW: Total sales amount per customer, year, and product category (Query 4.1 in VZ book) - **Query 4**

```
$C1 := ROLLUP(data:dataset1, nw:customerDim, nw:city);
$C2 := ROLLUP($C1,nw:orderDateDim, nw:quarter);
$C3 := SLICE($C2, nw:shippedDateDim);
$C4 := SLICE($C3, nw:supplierDim);
$C5 := SLICE($C4, nw:dueDateDim);
$C6 := ROLLUP($C5,nw:orderDateDim, nw:year);
$C7 := SLICE($C6, nw:employeeDim);
$C8 := ROLLUP($C7,nw:productDim, nw:category);
$C9 := SLICE($C8,  nw:quantity);
$C10 := SLICE($C9,  nw:unitPrice);
$C11 := SLICE($C10,  nw:discount);
$C12 := SLICE($C11,  nw:freight);
$C13 := SLICE($C12,  nw:netAmount);
$C14 := DRILLDOWN($C13,nw:customerDim, nw:customer);
$C15 := SLICE($C14, nw:orderDim);
$C16 := SLICE($C15, nw:shipperDim);
```

Cube Algebra simplification

- A more complex query, over the Northwind DW: Total sales amount per customer, year, and product category (Query 4.1 in VZ book) - **Query 4**

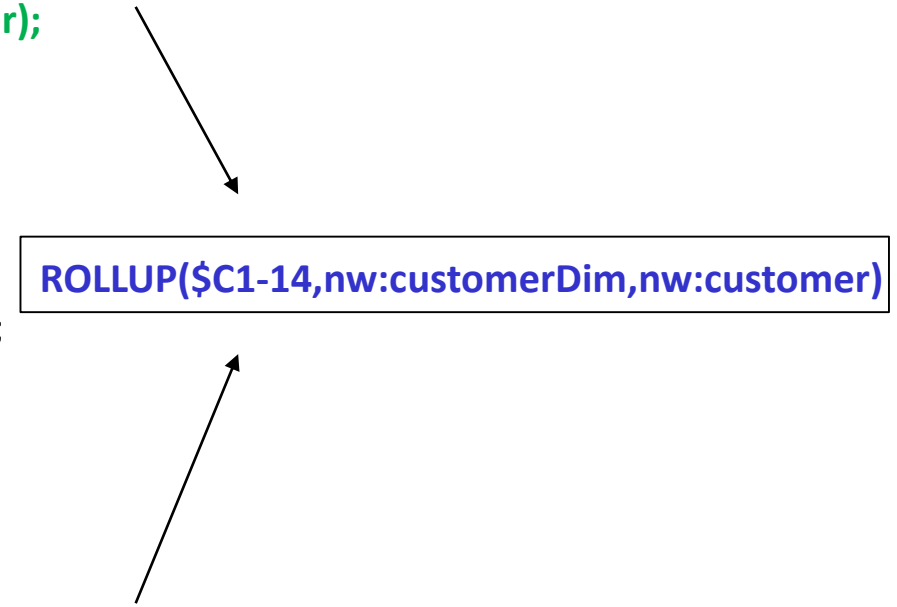
```
$C1 := ROLLUP(data:dataset1, nw:customerDim, nw:city);
$C2 := ROLLUP($C1,nw:orderDateDim, nw:quarter);
$C3 := SLICE($C2, nw:shippedDateDim);
$C4 := SLICE($C3, nw:supplierDim);
$C5 := SLICE($C4, nw:dueDateDim);
$C6 := ROLLUP($C5,nw:orderDateDim, nw:year);
$C7 := SLICE($C6, nw:employeeDim);
$C8 := ROLLUP($C7,nw:productDim, nw:category);
$C9 := SLICE($C8, nw:quantity);
$C10 := SLICE($C9,  nw:unitPrice);
$C11 := SLICE($C10,  nw:discount);
$C12 := SLICE($C11,  nw:freight);
$C13 := SLICE($C12,  nw:netAmount);
$C14 := DRILLDOWN($C13,nw:customerDim, nw:customer);
$C15 := SLICE($C14, nw:orderDim);
$C16 := SLICE($C15, nw:shipperDim);
```



Cube Algebra simplification

- A more complex query, over the Northwind DW: Total sales amount per customer, year, and product category (Query 4.1 in VZ book) - **Query 4**

```
$C1 := ROLLUP(data:dataset1, nw:customerDim, nw:city);
$C2 := ROLLUP($C1,nw:orderDateDim, nw:quarter);
$C3 := SLICE($C2, nw:shippedDateDim);
$C4 := SLICE($C3, nw:supplierDim);
$C5 := SLICE($C4, nw:dueDateDim);
$C6 := ROLLUP($C5,nw:orderDateDim, nw:year);
$C7 := SLICE($C6, nw:employeeDim);
$C8 := ROLLUP($C7,nw:productDim, nw:category);
$C9 := SLICE($C8,  nw:quantity);
$C10 := SLICE($C9,  nw:unitPrice);
$C11 := SLICE($C10,  nw:discount);
$C12 := SLICE($C11,  nw:freight);
$C13 := SLICE($C12, nw:netAmount);
$C14 := DRILLDOWN($C13,nw:customerDim, nw:customer);
$C15 := SLICE($C14, nw:orderDim);
$C16 := SLICE($C15, nw:shipperDim);
```



Cube Algebra simplification

- Total sales amount per customer, year, product category - **Query 4 Simplified**
- Replace ROLLUPS, put all together

```
$C1 := SLICE(data:dataset1, MEASURES(nw:quantity));
$C2 := SLICE($C1, MEASURES(nw:unitPrice));
$C3:= SLICE($C2, MEASURES(nw:discount));
$C4:= SLICE($C3, MEASURESn(w:freight));
$C5:= SLICE($C4, MEASURES(nw:netAmount));
$C6:= SLICE($C5, nw:dueDateDim);
$C7:= SLICE($C6, nw:employeeDim);
$C8:= SLICE($C7, nw:shippedDateDim);
$C9:= SLICE($C8, nw:supplierDim);
$C10:= SLICE($C9, nw:orderDim);
$C11:= SLICE($C10, nw:shipperDim);
$C12:= ROLLUP($C11,nw:customerDim,nw:customer)
$C13:= ROLLUP($C12,nw:orderDateDim,nw:year);
$C14:= ROLLUP($C13,nw:productDim,nw:category);
```

Cube Algebra simplification

- Another query over NW DW: Yearly sales amount for each pair of customer country and supplier countries (Query 4.2 in VZ book) – **Query 5**

```
$C1 := ROLLUP(<http://dwbook.org/cubes/instances/northwind#dataset1>,nw:customerDim, nw:city);
$C2 := ROLLUP($C1,nw:orderDateDim, nw:quarter);
$C3 := SLICE($C2, nw:shippedDateDim);
$C4 := ROLLUP($C3, nw:supplierDim, nw:country);
$C5 := SLICE($C4, nw:dueDateDim);
$C6 := ROLLUP($C5,nw:orderDateDim, nw:year);
$C7 := SLICE($C6, nw:employeeDim);
$C8 := SLICE($C7,nw:productDim);
$C9 := SLICE($C8, MEASURES(nw:quantity));
$C10 := SLICE($C9, MEASURES(nw:unitPrice));
$C11 := SLICE($C10, MEASURES(nw:discount));
$C12 := SLICE($C11, MEASURES(nw:freight));
$C13 := SLICE($C12, MEASURES(nw:netAmount));
$C14 := ROLLUP($C13,nw:customerDim, nw:country);
$C15 := SLICE($C14, nw:orderDim);
$C16 := SLICE($C15, nw:shipperDim);
```

Cube Algebra simplification

- Another query over NW DW: Yearly sales amount for each pair of customer country and supplier country (Query 4.2 in VZ book) – **Query 5**

```
$C1 := ROLLUP(data:dataset1, nw:customerDim, nw:city);
$C2 := ROLLUP($C1,nw:orderDateDim, nw:quarter);
$C3 := SLICE($C2, nw:shippedDateDim);
$C4 := ROLLUP($C3, nw:supplierDim, nw:country);
$C5 := SLICE($C4, nw:dueDateDim);
$C6 := ROLLUP($C5,nw:orderDateDim, nw:year);
$C7 := SLICE($C6, nw:employeeDim);
$C8 := SLICE($C7,nw:productDim);
$C9 := SLICE($C8, nw:quantity);
$C10 := SLICE($C9, nw:unitPrice);
$C11 := SLICE($C10, nw:discount);
$C12 := SLICE($C11, nw:freight);
$C13 := SLICE($C12, nw:netAmount);
$C14 := ROLLUP($C13,nw:customerDim, nw:country);
$C15 := SLICE($C14, nw:orderDim);
$C16 := SLICE($C15, nw:shipperDim);
```

Cube Algebra simplification

- Yearly sales amount for each pair of customer country and supplier country – **Query 5 Simplified**

```
$C1:= SLICE(data:dataset1, MEASURES(nw:quantity));
$C2:= SLICE($C1, MEASURES(nw:unitPrice));
$C3:= SLICE($C2, MEASURES(nw:discount));
$C4:= SLICE($C3, MEASURES((nw:freight));
$C5:= SLICE($C4, MEASURES(nw:netAmount));
$C6:= SLICE($C5,nw:dueDateDim);
$C7:= SLICE($C6,nw:employeeDim);
$C8:= SLICE($C7,nw:productDim);
$C9:= SLICE($C8,nw:shippedDateDim);
$C10:= SLICE($C9,nw:orderDim);
$C11:= SLICE($C10,nw:shipperDim);
$C12:=ROLLUP($C11,nw:customerDim,nw:country);
$C13:=ROLLUP($C12,nw:orderDateDim,nw:year);
$C14:=ROLLUP($C13,nw:supplierDim,nw:country);
```

Cube Algebra simplification

- Another query over NW DW: Total sales amount, and sum of the quantities sold for each order. (Variation of Query 4.13 in VZ book) – **Query 6**

```
$C1 := ROLLUP(data:dataset1, nw:customerDim, nw:city);
$C2 := ROLLUP($C1,nw:orderDateDim, nw:quarter);
$C3 := SLICE($C2, nw:shippedDateDim);
$C4 := ROLLUP($C3, nw:supplierDim, nw:country);
$C5 := SLICE($C4, nw:dueDateDim);
$C6 := SLICE($C5,nw:orderDateDim);
$C7 := SLICE($C6, nw:employeeDim);
$C8 := SLICE($C7,nw:productDim);
$C9 := SLICE($C10, nw:unitPrice);
$C10 := SLICE($C11, nw:discount);
$C11 := SLICE($C12, nw:freight);
$C12 := SLICE($C13, nw:netAmount);
$C13 := SLICE($C12, nw:shipperDim);
$C14 := SLICE($C13, nw:customerDim);
$C15 := SLICE($C14, nw:supplierDim);
```

Cube Algebra simplification

- Another query over NW DW: Total sales amount, and sum of the quantities sold for each order. (Variation of Query 4.13 in VZ book) – **Query 6**

```
$C1 := ROLLUP(data:dataset1, nw:customerDim, nw:city);
$C2 := ROLLUP($C1,nw:orderDateDim, nw:quarter);
$C3 := SLICE($C2, nw:shippedDateDim);
$C4 := ROLLUP($C3, nw:supplierDim, nw:country);
$C5 := SLICE($C4, nw:dueDateDim);
$C6 := SLICE($C5,nw:orderDateDim);
$C7 := SLICE($C6, nw:employeeDim);
$C8 := SLICE($C7,nw:productDim);
$C9 := SLICE($C10, nw:unitPrice);
$C10 := SLICE($C11, nw:discount);
$C11 := SLICE($C12, nw:freight);
$C12 := SLICE($C13, nw:netAmount);
$C13 := SLICE($C12, nw:shipperDim);
$C14 := SLICE($C13, nw:customerDim);
$C15 := SLICE($C14, nw:supplierDim);
```

Cube Algebra simplification

- Total sales amount, and sum of the quantities sold for each order - **Query 6 Simplified**

```
$C1 := SLICE(data:dataset1, MEASURES(nw:unitPrice));
$C2 := SLICE($C1, MEASURES(nw:discount));
$C3 := SLICE($C2, MEASURES(nw:freight));
$C4 := SLICE($C3, MEASURES(nw:netAmount));
$C5 := SLICE($C4, nw:customerDim);
$C6 := SLICE($C5, nw:dueDateDim);
$C7 := SLICE($C6, nw:employeeDim);
$C8 := SLICE($C7, nw:orderDateDim);
$C9 := SLICE($C8, nw:productDim);
$C10 := SLICE($C9, nw:shippedDateDim);
$C11 := SLICE($C10, nw:supplierDim);
$C12 := SLICE($C11, nw:shipperDim);
```

Cube Algebra simplification

- Queries containing DICE (Eurostat): Asylum applications submitted by African citizens, with destination France - **Query 7**

```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
PREFIX property: <http://eurostat.linked-statistics.org/property#>;
QUERY
$C1 := ROLLUP (data:migr_asyappctzm, schema:citizenshipDim,schema:continent);
$C2 := SLICE ($C1, schema:asylappDim);
$C3 := ROLLUP ($C2, schema:timeDim, schema:year);
$C4 := SLICE ($C3, schema:asylappDim);
$C5 := SLICE ($C4, schema:ageDim);
$C6 := SLICE ($C5, schema:sex);
$C7 := DICE ($C6, (schema:citizenshipDim|schema:continent|schema:continentName = "Africa"));
$C8 := DICE ($C7, (schema:destinationDim|property:geo|schema:countryName = "France"));
```

Cube Algebra simplification

- Queries containing DICE (Eurostat): Asylum applications submitted in 2013 by African citizens, with destination France - **Query 7 simplified**

The simplification returns:

```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
PREFIX property: <http://eurostat.linked-statistics.org/property#>;
QUERY
$C1 := SLICE (data:migr_asyappctzm, schema:ageDim);
$C2 := SLICE ($C1, schema:sex);
$C3 := SLICE ($C2, schema:asylappDim);
$C4 := ROLLUP ($C3, schema:timeDim, schema:year);
$C5 := ROLLUP ($C4, schema:citizenshipDim,schema:continent);
$C6 := DICE ($C5, (schema:citizenshipDim|schema:continent|schema:continentName = "Africa"));
$C7 := DICE ($C6, (schema:destinationDim|property:geo|schema:countryName = "France"));
```

Cube Algebra simplification

- Queries containing DICE (Eurostat): Asylum applications by year submitted by Asian citizens, where applications count > 5000 with destination France or Germany - **Query 8**

```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
PREFIX property: <http://eurostat.linked-statistics.org/property#>;
PREFIX sdmx-measure:<http://purl.org/linked-data/sdmx/2009/measure#> ;
QUERY
$C1 := ROLLUP (data:migr_asyappctzm, schema:citizenshipDim,schema:citAll);
$C2 := ROLLUP ($C1, schema:timeDim, schema:year);
$C3 := DRILLDOWN ($C2, schema:citizenshipDim,schema:continent);
$C4 := DICE ($C3, (schema:citizenshipDim|schema:continent|schema:continentName = "Asia"));
$C5 := DICE ($C4, ( sdmx-measure:obsValue > 5000 AND
            (schema:destinationDim|property:geo|schema:countryName = "France") OR
            (schema:destinationDim|property:geo|schema:countryName = "Germany")));
```

Cube Algebra simplification

- Queries containing DICE (Eurostat): Asylum applications by year submitted by Asian citizens, where applications count > 5000 with destination France or Germany **Query 8 simplified**

The simplification returns:

```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
PREFIX property: http://eurostat.linked-statistics.org/property#>;
PREFIX sdmx-measure:<http://purl.org/linked-data/sdmx/2009/measure#> ;
QUERY
$C1 := ROLLUP (data:migr_asyappctzm, schema:citizenshipDim,schema:citAll);
$C2 := ROLLUP ($C1, schema:timeDim, schema:year);
$C3 := DRILLDOWN ($C2, schema:citizenshipDim,schema:continent);
$C4 := DICE ($C3, (schema:citizenshipDim|schema:continent|schema:continentName = "Asia"));
$C5 := DICE ($C4, ( sdmx-measure:obsValue > 5000 AND
            (schema:destinationDim|property:geo|schema:countryName = "France") OR
            (schema:destinationDim|property:geo|schema:countryName = "Germany")));
```

Cube Algebra to SPARQL translation

Goal: Develop a mechanism to translate a sequence a Cube Algebra expressions into a SPARQL query

- Two cases:
  - Cube Algebra queries that do not contain DICE operations (1)
  - Cube Algebra queries that contain DICE operations (2)

Cube Algebra to SPARQL translation

(1) We apply the rules, s.t., for each dimension in the data cube, only one of the following conditions holds:

- No operation is performed over dimension D
- A ROLLUP operation is performed over dimension D
  - Implemented navigating the roll-up relationships between members, guided by the dimension hierarchy
- A SLICE operation is performed over dimension D
  - “Slice-out” dimensions, after aggregating measures up to the ALL level of the dimension (similar mechanism as in ROLLUP)

Cube Algebra to SPARQL translation

(2) Rules have already been applied to the first part of the query

- The second part of the query contains only DICE operations
- Each DICE is associated with a condition over measures and/or attribues
- We implement DICE conditions using SPARQL FILTER clauses

Steps:

1. Apply the previous method to produce a SPARQL query for the non-diced part
   - We call this the “inner query”
2. Create a new SPARQL query (“outer query”) s.t.:

   (a) The SELECT clause has the same variables than the one of the inner query

   (b) The WHERE clause of the outer query contains:
   - The inner query
   - Graph patterns to obtain the values of the attributes in DICE conditions
   - A FILTER clause with the conjunction of the conditions of all the DICE operations

Cube Algebra to SPARQL translation

Simplified Query 1

- Asylum applications by year and continent

```
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
QUERY
$C1 := ROLLUP (data:migr_asyappctzm, schema:citizenshipDim, schema:continent);
$C2 := ROLLUP ($C1, schema:timeDim, schema:year);
```

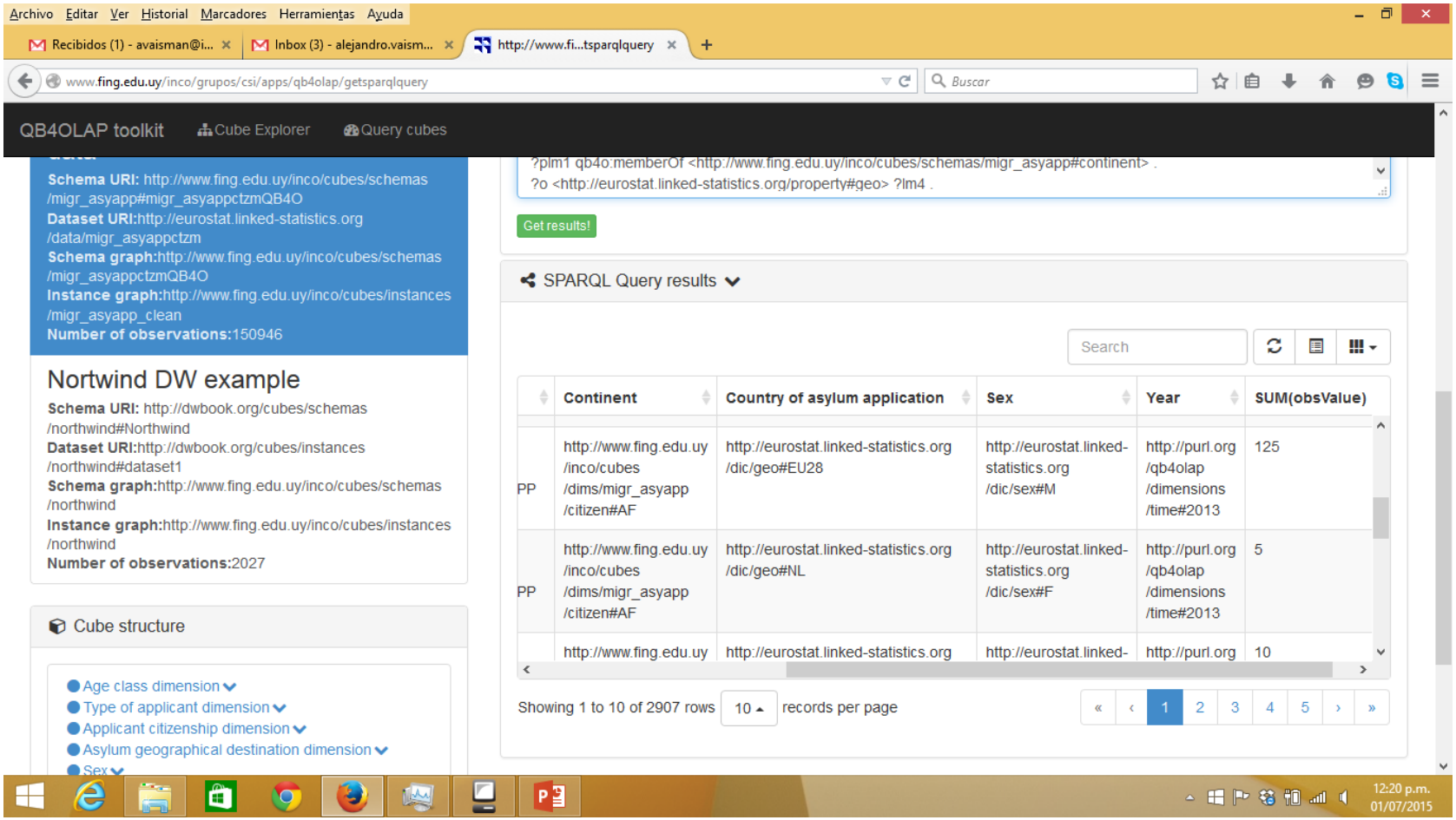
Cube Algebra to SPARQL translation

Query 1 Asylum applications by **year** and **continent**

```
SELECT ?lm1 ?lm2 ?plm1 ?lm4 ?lm5 ?plm2
        (SUM(<http://www.w3.org/2001/XMLSchema#integer>(?m1)) as ?ag1)
FROM <http://www.fing.edu.uy/inco/cubes/instances/migr_asyapp_clean>
FROM <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyappctzmQB4O>
 WHERE { ?o a qb:Observation .
?o qb:dataSet <http://eurostat.linked-statistics.org/data/migr_asyappctzm> .
?o <http://purl.org/linked-data/sdmx/2009/measure#obsValue> ?m1 .
?o <http://eurostat.linked-statistics.org/property#age> ?lm1 .
?o <http://eurostat.linked-statistics.org/property#asyl_app> ?lm2 .
?o <http://eurostat.linked-statistics.org/property#geo> ?lm4 .
?o <http://eurostat.linked-statistics.org/property#sex> ?lm5 .
?o <http://eurostat.linked-statistics.org/property#citizen> ?lm3 .
?lm3 qb4o:memberOf <http://eurostat.linked-statistics.org/property#citizen> .  ?lm3 skos:broader ?plm1 .
?plm1 qb4o:memberOf <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#continent> .
?o <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> ?lm6 .
?lm6 qb4o:memberOf <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> .
?lm6 skos:broader ?plm2 . ?plm2 qb4o:memberOf
        <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#year> . }
GROUP BY ?lm1 ?lm2 ?plm1 ?lm4 ?lm5 ?plm2
```

Bottom levels to be displayed

Cube Algebra to SPARQL translation



Cube Algebra to SPARQL translation

Simplified Query 1, dropping the dimensions not in the answer:

- Asylum applications by year and continent

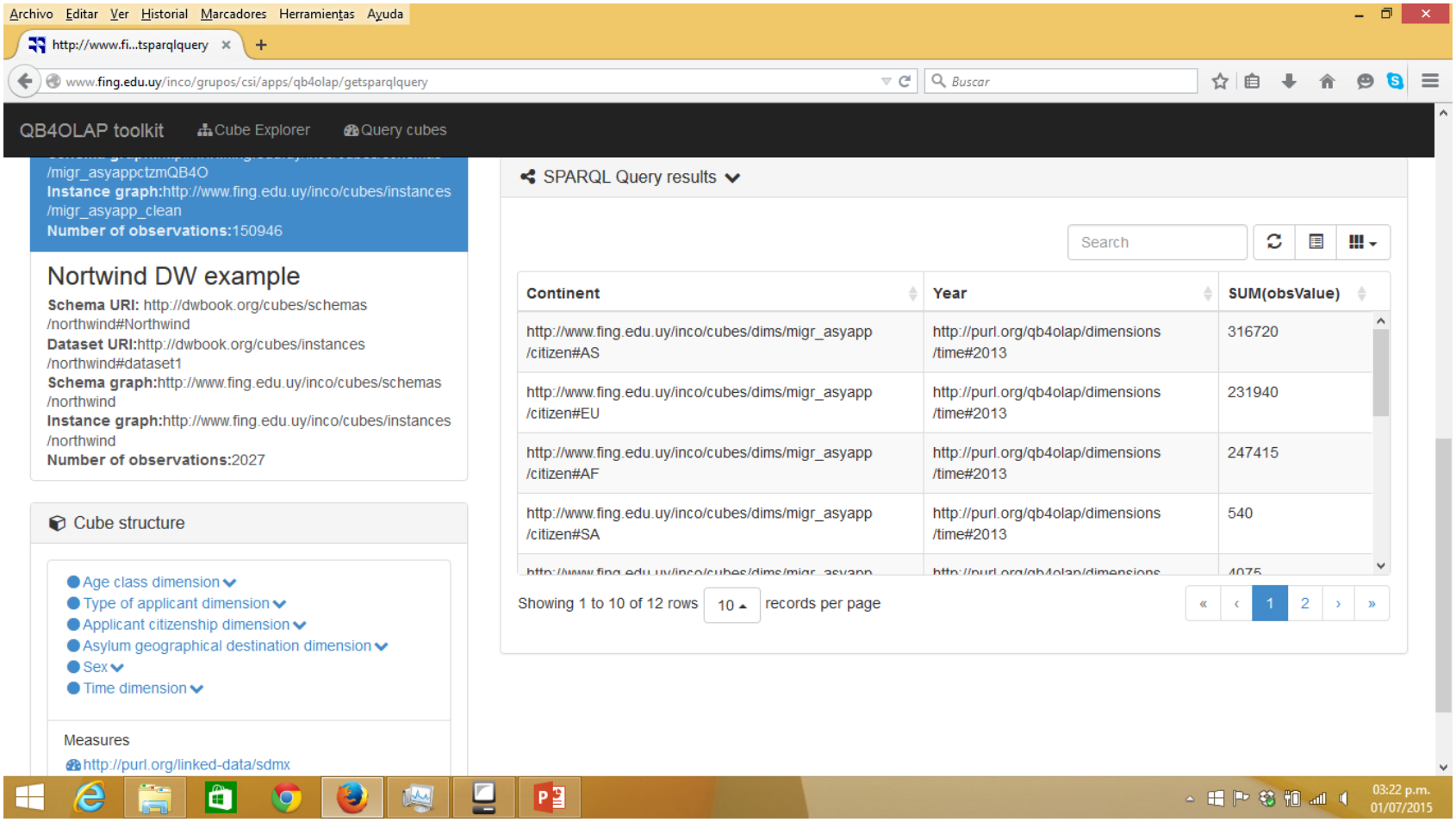
```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
QUERY
$C1 := ROLLUP
(<http://eurostat.linked-statistics.org/data/migr_asyappctzm>,
schema:citizenshipDim,schema:continent);
$C2 := SLICE($C1, schema:ageDim);
$C3 := SLICE($C2, schema:destinationDim);
$C4 := SLICE($C3, schema:sex);
$C5 := ROLLUP ($C4, schema:timeDim, schema:year);
$C6 := SLICE($C5, schema:asylappDim);
```

Cube Algebra to SPARQL translation

Query 1 dropping the dimensions not in the answer:

```
SELECT ?plm1 ?plm2  (SUM(<http://www.w3.org/2001/XMLSchema#integer>(?m1)) as ?ag1)
FROM <http://www.fing.edu.uy/inco/cubes/instances/migr_asyapp_clean>
FROM <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyappctzmQB4O>
 WHERE { ?o a qb:Observation .
?o qb:dataSet <http://eurostat.linked-statistics.org/data/migr_asyappctzm> .
?o <http://purl.org/linked-data/sdmx/2009/measure#obsValue> ?m1 .
?o <http://eurostat.linked-statistics.org/property#citizen> ?lm1 .
?lm1 qb4o:memberOf <http://eurostat.linked-statistics.org/property#citizen> .
?lm1 skos:broader ?plm1 .
?plm1 qb4o:memberOf <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#continent> .
?o <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> ?lm2 .
?lm2 qb4o:memberOf <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> .
?lm2 skos:broader ?plm2 .
?plm2 qb4o:memberOf <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#year> .
 }
GROUP BY ?plm1 ?plm2
```

Cube Algebra to SPARQL translation



Cube Algebra to SPARQL translation

Simplified Query 2

- Asylum applications by year

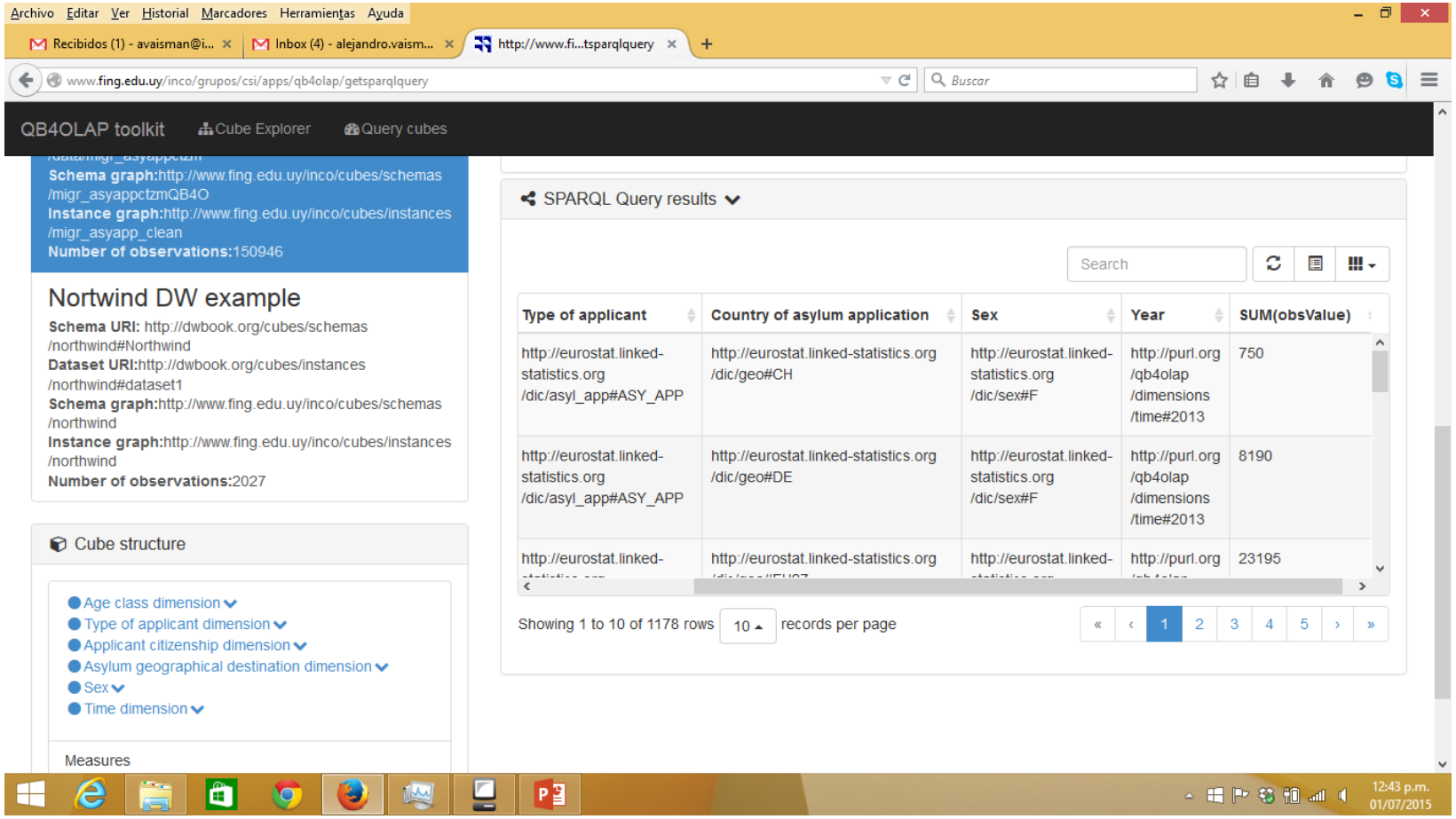
```
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
QUERY
$C1 := SLICE(data:migr_asyappctzm,D(schema:citizenshipDim));
$C2 := ROLLUP($C1,schema:timeDim,schema:year);
```

Cube Algebra to SPARQL translation

Query 2 Asylum applications by **year**

```
SELECT ?lm1 ?lm2 ?lm3 ?lm4 ?plm1
        (SUM(<http://www.w3.org/2001/XMLSchema#integer>(?m1)) as ?ag1)
FROM <http://www.fing.edu.uy/inco/cubes/instances/migr_asyapp_clean>
FROM <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyappctzmQB4O>
 WHERE { ?o a qb:Observation .
?o qb:dataSet <http://eurostat.linked-statistics.org/data/migr_asyappctzm> .
?o <http://purl.org/linked-data/sdmx/2009/measure#obsValue> ?m1 .
?o <http://eurostat.linked-statistics.org/property#age> ?lm1 .
?o <http://eurostat.linked-statistics.org/property#asyl_app> ?lm2 .
?o <http://eurostat.linked-statistics.org/property#geo> ?lm3 .
?o <http://eurostat.linked-statistics.org/property#sex> ?lm4 .
?o <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> ?lm5 .
?lm5 qb4o:memberOf <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> .
?lm5 skos:broader ?plm1 .
?plm1 qb4o:memberOf <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#year> .
 }
GROUP BY ?lm1 ?lm2 ?lm3 ?lm4 ?plm1
```

Cube Algebra to SPARQL translation



Cube Algebra to SPARQL translation

Simplified Query 3

- WBLD – Yearly Market capitalization by income level

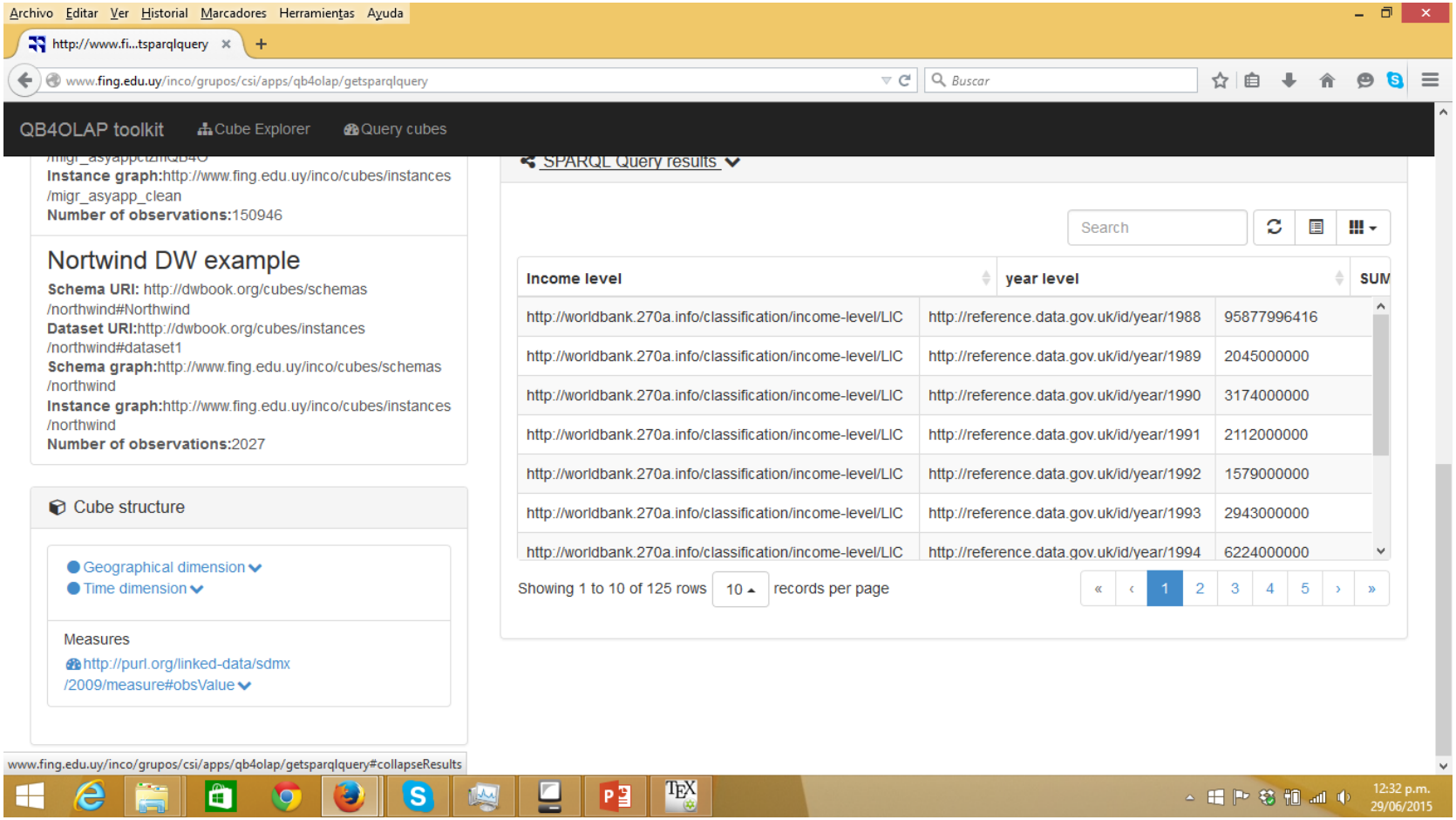
```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#>;
PREFIX data: <http://worldbank.270a.info/dataset/>;
QUERY
$C1 := ROLLUP (data:CM.MKT.LCAP.CD, schema:geoDim, schema:income)
```

Cube Algebra to SPARQL translation

Query 3

```
SELECT ?plm1 ?lm2  (SUM(?m1) as ?ag1)
FROM <http://www.fing.edu.uy/inco/cubes/instances/wbld>
FROM <http://www.fing.edu.uy/inco/cubes/schemas/wbld>
WHERE { ?o a qb:Observation .
 ?o qb:dataSet <http://worldbank.270a.info/dataset/CM.MKT.LCAP.CD> .
 ?o <http://purl.org/linked-data/sdmx/2009/measure#obsValue> ?m1 .
 ?o <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> ?lm2 .
 ?o <http://purl.org/linked-data/sdmx/2009/dimension#refArea> ?lm1 .
 ?lm1 qb4o:memberOf <http://purl.org/linked-data/sdmx/2009/dimension#refArea> .
 ?lm1 skos:broader ?plm1 .
 ?plm1 qb4o:memberOf <http://www.fing.edu.uy/inco/cubes/schemas/world-bank-indicators#income> .
}
GROUP BY ?plm1 ?lm2
ORDER  BY ?plm1 ?lm2
```

Cube Algebra to SPARQL translation

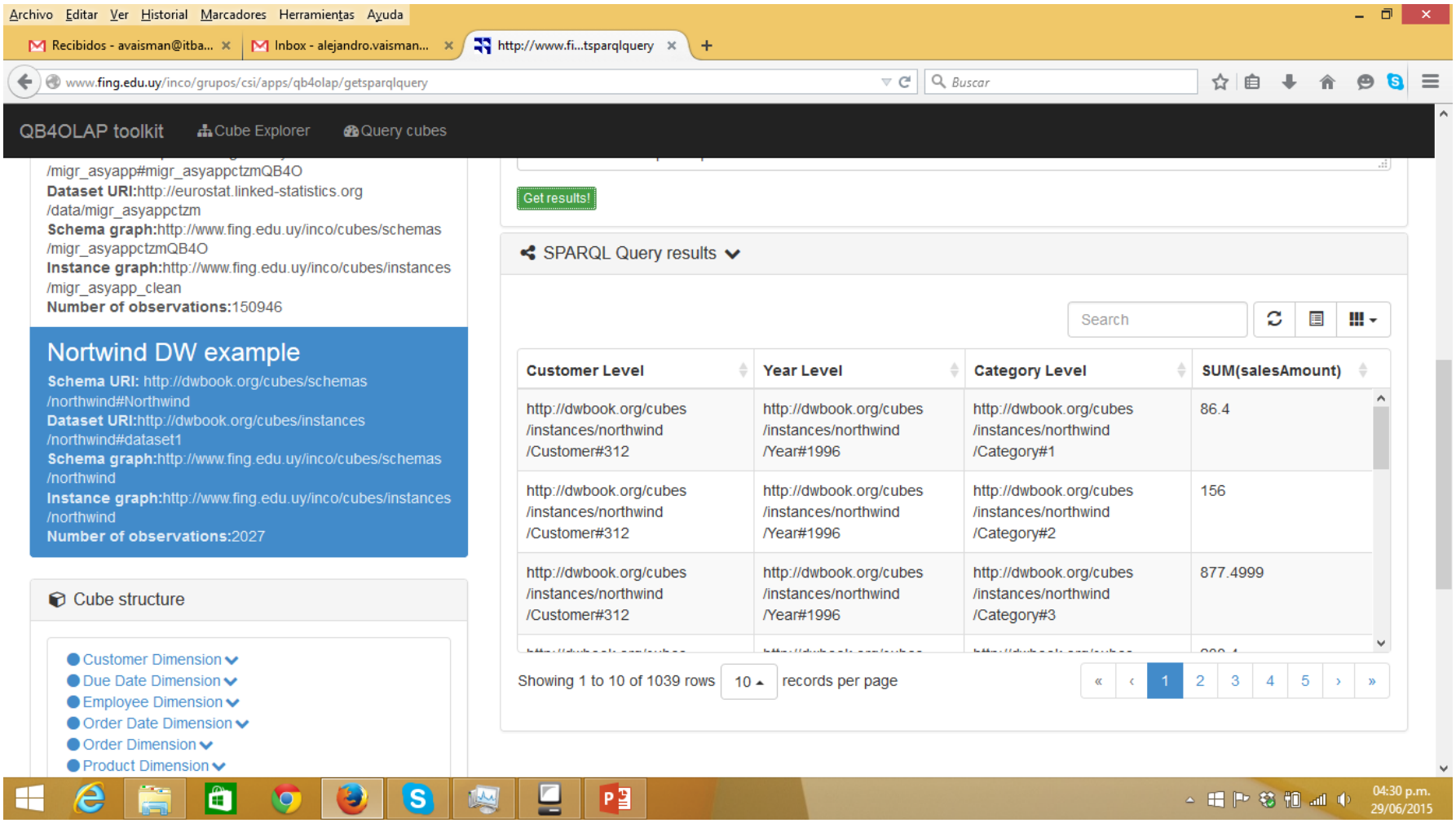


Cube Algebra to SPARQL translation

Query 4 Total sales amount per customer, year, and product category

```
SELECT ?lm1 ?plm4 ?plm5 (SUM(<http://www.w3.org/2001/XMLSchema#decimal>(?m1)) as ?ag1)
FROM <http://www.fing.edu.uy/inco/cubes/instances/northwind>
FROM <http://www.fing.edu.uy/inco/cubes/schemas/northwind>
 WHERE { ?o a qb:Observation .
?o qb:dataSet <http://dwbook.org/cubes/instances/northwind#dataset1> .
?o <http://dwbook.org/cubes/schemas/northwind#salesAmount> ?m1 .
?o <http://dwbook.org/cubes/schemas/northwind#customer> ?lm1 .
?lm1 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#customer> .
?o <http://dwbook.org/cubes/schemas/northwind#orderDate> ?lm2 .
?lm2 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#orderDate> .
?lm2 skos:broader ?plm1 . ?plm1 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#month> .
?plm1 skos:broader ?plm2 . ?plm2 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#quarter> .
?plm2 skos:broader ?plm3 .
?plm3 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#semester> .
?plm3 skos:broader ?plm4 . ?plm4 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#year> .
?o <http://dwbook.org/cubes/schemas/northwind#product> ?lm3 .
?lm3 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#product> . ?lm3 skos:broader ?plm5 .
?plm5 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#category> . }
GROUP BY ?lm1 ?plm4 ?plm5
```

Cube Algebra to SPARQL translation



Cube Algebra to SPARQL translation

- **Query 5** **Yearly** sales amount for each pair of **customer** **country**, supplier country

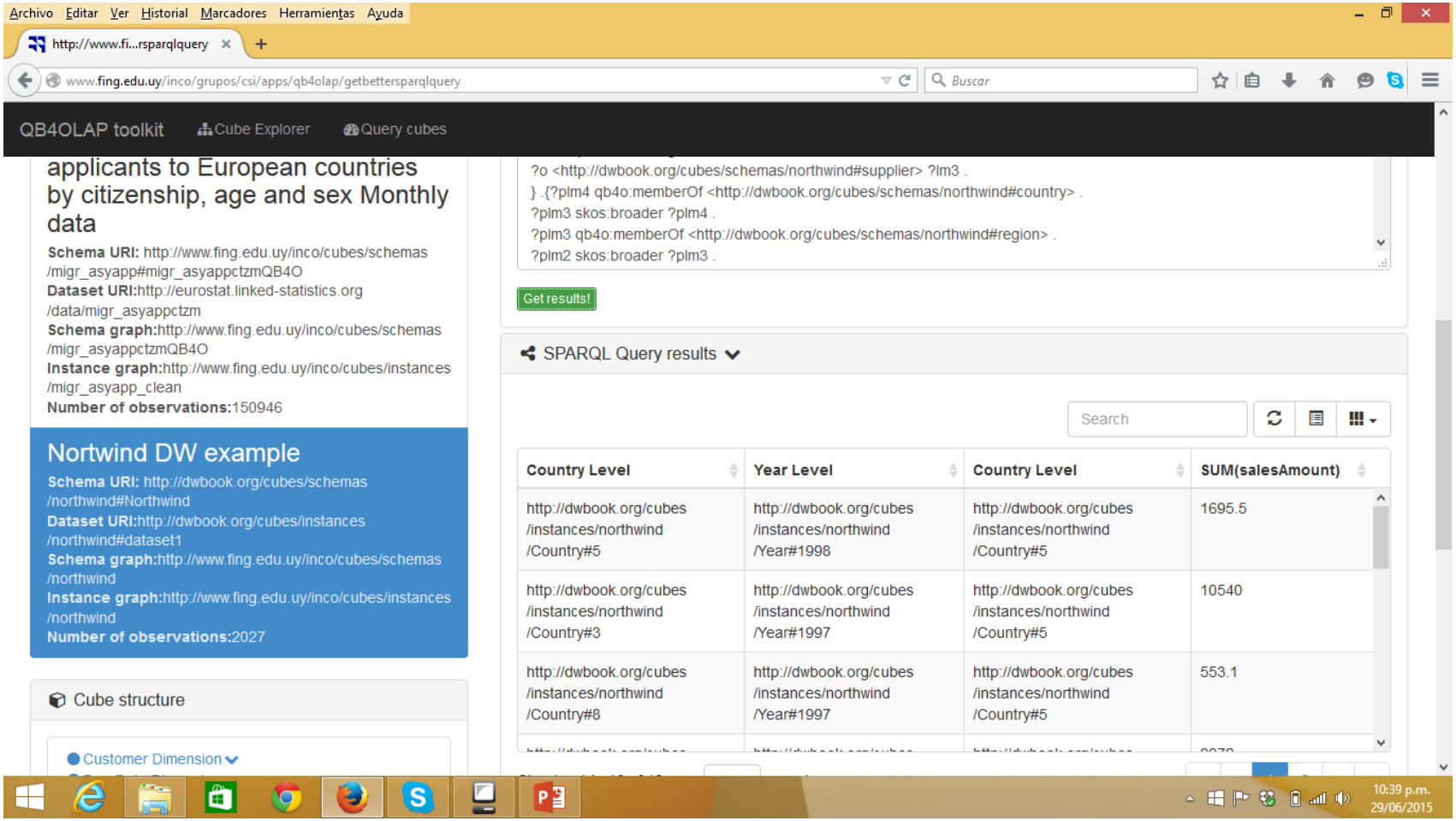
```
SELECT ?plm4 ?plm8 ?plm12  (SUM(<http://www.w3.org/2001/XMLSchema#decimal>(?m1)) as ?ag1)
FROM <http://www.fing.edu.uy/inco/cubes/instances/northwind>
FROM <http://www.fing.edu.uy/inco/cubes/schemas/northwind>
WHERE { ?o a qb:Observation .
?o qb:dataSet <http://dwbook.org/cubes/instances/northwind#dataset1> .
?o <http://dwbook.org/cubes/schemas/northwind#salesAmount> ?m1 .
?o <http://dwbook.org/cubes/schemas/northwind#customer> ?lm1 .
?lm1 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#customer> .
?lm1 skos:broader ?plm1 . ?plm1 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#city> .
?plm1 skos:broader ?plm2 . ?plm2 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#state> .
?plm2 skos:broader ?plm3 . ?plm3 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#region> .
?plm3 skos:broader ?plm4 .?plm4 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#country> .
?o <http://dwbook.org/cubes/schemas/northwind#orderDate> ?lm2 .
?lm2 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#orderDate> .
?lm2 skos:broader ?plm5 . ?plm5 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#month> .
?plm5 skos:broader ?plm6 .?plm6 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#quarter> .
?plm6 skos:broader ?plm7 . ?plm7 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#semester> .
?plm7 skos:broader ?plm8 . ?plm8 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#year> .
```

Cube Algebra to SPARQL translation

- **Query 5** Yearly sales amount for each pair of customer country, **supplier country**

```
SELECT ?plm4 ?plm8 ?plm12  (SUM(<http://www.w3.org/2001/XMLSchema#decimal>(?m1)) as ?ag1)
FROM <http://www.fing.edu.uy/inco/cubes/instances/northwind>
FROM <http://www.fing.edu.uy/inco/cubes/schemas/northwind>
WHERE { ?o a qb:Observation .
…….
?o <http://dwbook.org/cubes/schemas/northwind#supplier> ?lm3 .?lm3 qb4o:memberOf
<http://dwbook.org/cubes/schemas/northwind#supplier> .
?lm3 skos:broader ?plm9 .
?plm9 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#city> .
?plm9 skos:broader ?plm10 .
?plm10 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#state> .
?plm10 skos:broader ?plm11 .
?plm11 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#region> .
?plm11 skos:broader ?plm12 .
?plm12 qb4o:memberOf <http://dwbook.org/cubes/schemas/northwind#country> . }
GROUP BY ?plm4 ?plm8 ?plm12
ORDER BY ?plm4 ?plm8 ?plm12
```

Cube Algebra to SPARQL translation

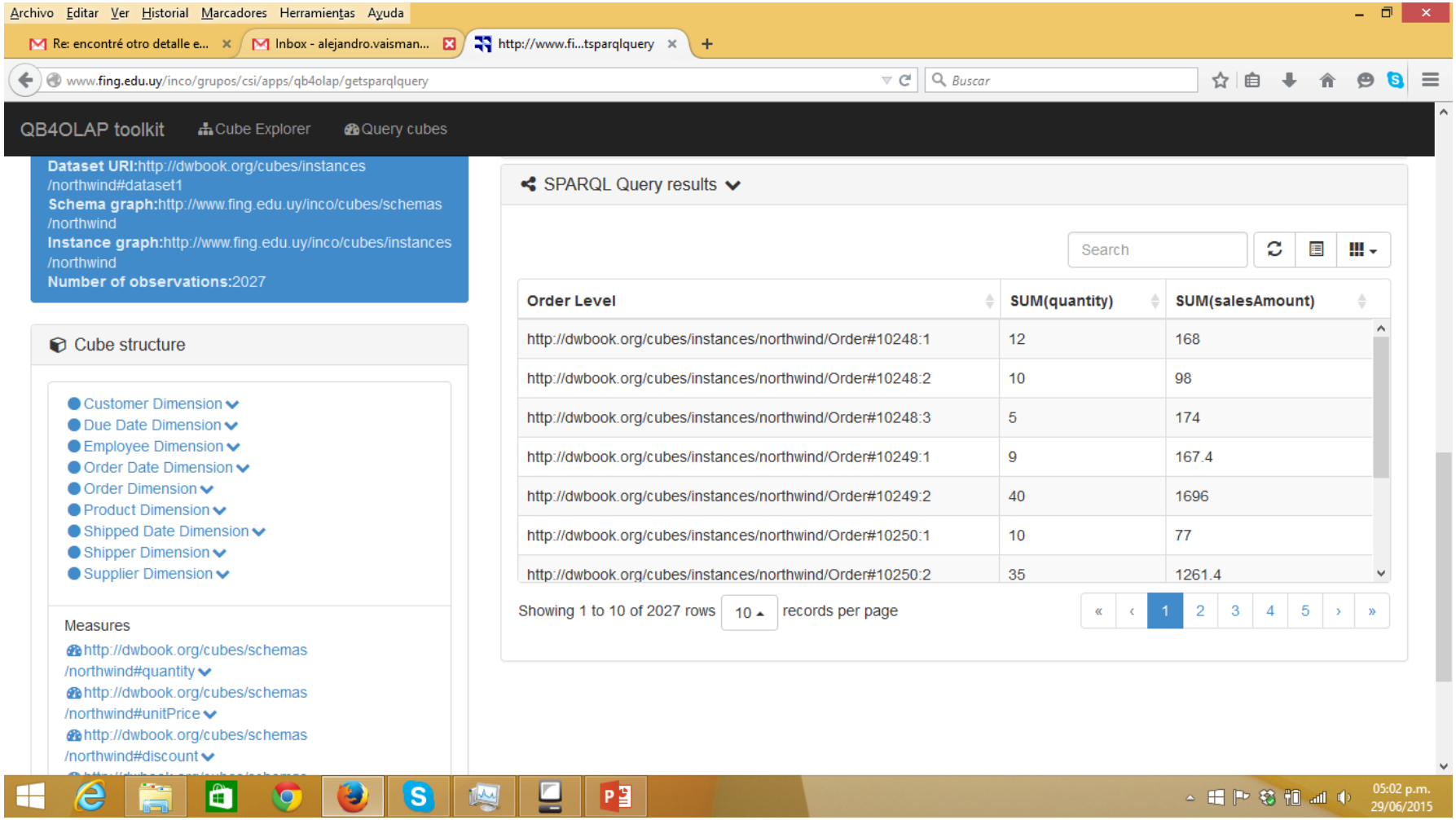


Cube Algebra to SPARQL translation

Query 6 Total sales amount, and sum of the quantities sold for each order

```
PREFIX qb: <http://purl.org/linked-data/cube#>
PREFIX qb4o: <http://purl.org/qb4olap/cubes#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
SELECT ?lm1  (SUM(<http://www.w3.org/2001/XMLSchema#integer>(?m1)) as ?ag1)
(SUM(<http://www.w3.org/2001/XMLSchema#decimal>(?m2)) as ?ag2)
FROM <http://www.fing.edu.uy/inco/cubes/instances/northwind>
FROM <http://www.fing.edu.uy/inco/cubes/schemas/northwind>
 WHERE { ?o a qb:Observation .
?o qb:dataSet <http://dwbook.org/cubes/instances/northwind#dataset1> .
?o <http://dwbook.org/cubes/schemas/northwind#quantity> ?m1 .
?o <http://dwbook.org/cubes/schemas/northwind#salesAmount> ?m2 .
?o <http://dwbook.org/cubes/schemas/northwind#order> ?lm1 .
 }
GROUP BY ?lm1
ORDER BY ?lm1
```

Cube Algebra to SPARQL translation

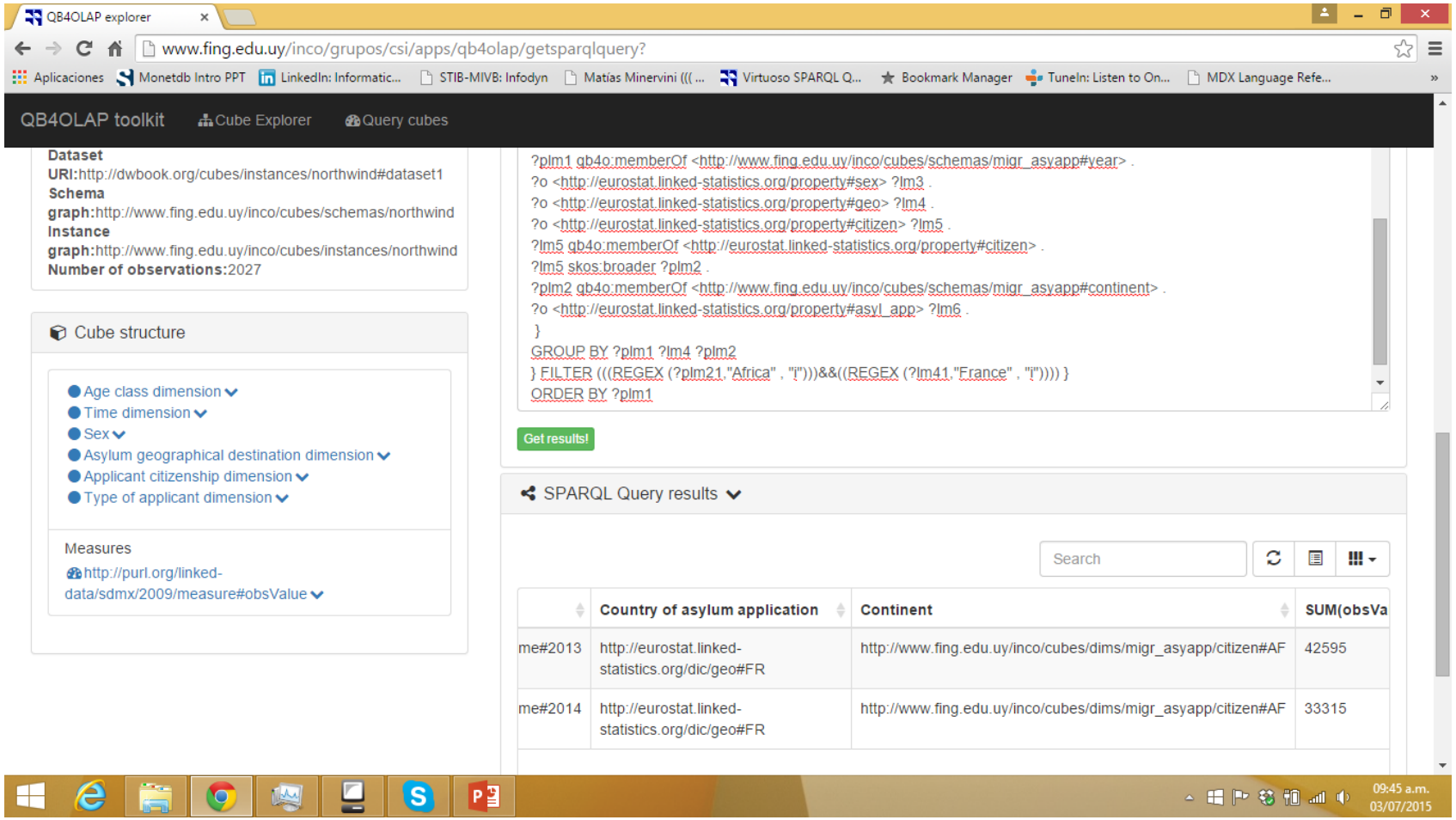


Cube Algebra to SPARQL translation

Query 7 Applications submitted **by African citizens, with destination France**

```
SELECT ?ag1 ?lm1 ?plm1 ?lm3 ?lm4 ?plm2
 WHERE {{ ?plm2 <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#continentName> ?plm21 .
 ?lm4 <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#countryName> ?lm41 .} .
{SELECT ?lm1 ?plm1 ?lm3 ?lm4 ?plm2  (SUM(<http://www.w3.org/2001/XMLSchema#integer>(?m1)) as ?ag1)
FROM <http://www.fing.edu.uy/inco/cubes/instances/migr_asyapp_clean>
FROM <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyappctzmQB4O>
WHERE { ?o a qb:Observation . ?o qb:dataSet <http://eurostat.linked-statistics.org/data/migr_asyappctzm> .
?o <http://purl.org/linked-data/sdmx/2009/measure#obsValue> ?m1 .
?o <http://eurostat.linked-statistics.org/property#age> ?lm1 .
?o <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> ?lm2 .
?lm2 qb4o:memberOf <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> . ?lm2 skos:broader ?plm1 .
?plm1 qb4o:memberOf <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#year> .
?o <http://eurostat.linked-statistics.org/property#sex> ?lm3 . ?o <http://eurostat.linked-statistics.org/property#geo> ?lm4 .
?o <http://eurostat.linked-statistics.org/property#citizen> ?lm5 .
?lm5 qb4o:memberOf <http://eurostat.linked-statistics.org/property#citizen> . ?lm5 skos:broader ?plm2 .
?plm2 qb4o:memberOf <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#continent> .
?o <http://eurostat.linked-statistics.org/property#asyl_app> ?lm6 .}
GROUP BY ?lm1 ?plm1 ?lm3 ?lm4 ?plm2 }
FILTER ((REGEX (?plm21,"Africa" , "i")) && (REGEX (?lm41,"France" , "i"))) }
```

Cube Algebra to SPARQL translation



Cube Algebra to SPARQL translation

Query 8 Asylum applications by year submitted by Asian citizens, where applications count > 5000, and destination France or Germany

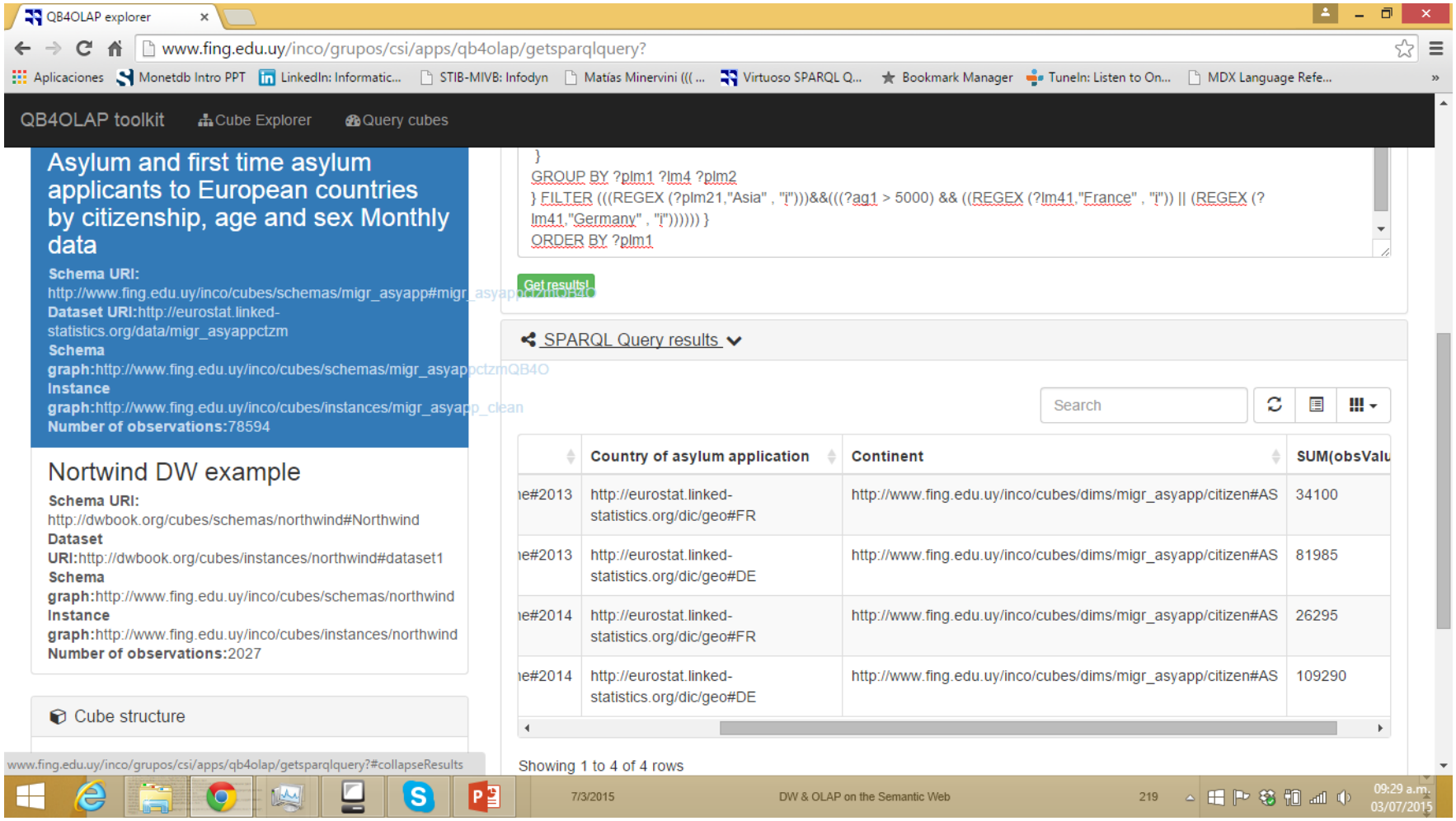
```
PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;
PREFIX data: <http://eurostat.linked-statistics.org/data/>;
PREFIX property: <http://eurostat.linked-statistics.org/property#>;
PREFIX sdmx-measure:<http://purl.org/linked-data/sdmx/2009/measure#> ;
QUERY
$C1 := ROLLUP (data:migr_asyappctzm, schema:citizenshipDim,schema:citAll);
$C2 := ROLLUP ($C1, schema:timeDim, schema:year);
$C3 := DRILLDOWN ($C2, schema:citizenshipDim,schema:continent);
$C4 := SLICE ($C3, schema:asylappDim);
$C5 := SLICE ($C4, schema:ageDim);
$C6 := SLICE ($C5, schema:sex);
$C7 := DICE ($C6, (schema:citizenshipDim|schema:continent|schema:continentName = "Asia"));
$C8 := DICE ($C7, ( sdmx-measure:obsValue > 5000 AND
(schema:destinationDim|property:geo|schema:countryName = "France") OR
(schema:destinationDim|property:geo|schema:countryName = "Germany")));
```

Cube Algebra to SPARQL translation

Query 8 Asylum applications by year submitted by Asian citizens, where applications count > 5000 with destination France or Germany

```
SELECT ?ag1 ?plm1 ?lm4 ?plm2
WHERE {{ ?plm2 <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#continentName> ?plm21 .
?lm4 <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#countryName> ?lm41 .}.
{ SELECT ?plm1 ?lm4 ?plm2  (SUM(<http://www.w3.org/2001/XMLSchema#integer>(?m1)) as ?ag1)
FROM <http://www.fing.edu.uy/inco/cubes/instances/migr_asyapp_clean>
FROM <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyappctzmQB4O>
WHERE { ?o a qb:Observation . ?o qb:dataSet <http://eurostat.linked-statistics.org/data/migr_asyappctzm> .
?o <http://purl.org/linked-data/sdmx/2009/measure#obsValue> ?m1 .
?o <http://eurostat.linked-statistics.org/property#age> ?lm1 .
?o <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> ?lm2 .
?lm2 qb4o:memberOf <http://purl.org/linked-data/sdmx/2009/dimension#refPeriod> . ?lm2 skos:broader ?plm1 .
?plm1 qb4o:memberOf <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#year> .
?o <http://eurostat.linked-statistics.org/property#sex> ?lm3 . ?o <http://eurostat.linked-statistics.org/property#geo> ?lm4 .
?o <http://eurostat.linked-statistics.org/property#citizen> ?lm5 .
?lm5 qb4o:memberOf <http://eurostat.linked-statistics.org/property#citizen> .
?lm5 skos:broader ?plm2 .?plm2 qb4o:memberOf <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#continent> .
?o <http://eurostat.linked-statistics.org/property#asyl_app> ?lm6 . }
GROUP BY ?plm1 ?lm4 ?plm2
} FILTER (((REGEX (?plm21,"Asia" , "i")))&&(((?ag1 > 5000) && ((REGEX (?lm41,"France" , "i")) || (REGEX (?lm41,"Germany" ,
"i")))))) }
```

Cube Algebra to SPARQL translation



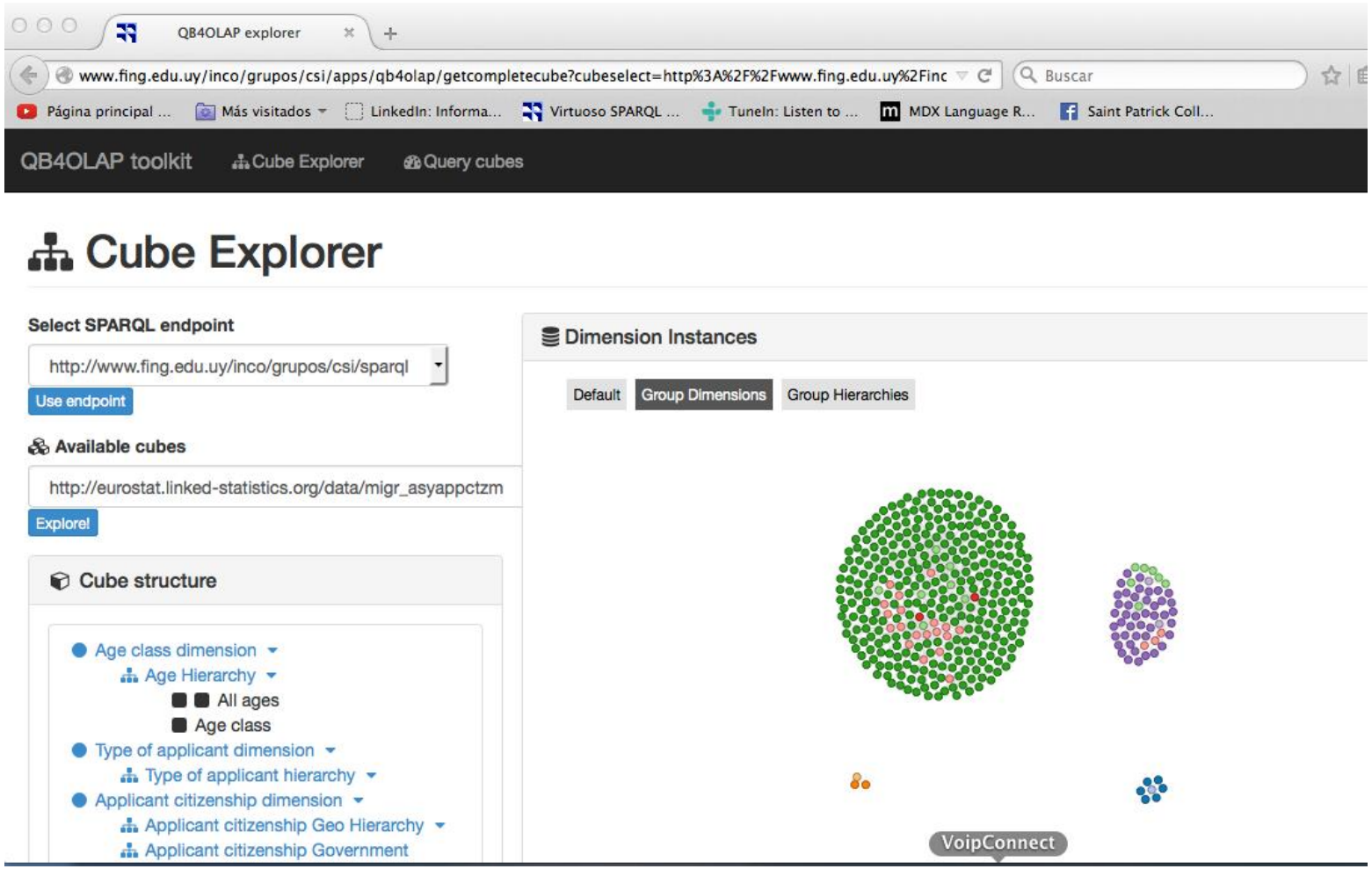
Exploring the cube dimensions

Implementation: Allows describing and querying QB4OLAP cubes

The **cube explorer** can be accessed at:

http://www.fing.edu.uy/inco/grupos/csi/apps/qb4olap/explorer

Exploring the cube dimensions

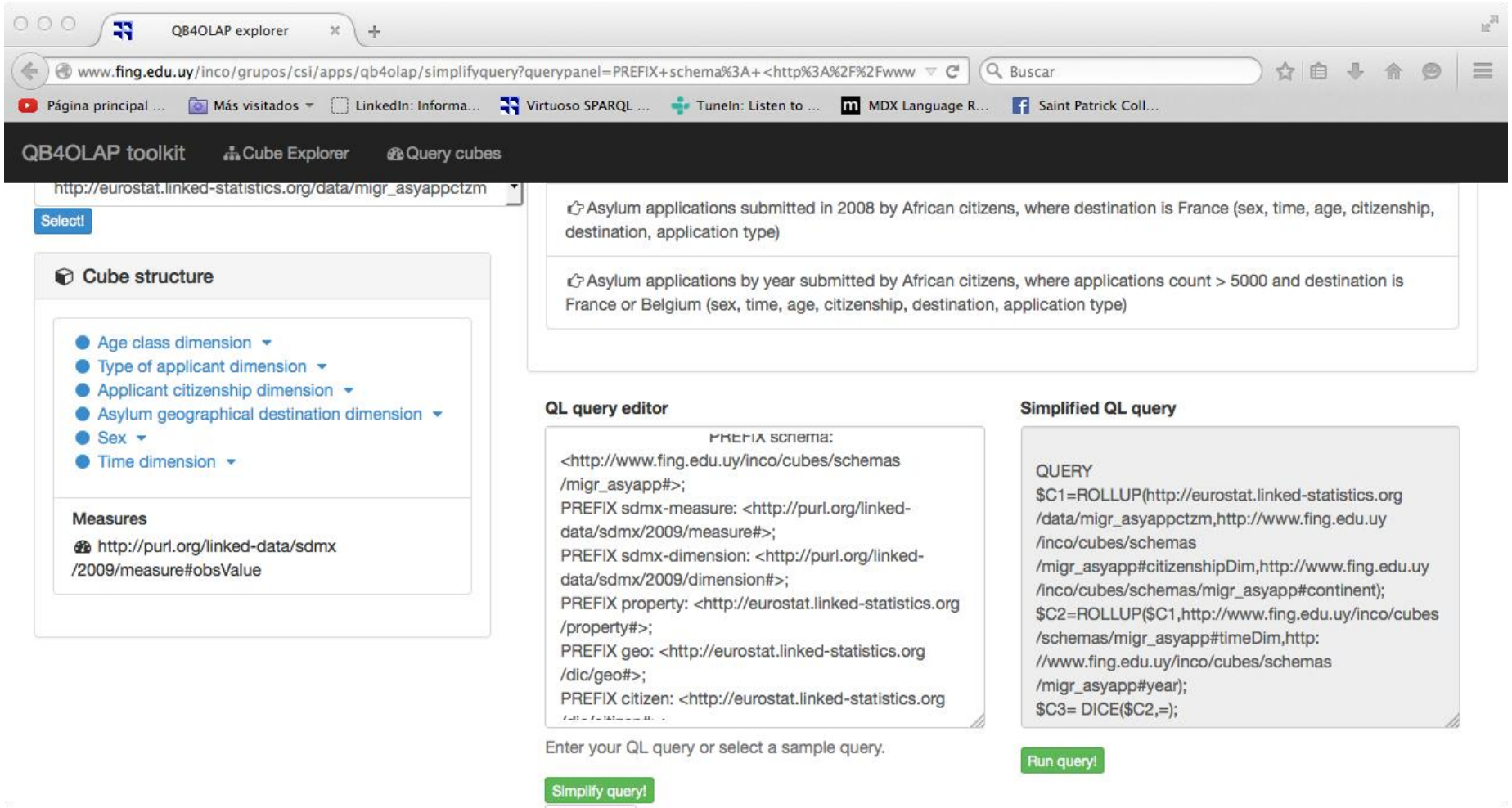


Querying QB4OLAP data cubes with Cube Algebra

The **query tool** can be accessed at:

http://www.fing.edu.uy/inco/grupos/csi/apps/qb4olap/queries

Querying QB4OLAP data cubes with Cube Algebra



Outline

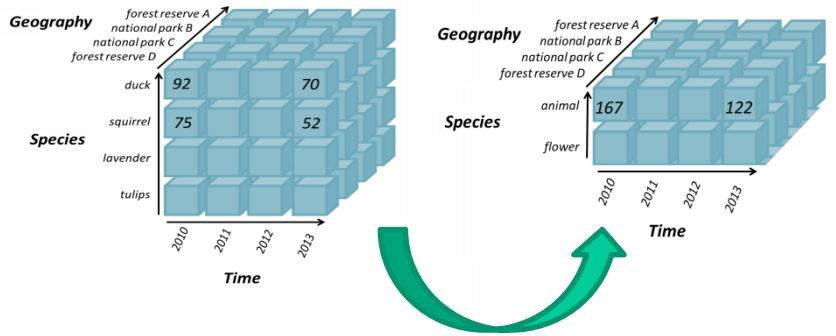
- The Semantic Web
- RDF and SPARQL Basics
- Vocabularies for OLAP on the SW: QB and QB4OLAP
- Modeling Data Cubes on the Semantic Web using QB4OLAP
- Querying Data Cubes on the Semantic Web
- **Summary**

Summary

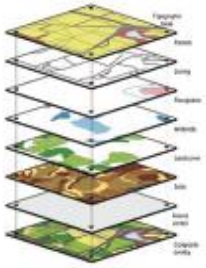
- Publishing MD on the SW: Allows sharing data cubes (DC), combine then with traditional DCs, build cubes on-the-fly over web data, etc.
- QB not enough for OLAP analysis, QB4OLAP proposed
  - Check `http://ec.europa.eu/eurostat` and compare asylum application data
- QB4OLAP compatible with QB, no need to copy observations
- Cube Algebra (CA) allows writing OLAP queries without knowledge of RDF/SPARQL
- Simplification of queries in CA
- Translation of CA into SPARQL, and SPARQL “optimization” (in progress)

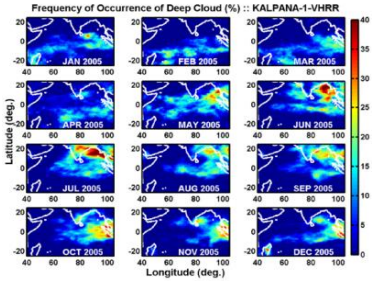
A generic MD view of data

Traditionally: **Warehouse + OLAP cubes**



Currently: **New kinds of data**

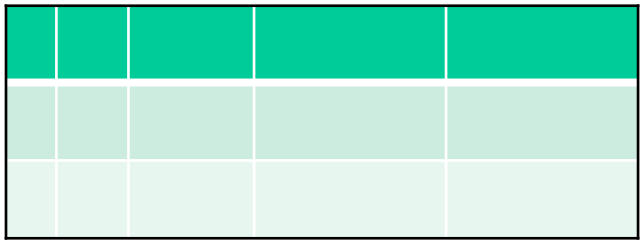


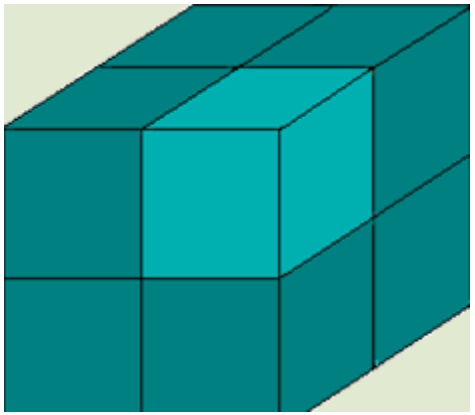


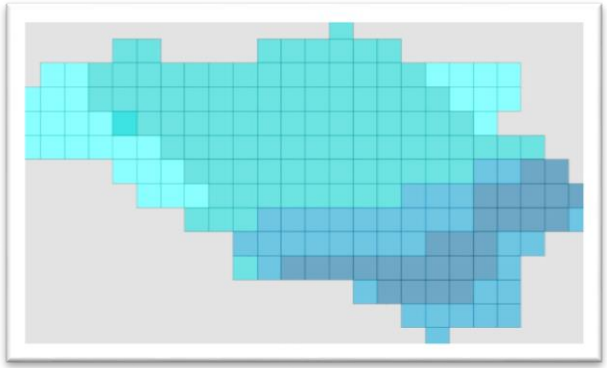
* Web logs
* Spatial
* Semantic web (RDF)
* etc…

A generic MD view of data

- Manipulation of different kinds of data
- Several query languages
- Ad-hoc information integration

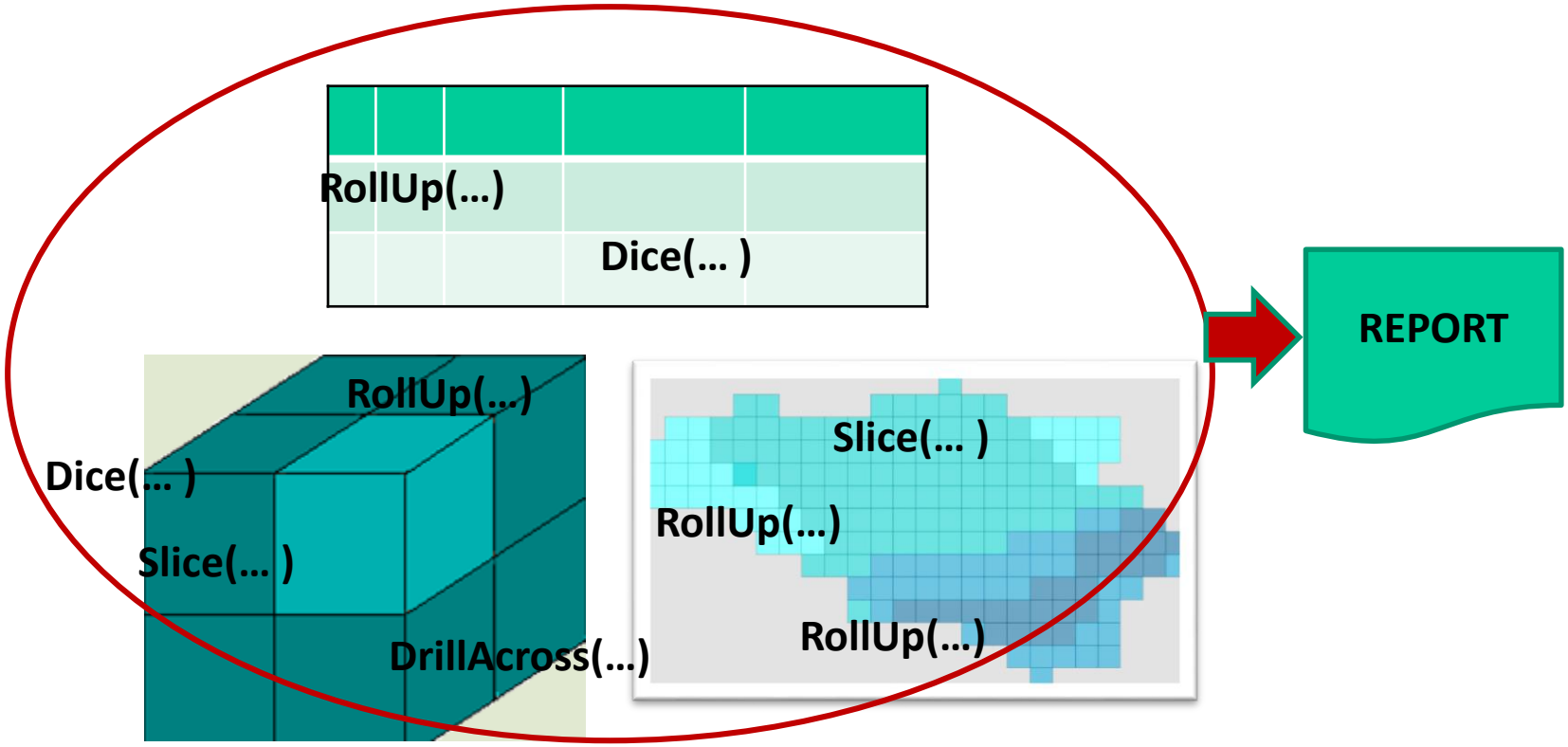




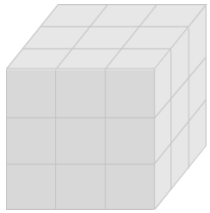


A generic MD view of data

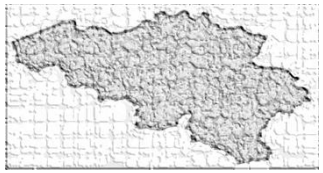
We just want to manipulate OLAP cubes



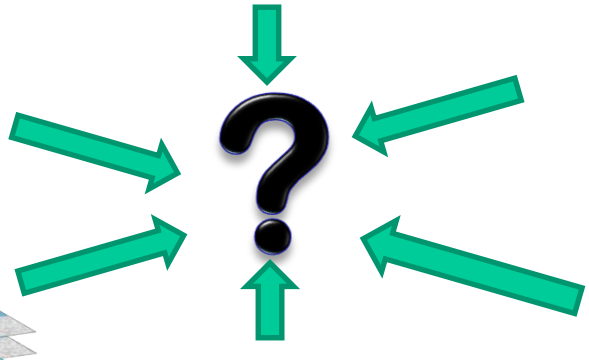
A generic MD view of data

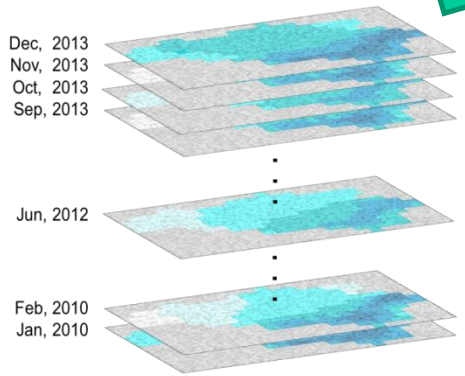


OLAP Cubes

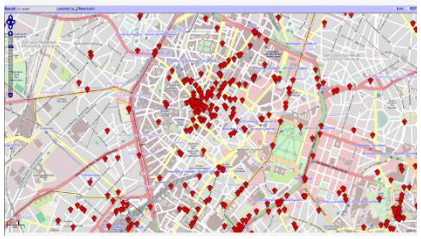


Spatial continuous





Spatiotemporal continuous

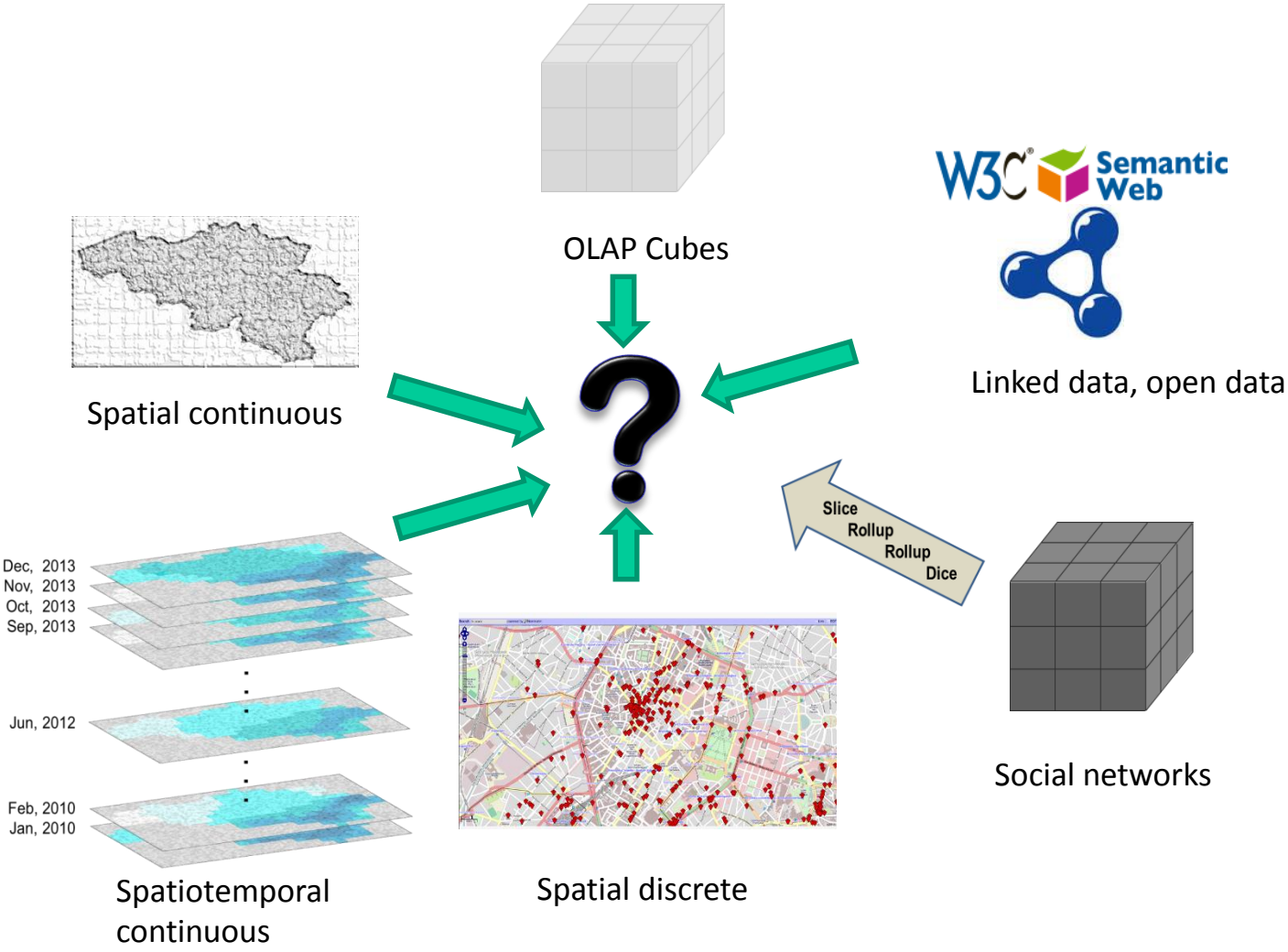


Spatial discrete

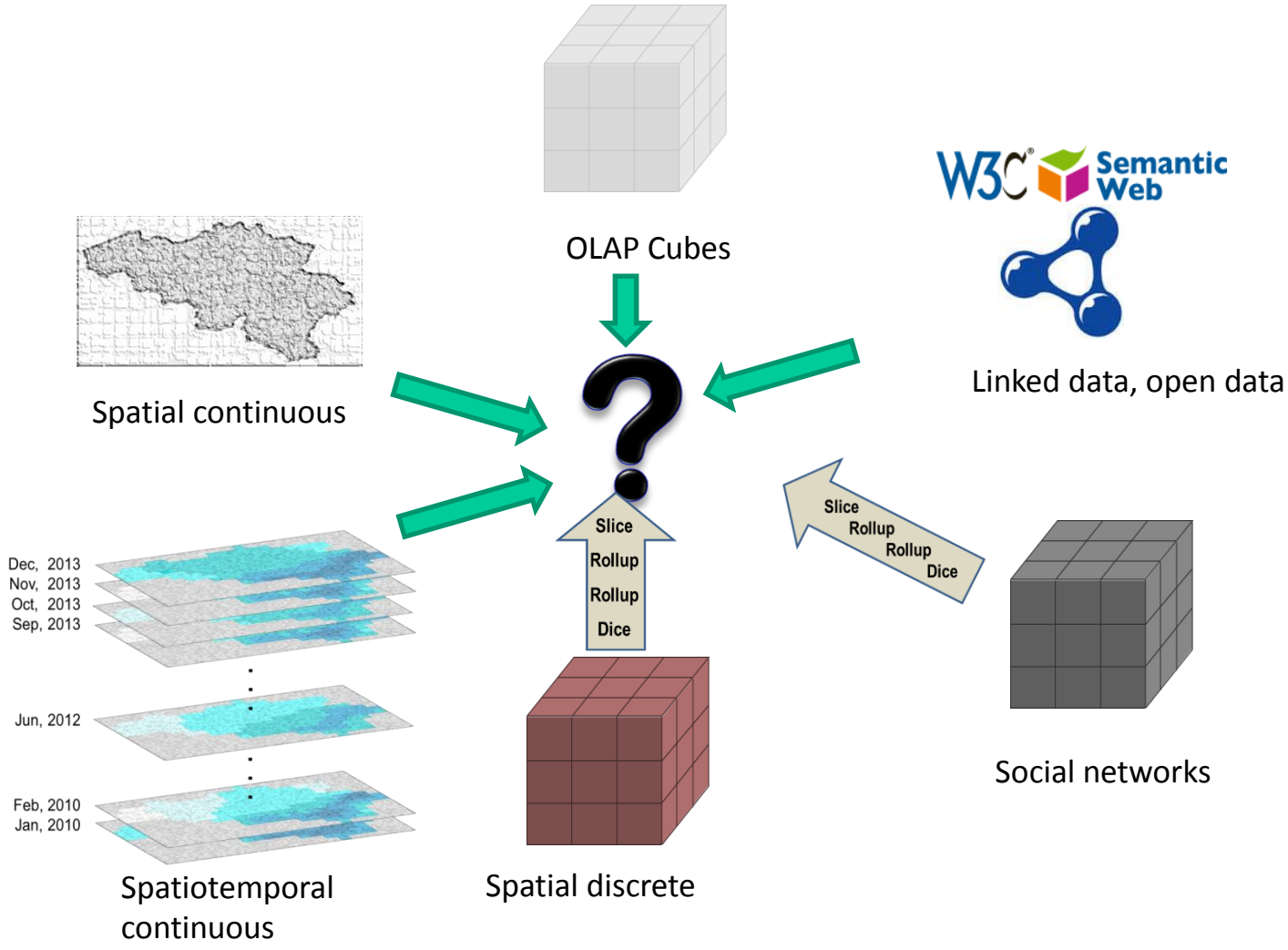
Linked data, open data

Social networks

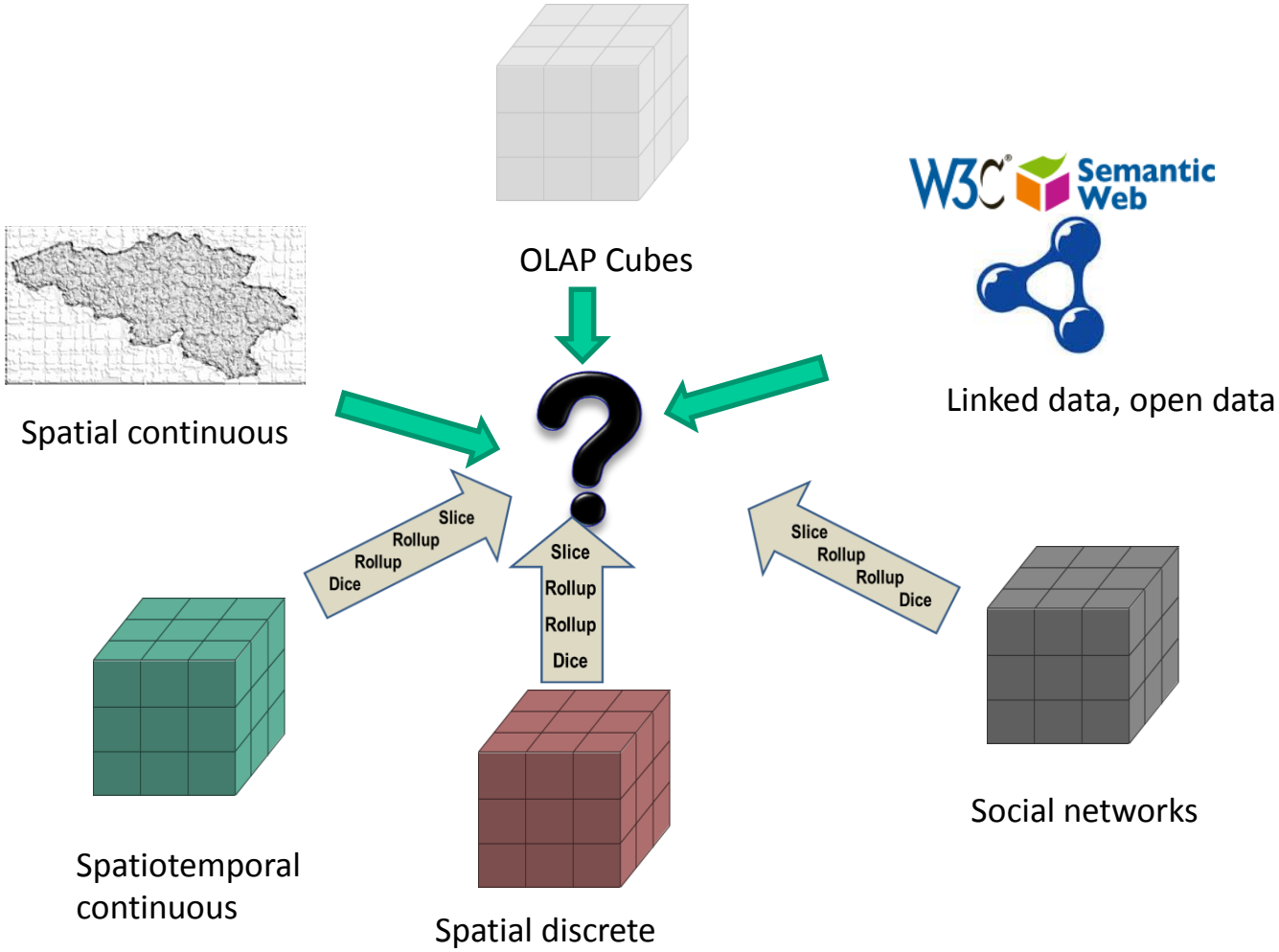
A generic MD view of data



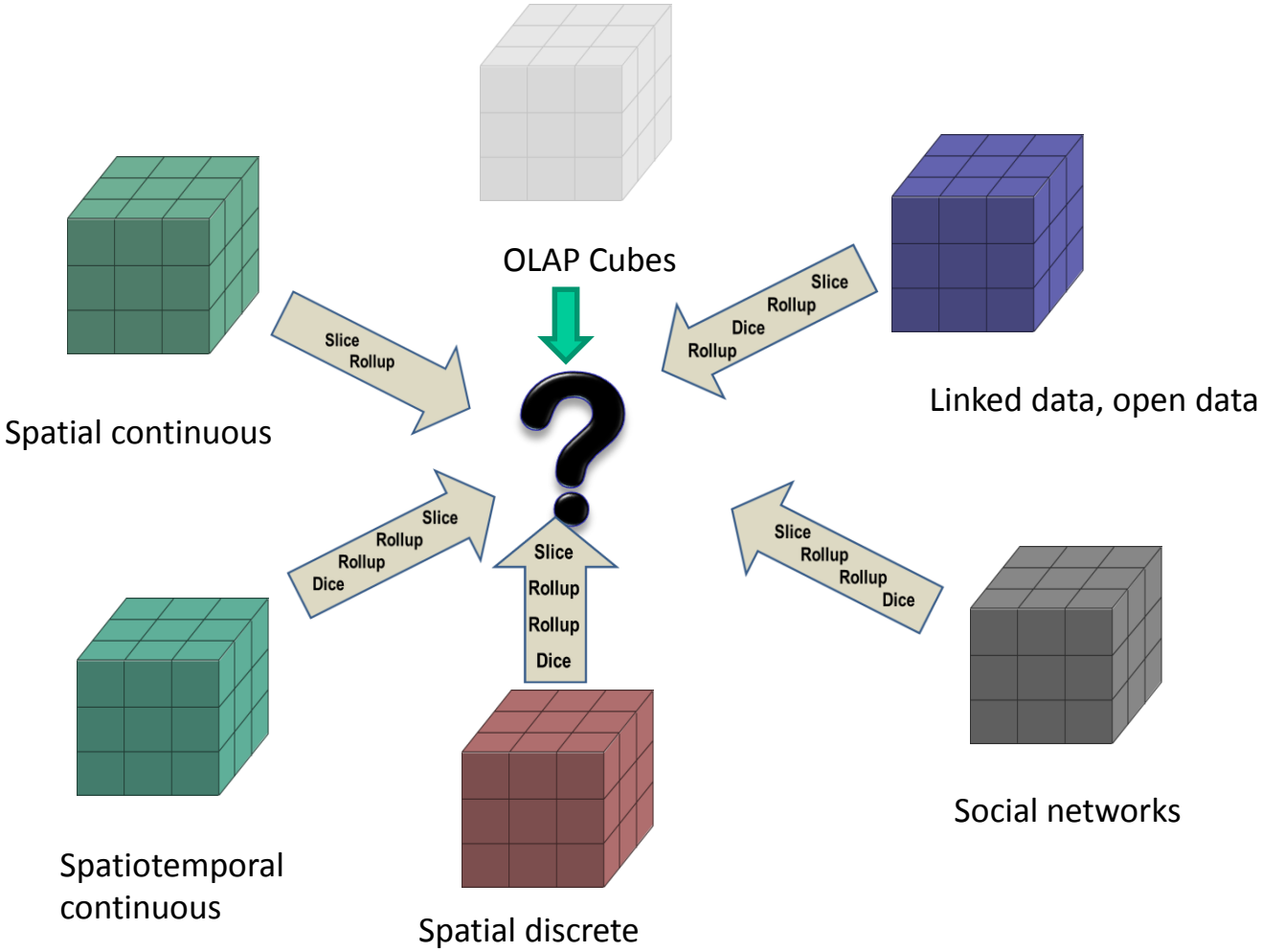
A generic MD view of data



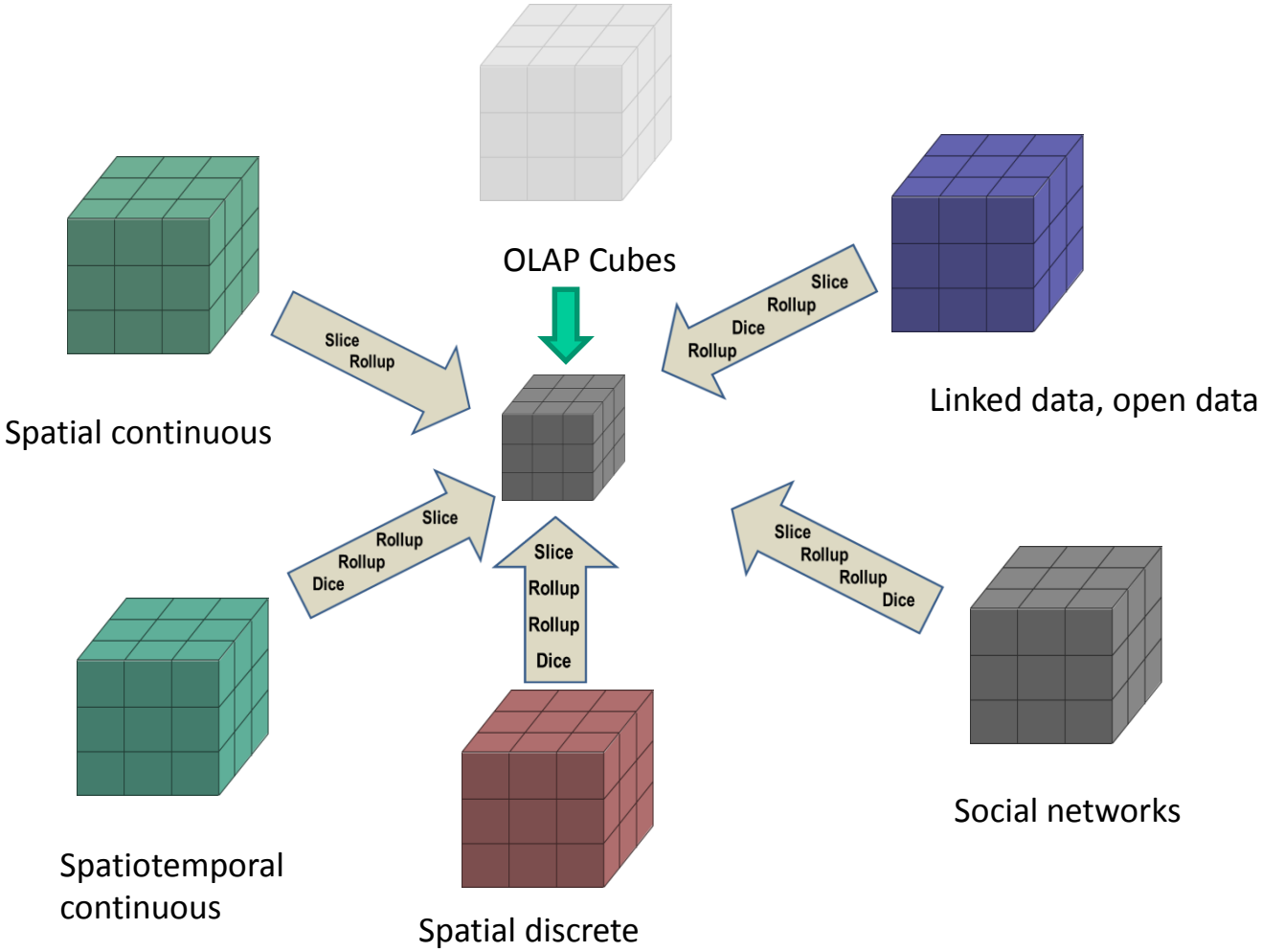
A generic MD view of data



A generic MD view of data

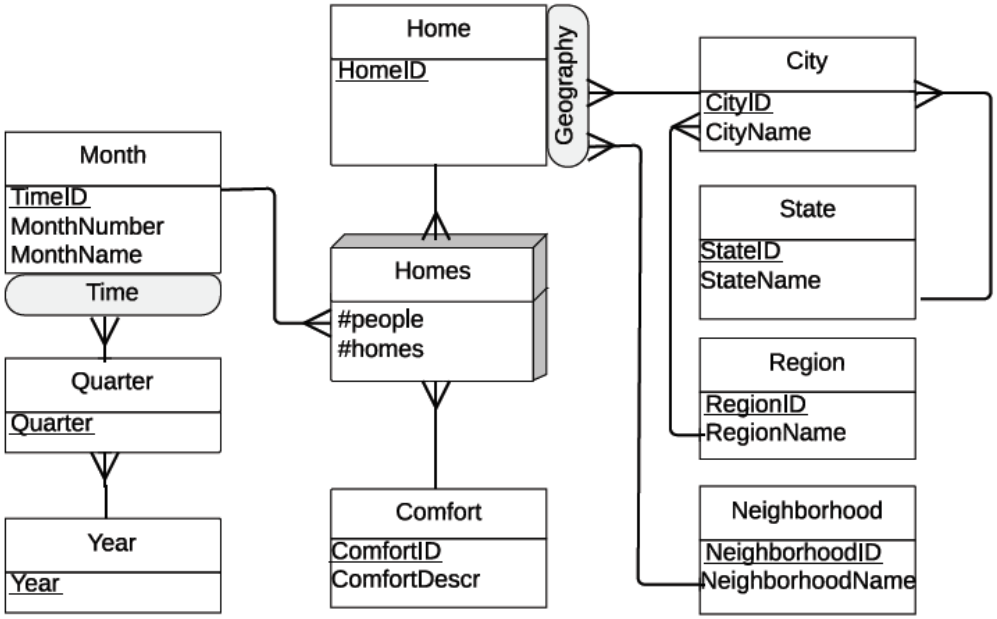


A generic MD view of data



Exercises

- Represent, in QB4OLAP, the DW representing housing situation across time in Uruguay.



Exercises

- Go to http://ec.europa.eu/eurostat/data/database , look in “tables by themes” -> “population and social conditions” -> Asylum and managed migration”.
- Go to the “table” interface, e.g., http://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=migr_asyappctzm&lang=en
- Write, in Cube Algebra, the queries that produce the information in the Eurostat tables, compare the results.

Exercises

- Example: monthly asylum applications by country

PREFIX schema: <http://www.fing.edu.uy/inco/cubes/schemas/migr_asyapp#>;

PREFIX sdmx: <http://purl.org/linked-data/sdmx/2009/dimension#>;

QUERY

$C1 := SLICE (<http://eurostat.linked-statistics.org/data/migr_asyappctzm>,

schema:citizenshipDim);

$C2 := SLICE($C1, schema:ageDim);

$C3 := SLICE($C2, schema:sex);

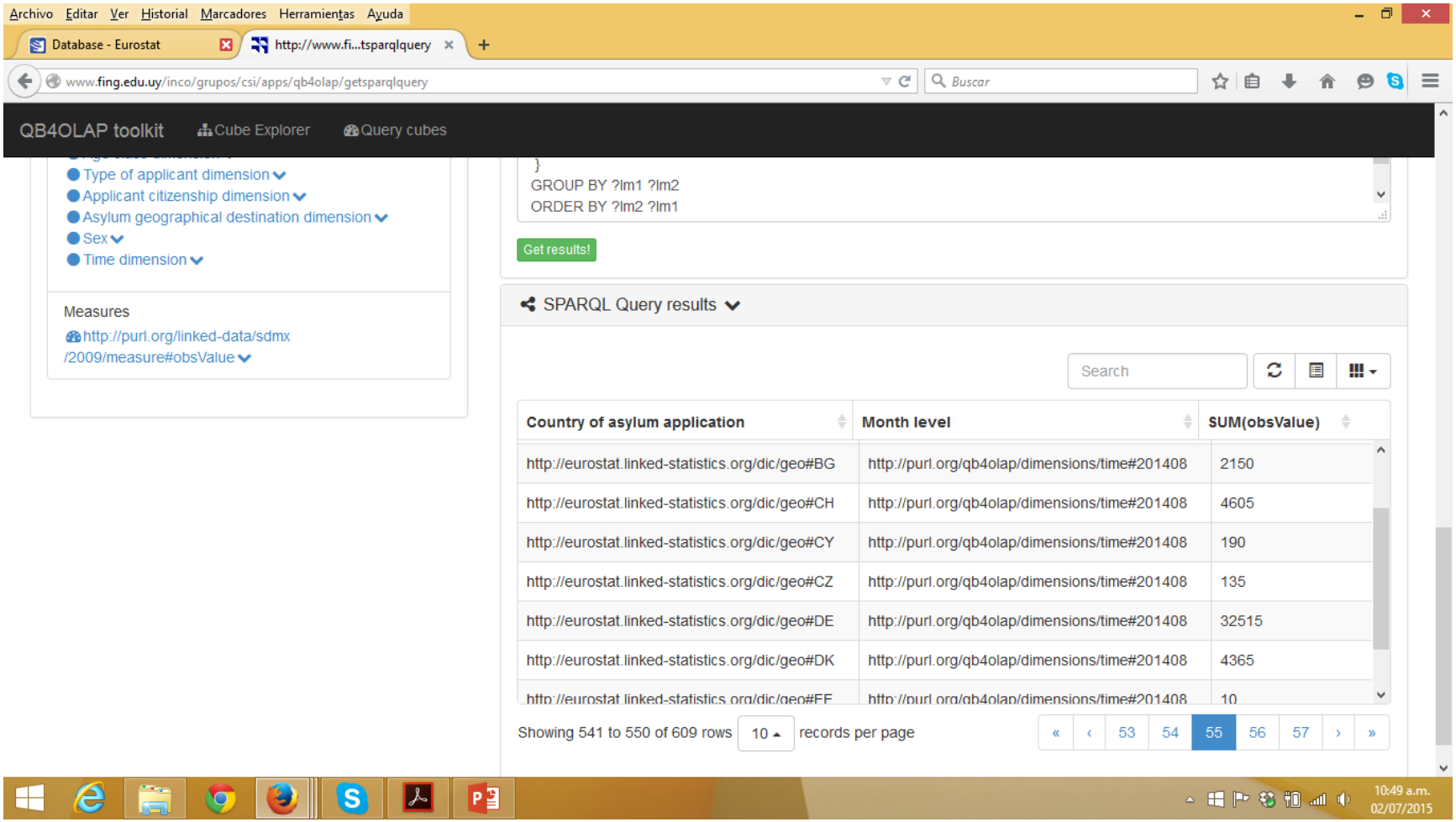
$C4 := SLICE($C3, schema:asylappDim);

$C5 := ROLLUP ($C4, schema:timeDim, sdmx:refPeriod);

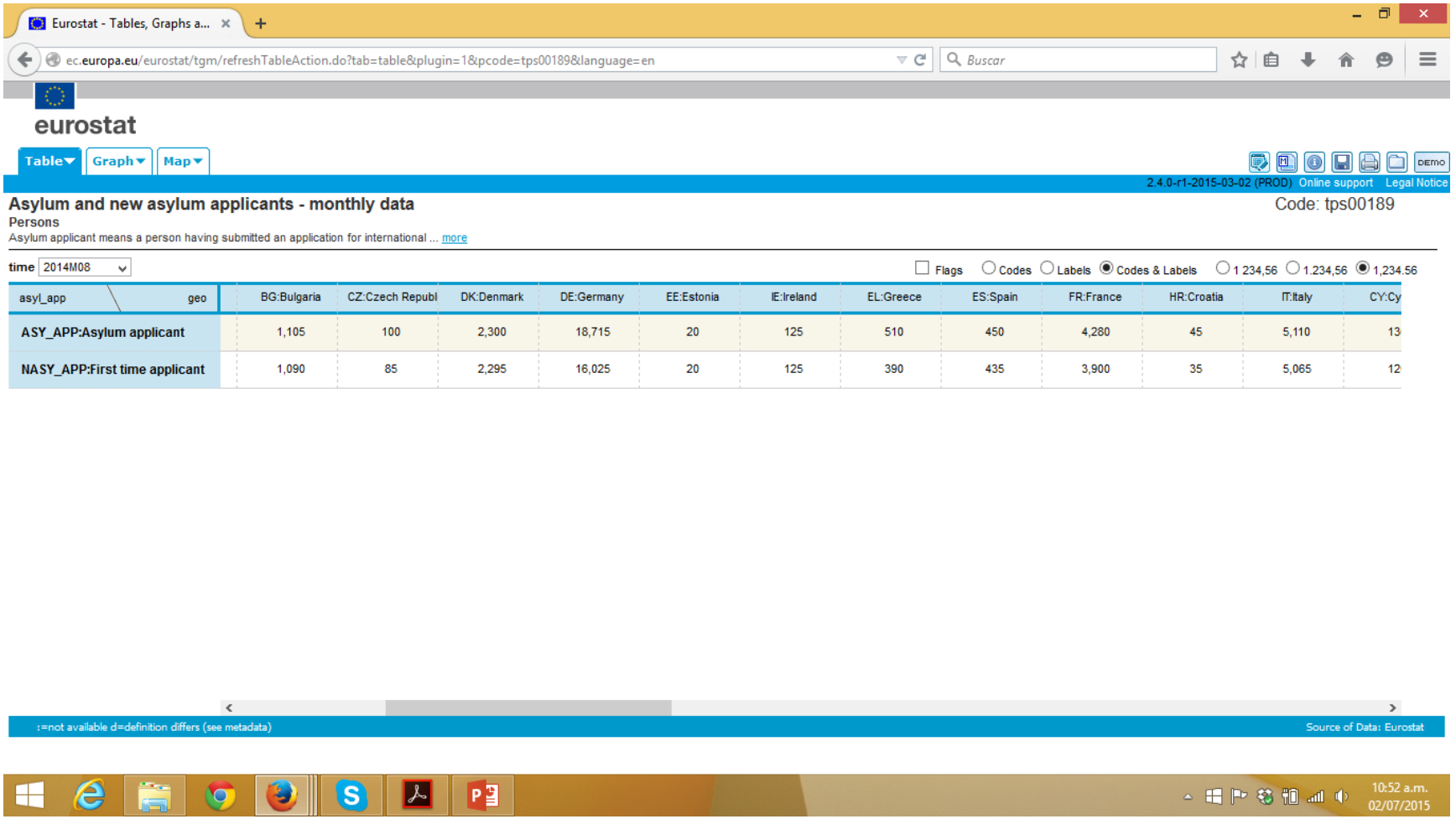
$C6 := ROLLUP($C5, schema:destinationDim,

<http://eurostat.linked-statistics.org/property#geo>);

Exercises


Archivo Editar Ver Historial Marcadores Herramientas Ayuda
Database - Eurostat
http://www.fi...tsparqlquery
www.fing.edu.uy/inco/grupos/csi/apps/qb4olap/getsparqlquery
Buscar
QB4OLAP toolkit
Cube Explorer
Query cubes
Type of applicant dimension
Applicant citizenship dimension
Asylum geographical destination dimension
Sex
Time dimension
Measures
http://purl.org/linked-data/sdmx/2009/measure#obsValue
}
GROUP BY ?lm1 ?lm2
ORDER BY ?lm2 ?lm1
Get results!
SPARQL Query results
Search
Country of asylum application
Month level
SUM(obsValue)
http://eurostat.linked-statistics.org/dic/geo#BG
http://purl.org/qb4olap/dimensions/time#201408
2150
http://eurostat.linked-statistics.org/dic/geo#CH
http://purl.org/qb4olap/dimensions/time#201408
4605
http://eurostat.linked-statistics.org/dic/geo#CY
http://purl.org/qb4olap/dimensions/time#201408
190
http://eurostat.linked-statistics.org/dic/geo#CZ
http://purl.org/qb4olap/dimensions/time#201408
135
http://eurostat.linked-statistics.org/dic/geo#DE
http://purl.org/qb4olap/dimensions/time#201408
32515
http://eurostat.linked-statistics.org/dic/geo#DK
http://purl.org/qb4olap/dimensions/time#201408
4365
Showing 541 to 550 of 609 rows
10
records per page
53
54
55
56
57
10:49 a.m.
02/07/2015

Exercises


Eurostat - Tables, Graphs a...
ec.europa.eu/eurostat/tgm/refreshTableAction.do?tab=table&plugin=1&pcode=tps00189&language=en
Buscar
eurostat
Table
Graph
Map
DEMO
2.4.0-r1-2015-03-02 (PROD) Online support Legal Notice
Asylum and new asylum applicants - monthly data
Code: tps00189
Persons
Asylum applicant means a person having submitted an application for international ... more
time 2014M08
Flags
Codes
Labels
Codes & Labels
1 234,56
1.234,56
1,234.56
asyl_app
geo
BG:Bulgaria
CZ:Czech Republ
DK:Denmark
DE:Germany
EE:Estonia
IE:Ireland
EL:Greece
ES:Spain
FR:France
HR:Croatia
IT:Italy
CY:Cy
ASY_APP:Asylum applicant
1,105
100
2,300
18,715
20
125
510
450
4,280
45
5,110
13
NASY_APP:First time applicant
1,090
85
2,295
16,025
20
125
390
435
3,900
35
5,065
12
:=not available d=definition differs (see metadata)
Source of Data: Eurostat
10:52 a.m.
02/07/2015

Thanks!

Contact: avaisman@itba.edu.ar