Spatial Data Warehouses on the Semantic Web

- Multidimensional data warehouses (a.k.a. data cubes) on the (geospatial) Semantic Web
- Difficult to formulate SOLAP queries in SPARQL
- Spatial functions in a SOLAP query involve several triple patterns and nesting SOLAP in SPARQL increases complexity

Contributions

- Semi-automated generation of SOLAP operations
- Formulation of SPARQL queries using a graphical interface (GUI) displaying MD schema elements
- Intuitive query interface for non-experts
- Nested SOLAP in SPARQL: \((s\text{-}dice(s\text{-}roll-up(s\text{-}slice(C))))\)

QB4SOLAP Vocabulary

QB4SOLAP defines spatial extensions of:
- Facts
- Measures
- Attributes
- Levels
- Hierarchies
- Dimensions
- Hierarchy steps
- Level members
- Aggregate functions
- Topological relations

GeoSemOLAP Workflow

- Select a SOLAP operator
- Select MD elements and spatial operations
- Generate SPARQL Query
- Aggregate/Disaggregate
- Show the results
- Execute the Query
- Edit the Query (Optional)

GeoSemOLAP Architecture

Graphical User Interface (GUI)
Metadata Manager
Query Generator
Data Processor
SPARQL Endpoint

Spatial OLAP (SOLAP) Queries

**s-slice:** removes a dimension from a cube by choosing a single spatial attribute
**s-dice:** keeps the cells of the cube that satisfy a spatial predicate over dimension levels, attributes, and measures
**s-roll-up:** aggregates measures of a spatial data cube by using an aggregate function and a spatial function

Future Perspectives

- Map of sales data
- Total sales to customers by city of their closest supplier

SOLAP Queries in SPARQL: s-roll-up

```
1 SELECT ?obs ?supCity (SUM(?sales)) AS ?totalSales
2 WHERE {(?obs rdf:type ?qbo:Observation) .
3  gwn:customerID ?cust ;
4  gwn:supplierID ?sup ;
5  gwn:salesAmount ?sales .
6  ?cust qb4s:memberOf gwn:customer ;
7  gwn:customerGeo ?custGeo .
8  ?sup qb4s:memberOf ?gnw:supplier ;
9  gwn:supplierGeo ?supGeo ;
10  ?supCity qb4s:memberOf gwn:city .
#Inner select for the distance function
11  (SELECT ?cust1 (MIN(?distance)) AS ?minDistance)
12  WHERE {(?obs rdf:type ?qbo:Observation) .
13  gwn:customerID ?cust1 ;
14  gwn:supplierID ?sup1 .
15  ?sup1 gwn:supplierGeo ?sup1Geo .
17  BIND (bif:st_distance(?cust1Geo, ?sup1Geo) AS ?distance))
18  GROUP BY ?cust1 }
20  FILTER (?cust = ?cust1 && bif:st_distance
21  (?custGeo, ?supGeo) = ?minDistance))
23  GROUP BY ?supCity ?obs
```

References


Acknowledgements

This research is partially funded by the European Commission through the Erasmus Mundus Joint Doctorate IT4IS-DC and the Danish Council for Independent Research (DFF) under grant agreement no. DFF-4083-00301.