A geospatial extension of RDF and an implementation on top of an RDBMS

Kostis Kyzirakos

MaDgIK Lab
Dept. of Informatics & Telecommunications
University of Athens

Joint Work with …

- Prof. Manolis Koubarakis
- Manos Karpathiotakis

Outline

1. Introduction
2. The Data Model stRDF
3. The Query Language stSPARQL
4. The System Strabon
5. The Semantic Registry
6. Experimental evaluation
7. Future Work

Motivation

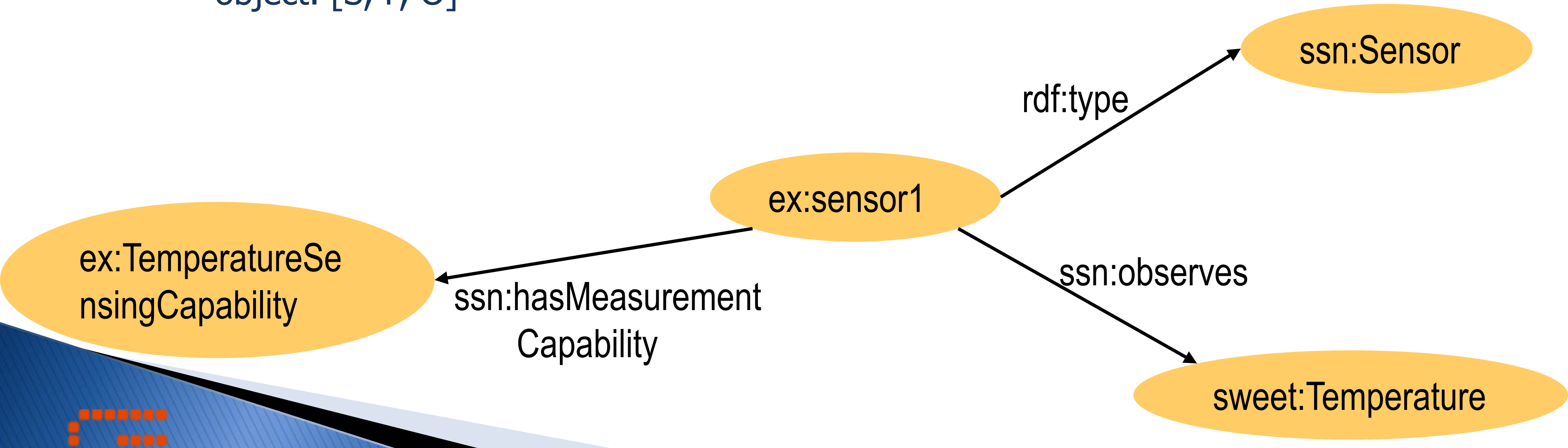
The vision of the **Semantic Sensor Web: annotate sensor data** and services to enable discovery, integration, interoperability etc.

Sensor annotations involve **thematic, spatial and temporal metadata.**

How about using RDF?

RDF: Resource Description Framework

- W3C recommendation
- RDF is graphical formalism (+ XML syntax + semantics)
  - For representing metadata
  - For describing the semantics of information in a machine- accessible way
  - Resources are described in terms of properties and property values using RDF statements
  - Statements are represented as triples, consisting of a subject, predicate and object. [S, P, O]



Motivation (cont'd)

How about using RDF?

Good idea. But **RDF can represent only thematic metadata** properly. We want to take into account spatial and temporal information to aid sensor publication and discovery

What can we do about spatial and temporal metadata?

- Answer: Extend RDF to represent spatial and temporal metadata.

Our Approach

- Resource metadata is modeled using stRDF, an extension of RDF, that can be used to represent thematic, spatial and temporal metadata.
- Resource metadata are queried using stSPARQL, an extension to SPARQL for querying stRDF data.
- Design and implement a storage and query evaluation module for stRDF/stSPARQL.
- Design and implement a semantic registry that stores metadata about SSW resources which is used for resource discovery using thematic, spatial and temporal criteria.

Outline

1. Introduction
2. **The Data Model stRDF**
3. The Query Language stSPARQL
4. The System Strabon
5. The Semantic Registry
6. Experimental evaluation
7. Future Work

From RDF to sRDF - Example

```
ex:sensor1 rdf:type ex:Sensor ;
           ex:measures ex:Temperature ;
           ex:hasLocation ex:location1 .

ex:location1 strdf:hasSpatialExtent
       "POINT(37.94194 23.63722)
        <http://srid.org/ref/epsg/4326/>"
        ^^ogc:WKT .
```

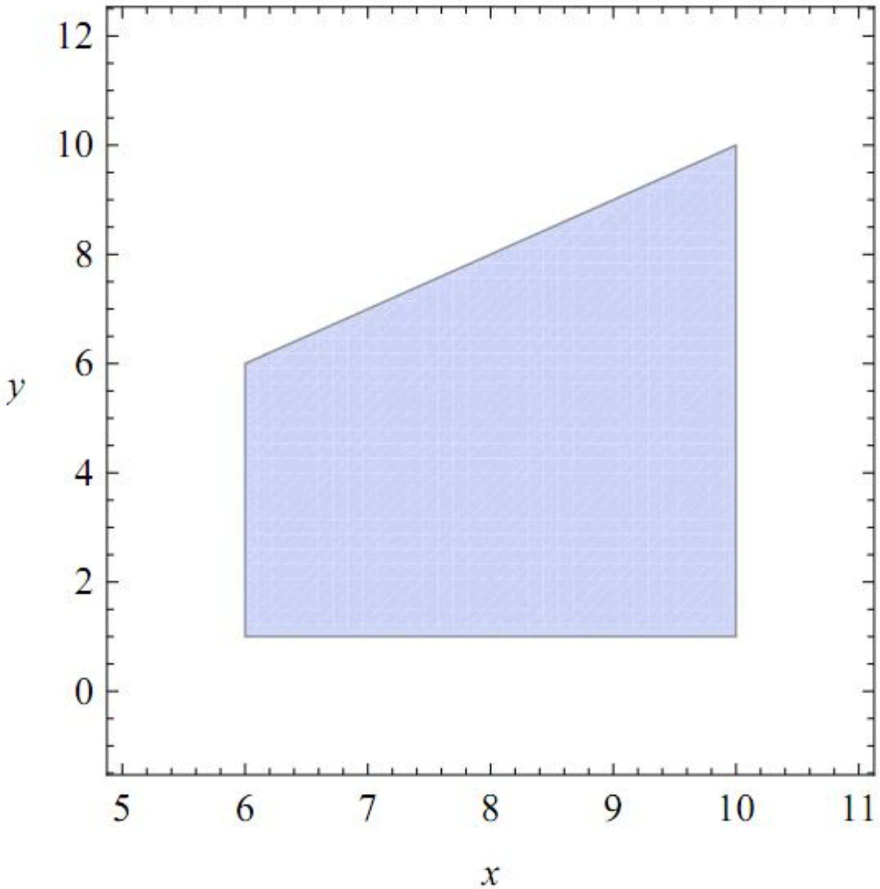
New Kind of Typed Literals

RDF

Spatial Literals

sRDF

Spatial Metadata - Example



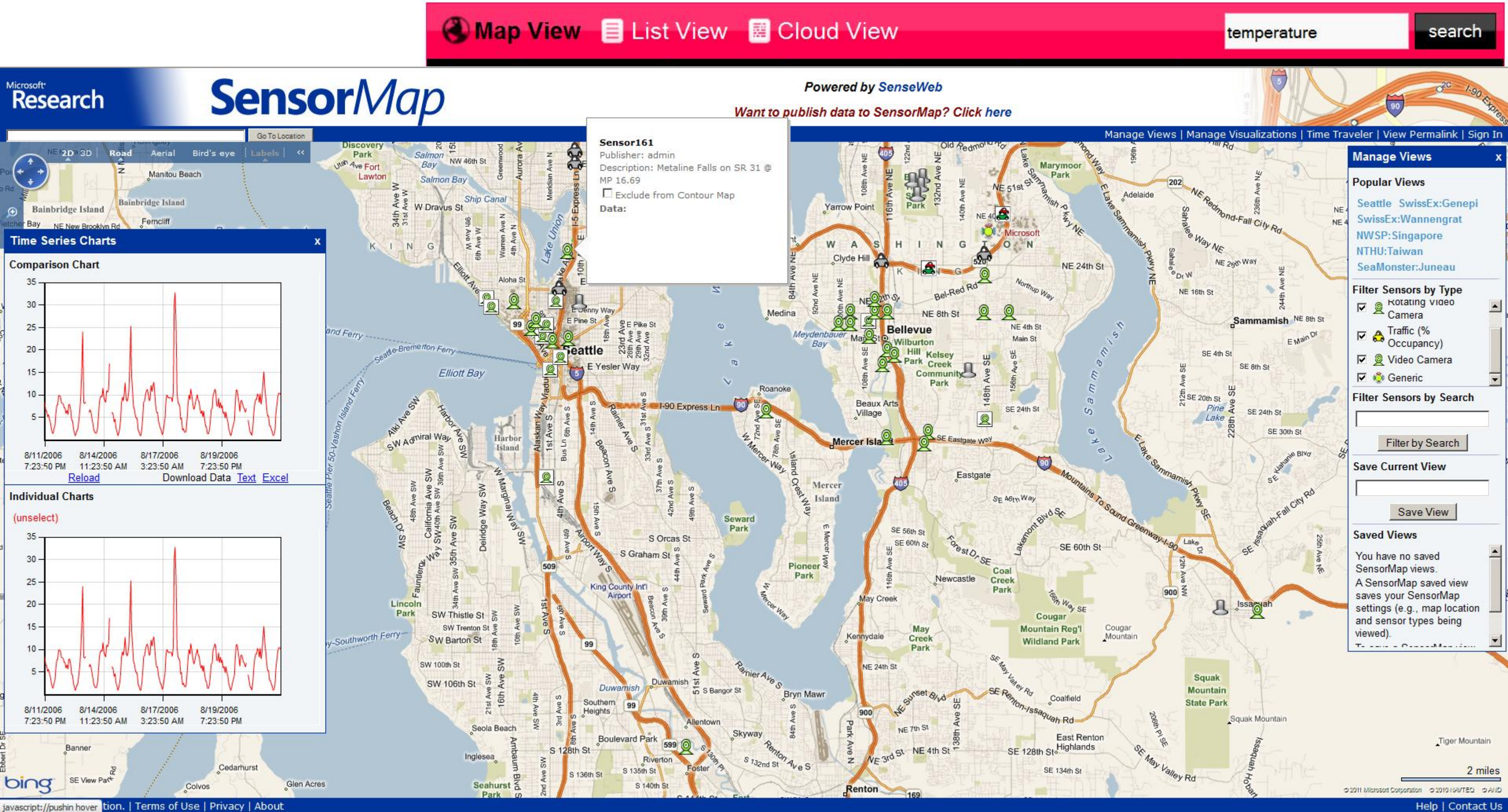

```
POLYGON((6 1,10 1,
          10 10,6 6,
          6 1))
```

Outline

1. Introduction
2. The Data Model stRDF
3. **The Query Language stSPARQL**
4. The System Strabon
5. The Semantic Registry
6. Experimental evaluation
7. Future Work

Example – Dataset

- Collection of sensor metadata



Example – Dataset

- Metadata about a deployed sensor:

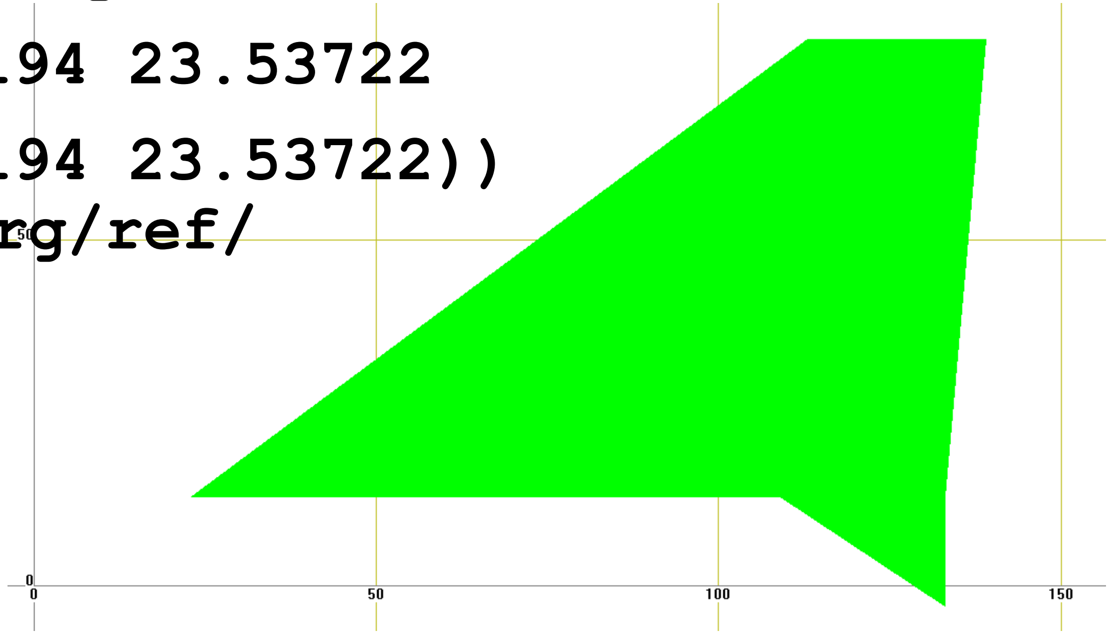
```
ex:sensor1 rdf:type ex:Sensor ;
           ex:measures ex:Temperature ;
           ex:hasLocation ex:location1 .

ex:location1 strdf:hasSpatialExtent
      "POINT(37.94194 23.63722)
       <http://srid.org/ref/epsg/4326/>"
       ^^ogc:WKT .
```

Example – Dataset

- Metadata about geographical areas:

```
ex:area1 rdf:type ex:UrbanArea ;
         ex:hasName "Athens" ;
         strdf:hasSpatialExtent
 "POLYGON((37.74194 23.53722
   … … … … 37.74194 23.53722))
  <http://srid.org/ref/
   epsg/4326/>"
   ^^ogc:WKT
```



Example – Queries

- **Spatial selection.** Find the URIs of the sensors that are inside the rectangle R(37, 23, 38, 24).

```
select ?S
where {?S rdf:type ex:Sensor ;
          ssn:haslocation ?L .
       ?L strdf:hasSpatialExtent ?GEO .
       filter(?GEO inside
             "POLYGON((37 23,37 24,38 24,
              38 23,37 23))<http://srid.org/
              ref/epsg/4326/>"^^ogc:WKT) }
```

Answer

?S
ex:sensor1

Example – Queries

- **Spatial join.** Find the URIs of the sensors that are located inside an urban Area.

```
select ?S
where {?S  rdf:type ssn:Sensor ;
           ssn:haslocation ?L .
       ?UA rdf:type ex:UrbanArea ;
           strdf:hasSpatialExtent ?UAGEO.
     filter(?GEO inside ?UAGEO) }
```

Answer

?S
ex:sensor1

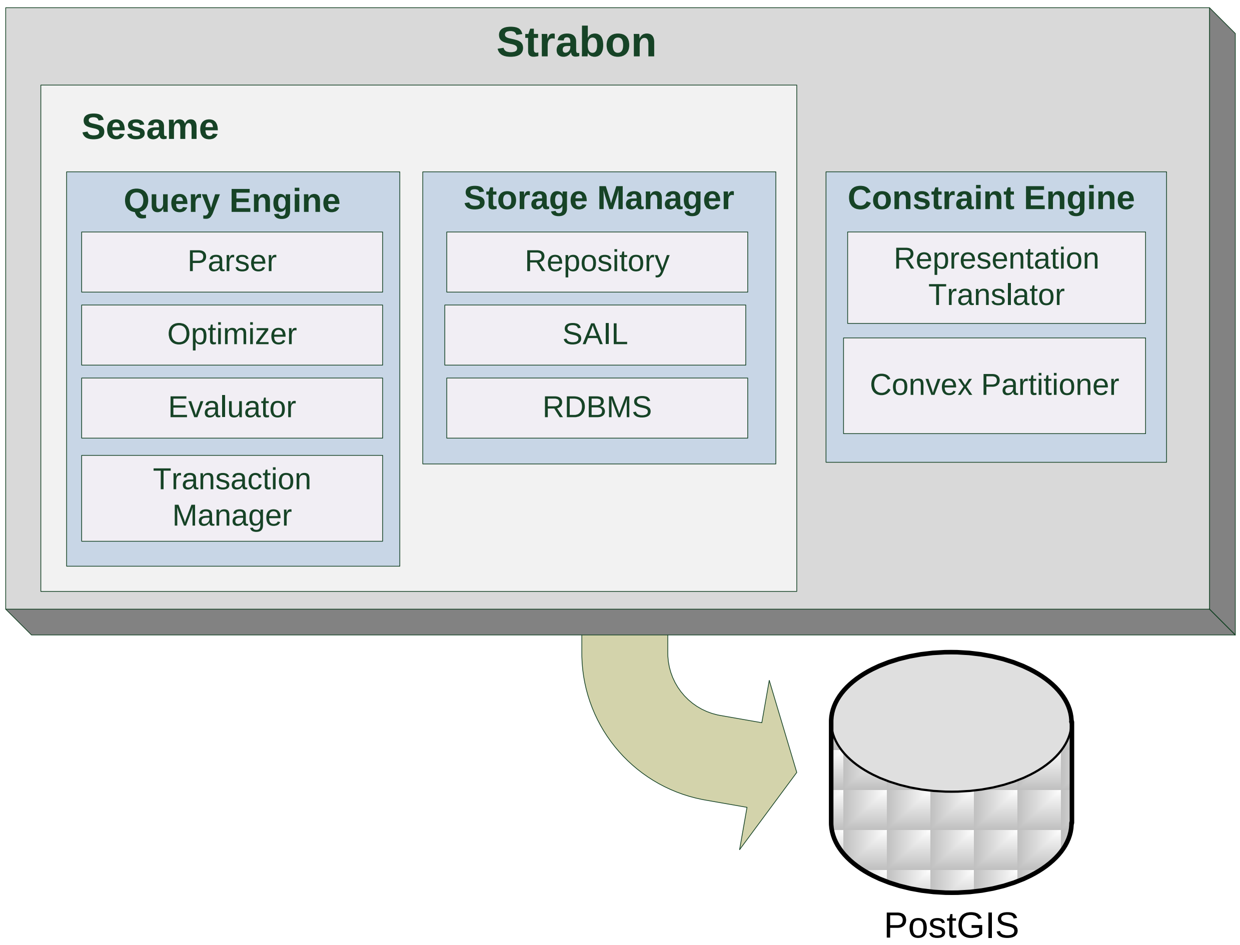
What is new in stSPARQL syntax?

- *k-ary spatial terms*
  - quantifier-free formulas (constants)
  - spatial variables
  - projections of *k-ary spatial terms*
  - the result of **set operations** on *k-ary spatial terms: intersection, union, difference*
  - the result of **geometric operations** on *k-ary spatial terms***:** *boundary, buffer, minimum bounding box*
- Metric spatial terms
  - *VOL, AREA, LEN, MAX, MIN*
- **Select clause:** construction of new spatial terms
  - *intersection, union, difference, projection of spatial terms*
- **Where clause:** Quad patterns to refer to the valid time of a triple
- **Filter clause:**
  - **Spatial predicates (topological):** *disjoint, touch, equals, inside, covered by, contains, covers, overlap*
  - **Temporal predicates:** *before, equal, meets, overlaps, during, starts, finishes*

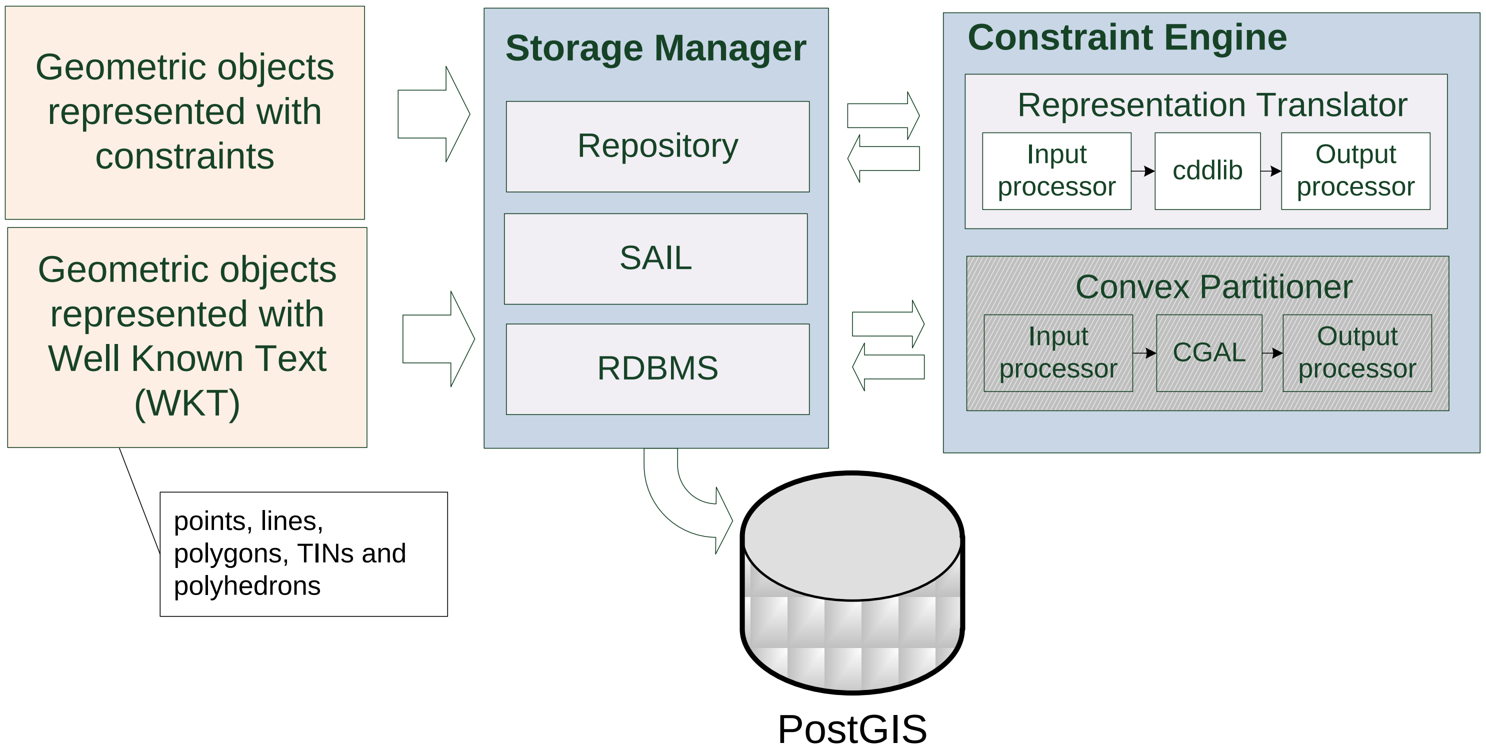
Outline

1. Introduction
2. The Data Model stRDF
3. The Query Language stSPARQL
4. **The System Strabon**
5. The Semantic Registry
6. Experimental evaluation
7. Future Work

The System Strabon



Storing stRDF Data



Storage Scheme

type_2

SUBJECT	OBJECT
1	3

measures_4

SUBJECT	OBJECT
1	5

hasLocation_6

SUBJECT	OBJECT
1	7

hasSpatialExtent_8

SUBJECT	OBJECT
1	9

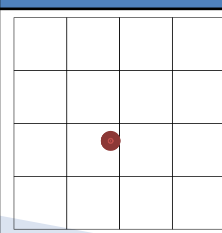
uri_values

ID	VALUE
1	ex:sensor1
2	rdf:type
3	ex:Sensor
4	ex:measures
5	ex:Temperature
6	ex:hasLocation
7	ex:location1
8	strdf:hasSpatialExtent

label_values

ID	VALUE
9	POINT(37.94194 23.63722) <http://srid.org/ref/epsg/4326/>

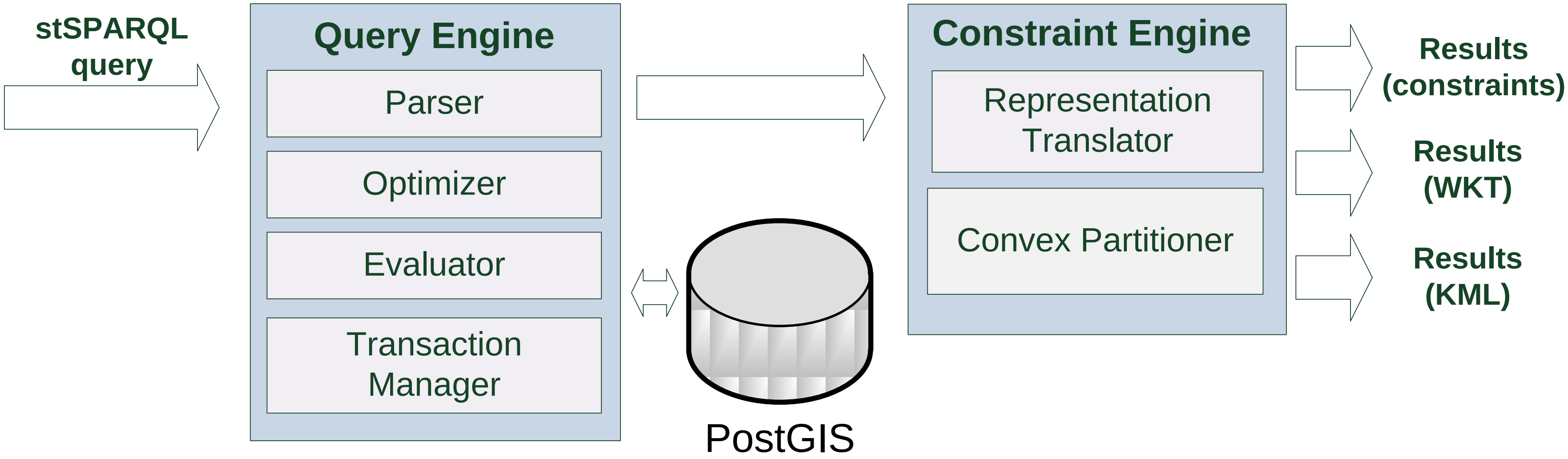
geo_values

ID	VALUE
9	

datatype_values

ID	VALUE
9	ogc:WKT

Evaluating stSPARQL Queries



Evaluating stSPARQL Queries (1/5)

- **Spatial selection.** Find the URIs of the sensors that are inside the rectangle R(37, 23, 38, 24).

```
select ?S
where {?S rdf:type ex:Sensor ;
          ssn:haslocation ?L .
       ?L strdf:hasSpatialExtent ?GEO .
       filter(?GEO inside
             "POLYGON((37 23,37 24,38 24,
              38 23,37 23))<http://srid.org/
              ref/epsg/4326/>"^^ogc:WKT) }
```

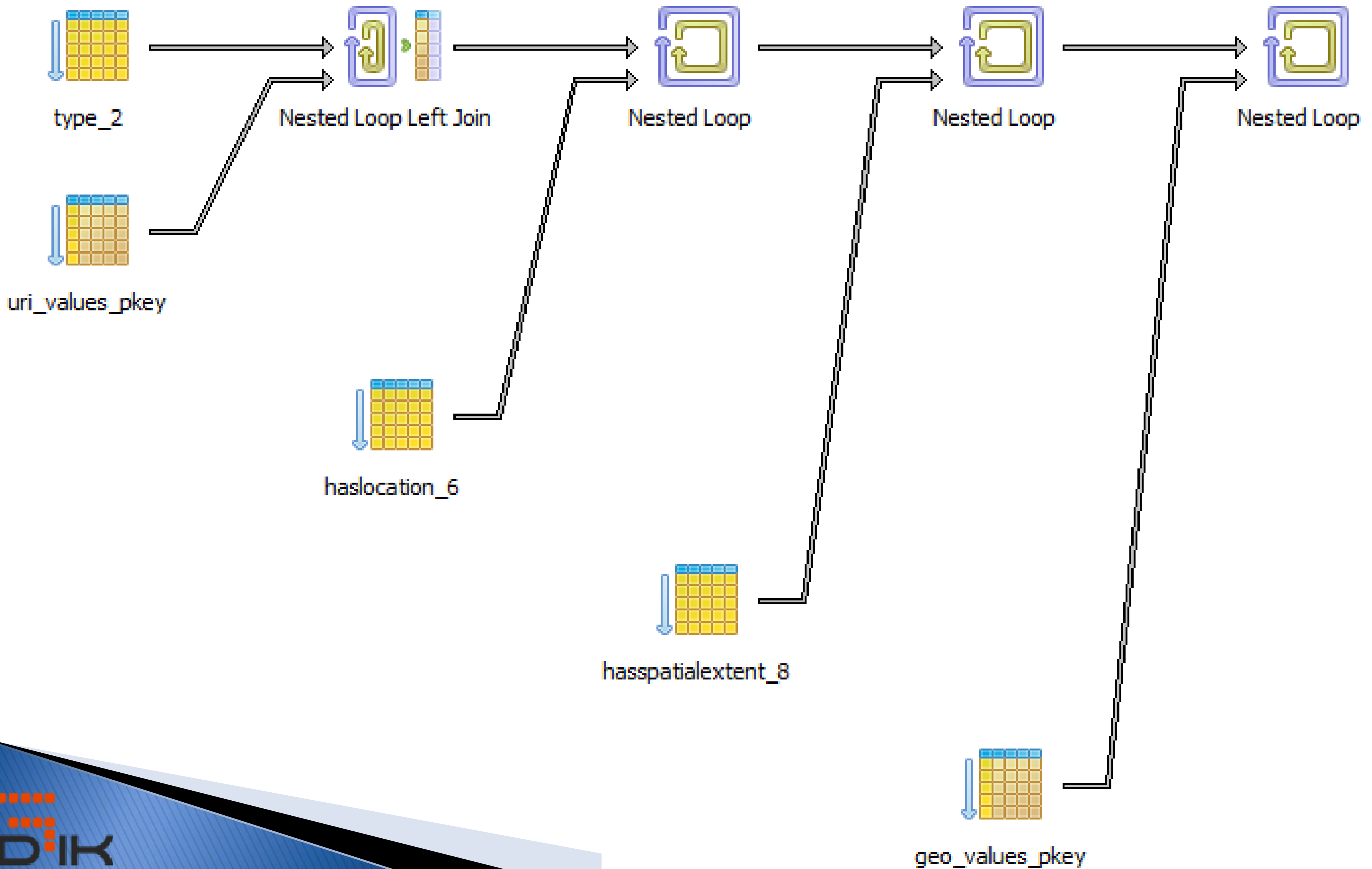
Evaluating stSPARQL Queries (2/5)

?S rdf:type exSecs?s
?S ssn:haslocation ?L
?L strdf:hasSpatialExtent ?GEO
filter(?GEO inside "POLYGON...

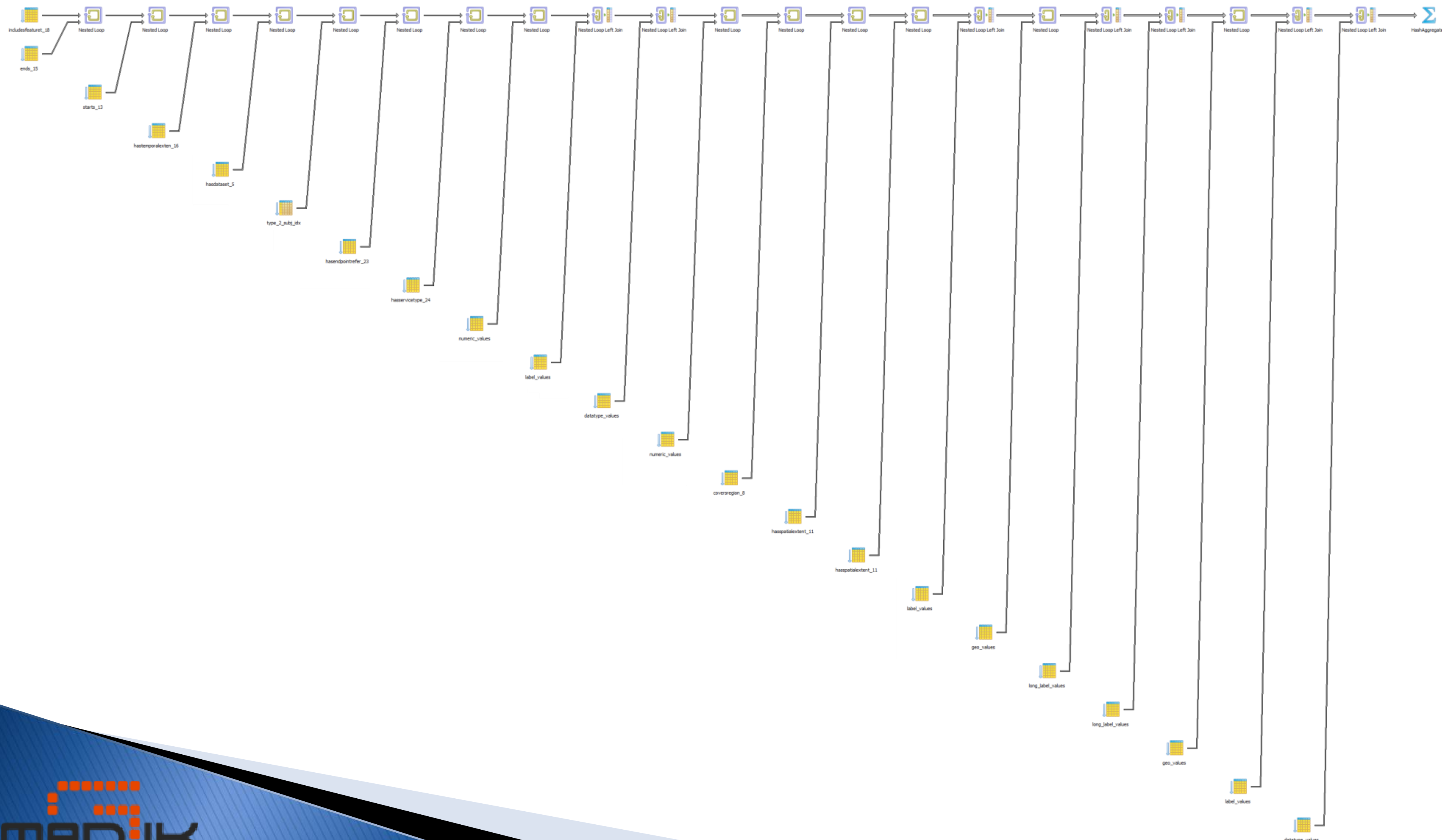
SQL Query

```
SELECT u_S.value
FROM type_2 t0
INNER JOIN haslocation_6 h1 ON (h1.subj=t0.subj)
INNER JOIN hasspatialextent_8 h2 ON (h2.subj=h1.obj)
INNER JOIN geo_values l_GEO ON (l_GEO.id=h2.obj)
LEFT JOIN uri_values u_S ON (u_S.id=t0.subj)
WHERE t0.obj=3
 AND (l_GEO.strdfgeo @  ST_GeomFromText('POLYGON
  ((37 23, 37 24, 38 24, 38 23, 37 23))',4326))
```

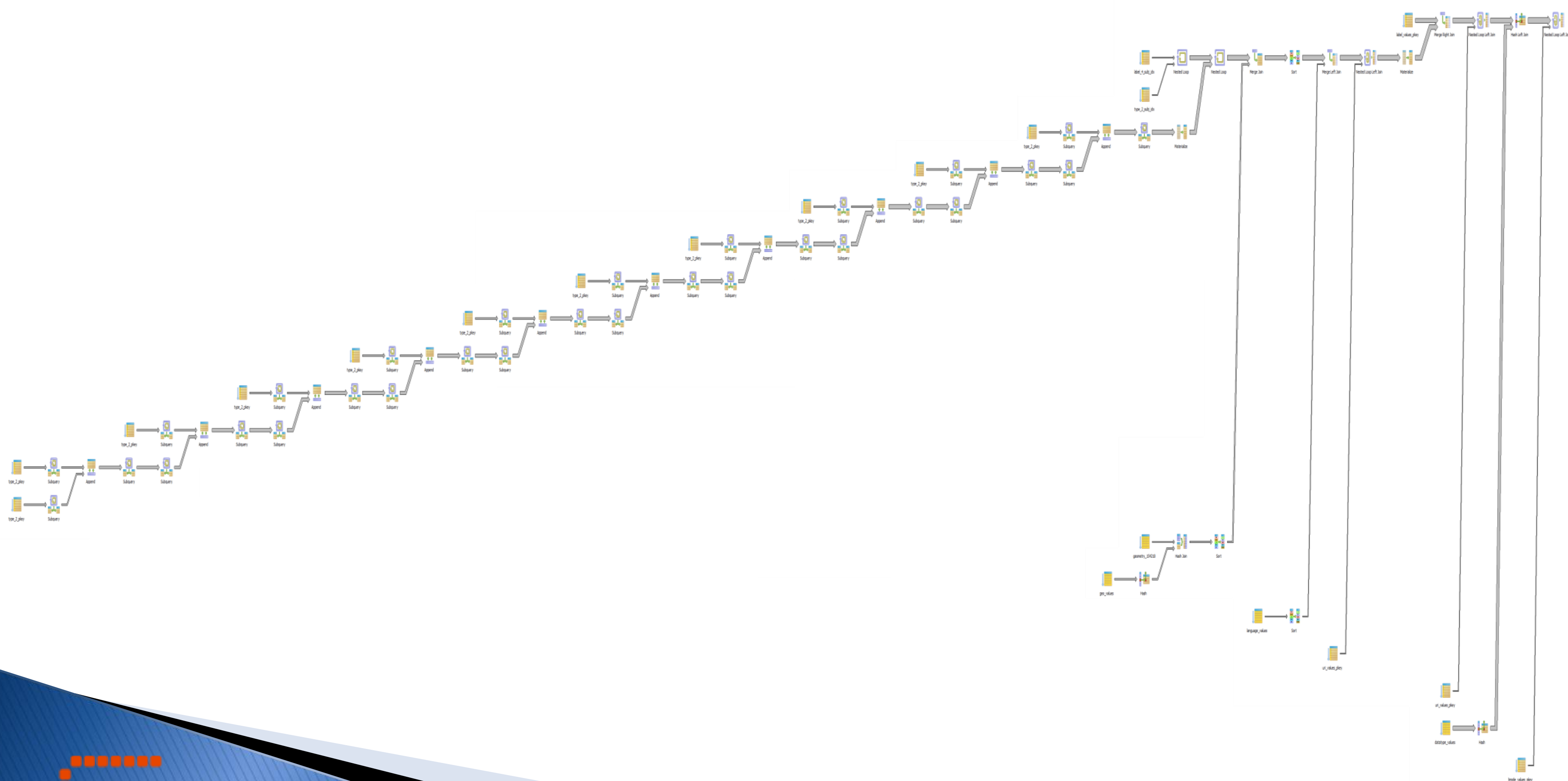
Evaluating stSPARQL Queries (3/5)



Evaluating stSPARQL Queries (4/5)



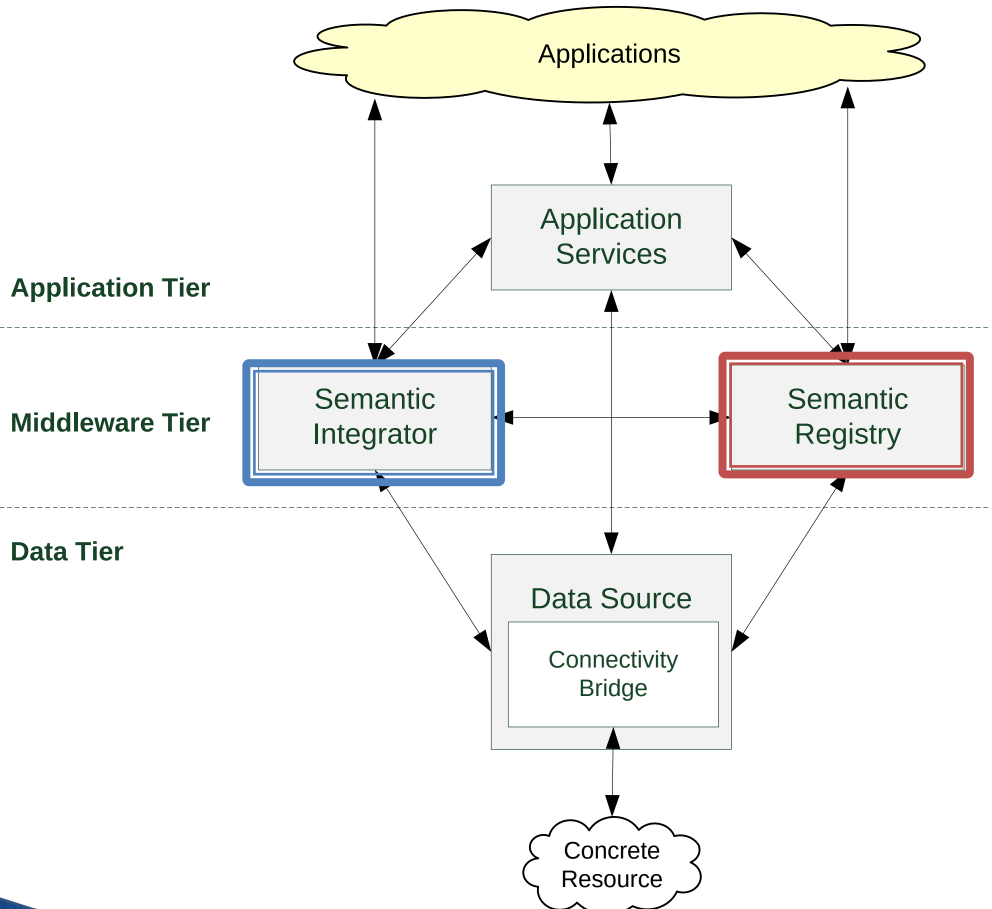
Evaluating stSPARQL Queries (5/5)



Outline

1. Introduction
2. The Data Model stRDF
3. The Query Language stSPARQL
4. The System Strabon
5. **The Semantic Registry**
6. Experimental evaluation
7. Future Work

[SemsorGrid4Env] A service oriented architecture for a Semantic Sensor Grid

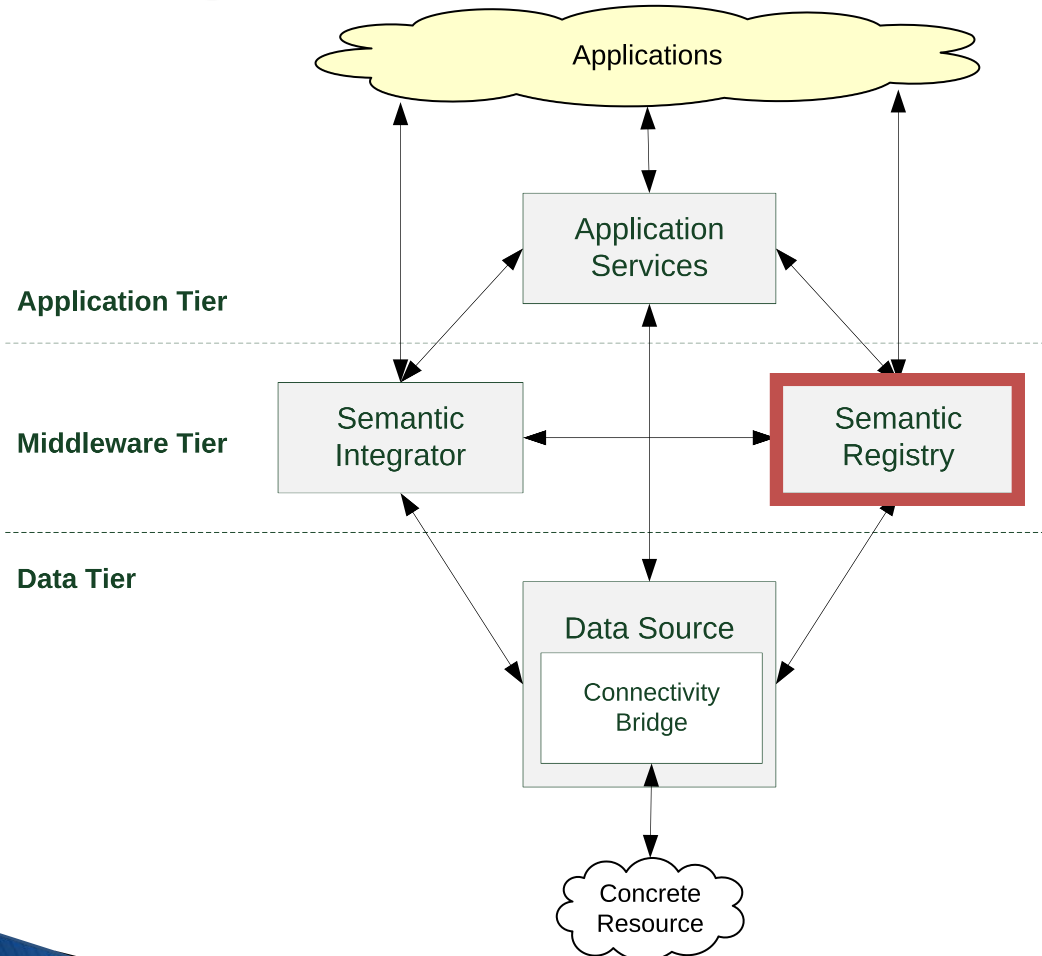


Services that provide domain specific functionality to consumers & applications

A service that can be accessed by prospective clients who want to manage thematic and spatial metadata.

Services that provide access to streaming and stored data sources

The Architecture of the Semantic Registry – First Look



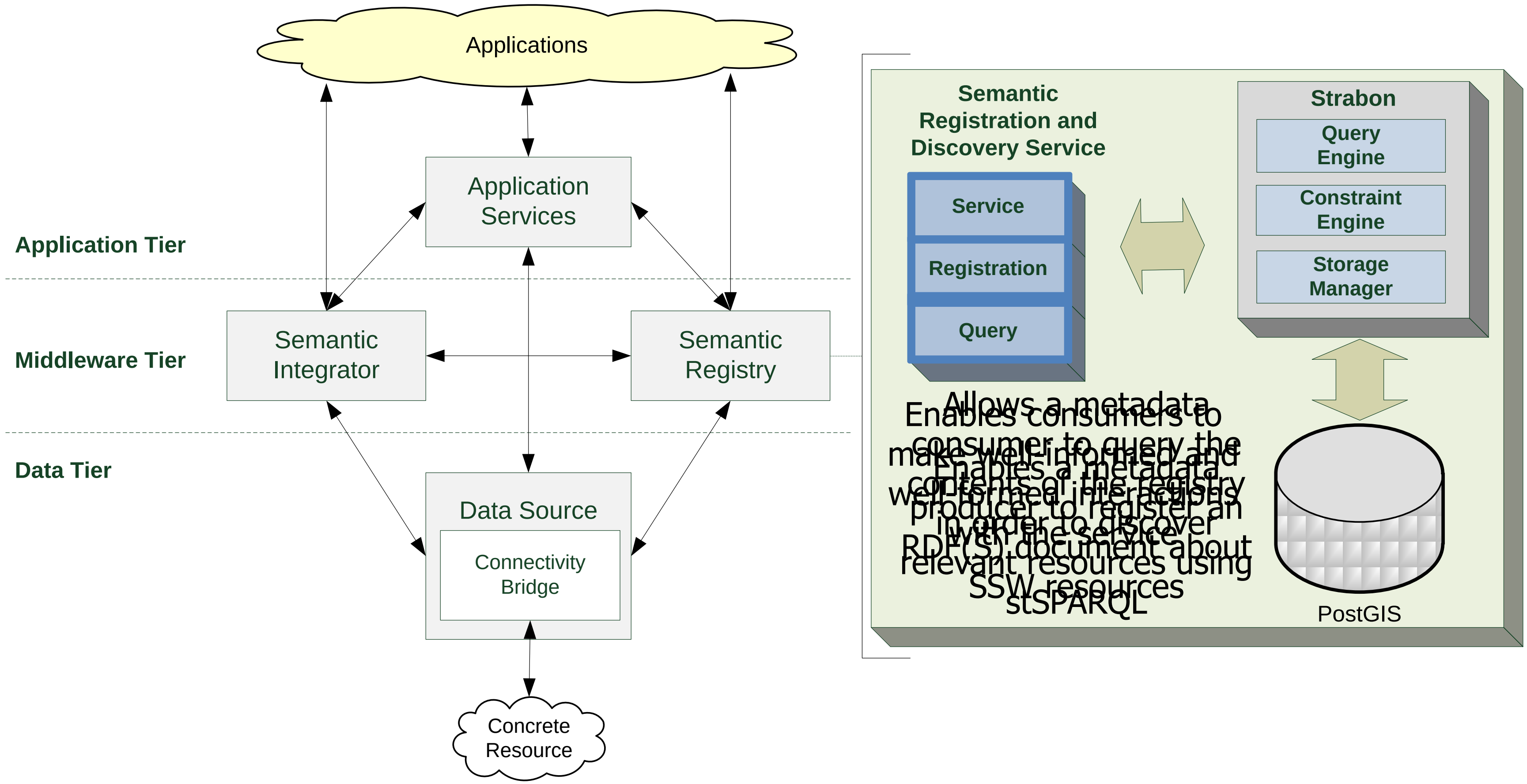
The Semantic Registry stores semantic annotations of available resources:

- sensors
- sensor networks
- data sources
- web services.

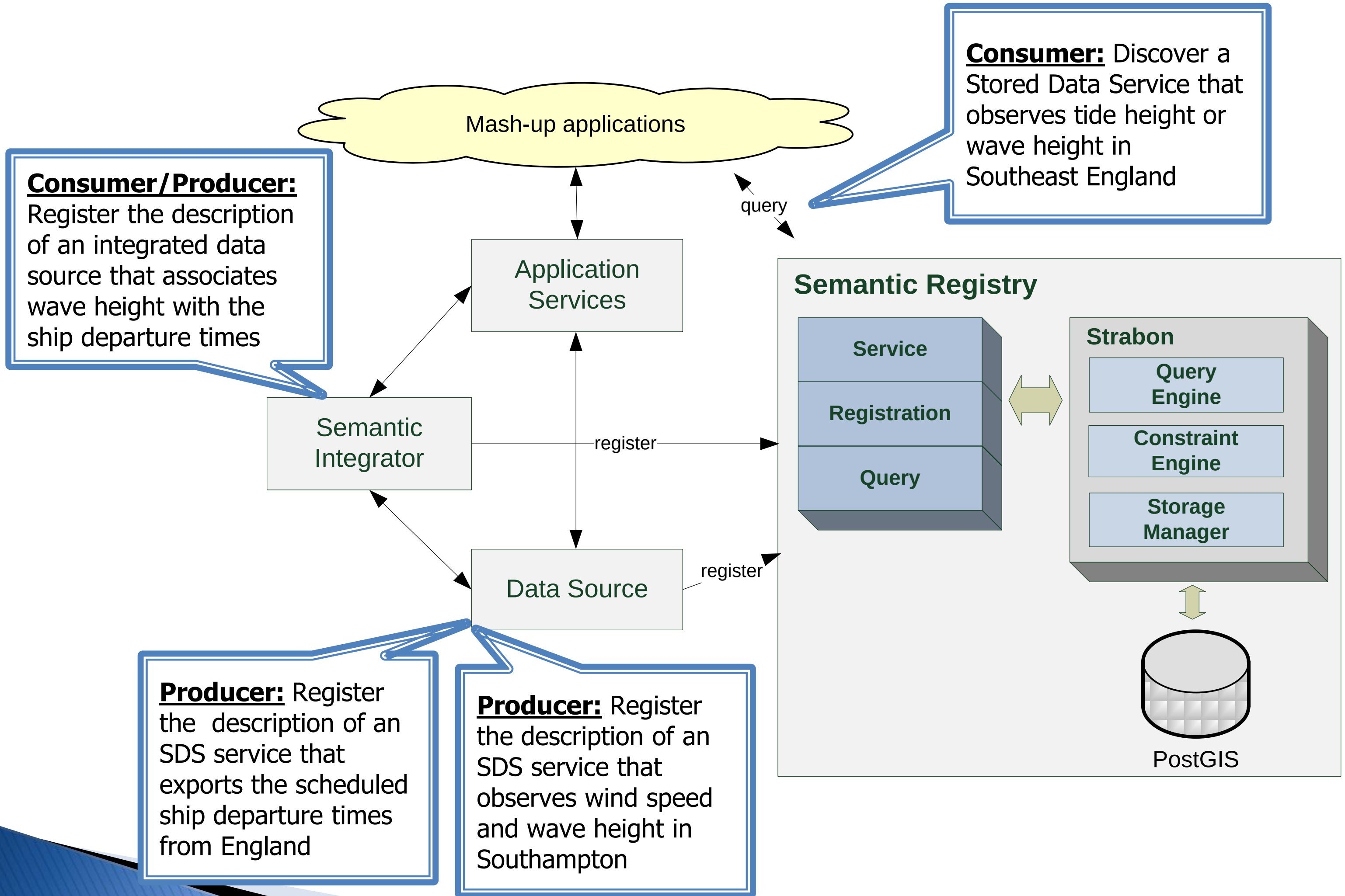
Annotations are registered by service providers.

Clients discover resources by posing queries.

Interfaces Implemented



Example Orchestration



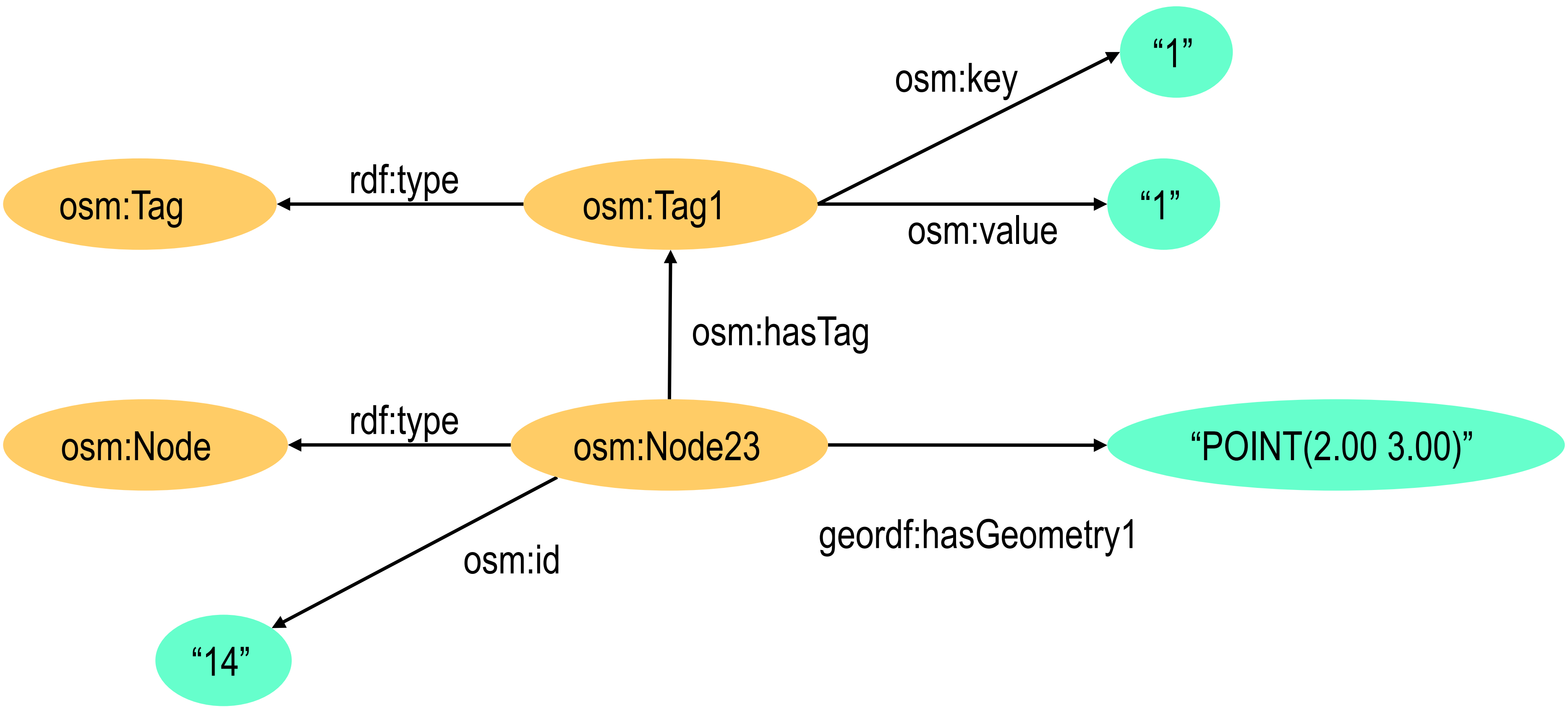
Outline

1. Introduction
2. The Data Model stRDF
3. The Query Language stSPARQL
4. The System Strabon
5. The Semantic Registry
6. **Experimental evaluation**
7. Future Work

Experimental evaluation

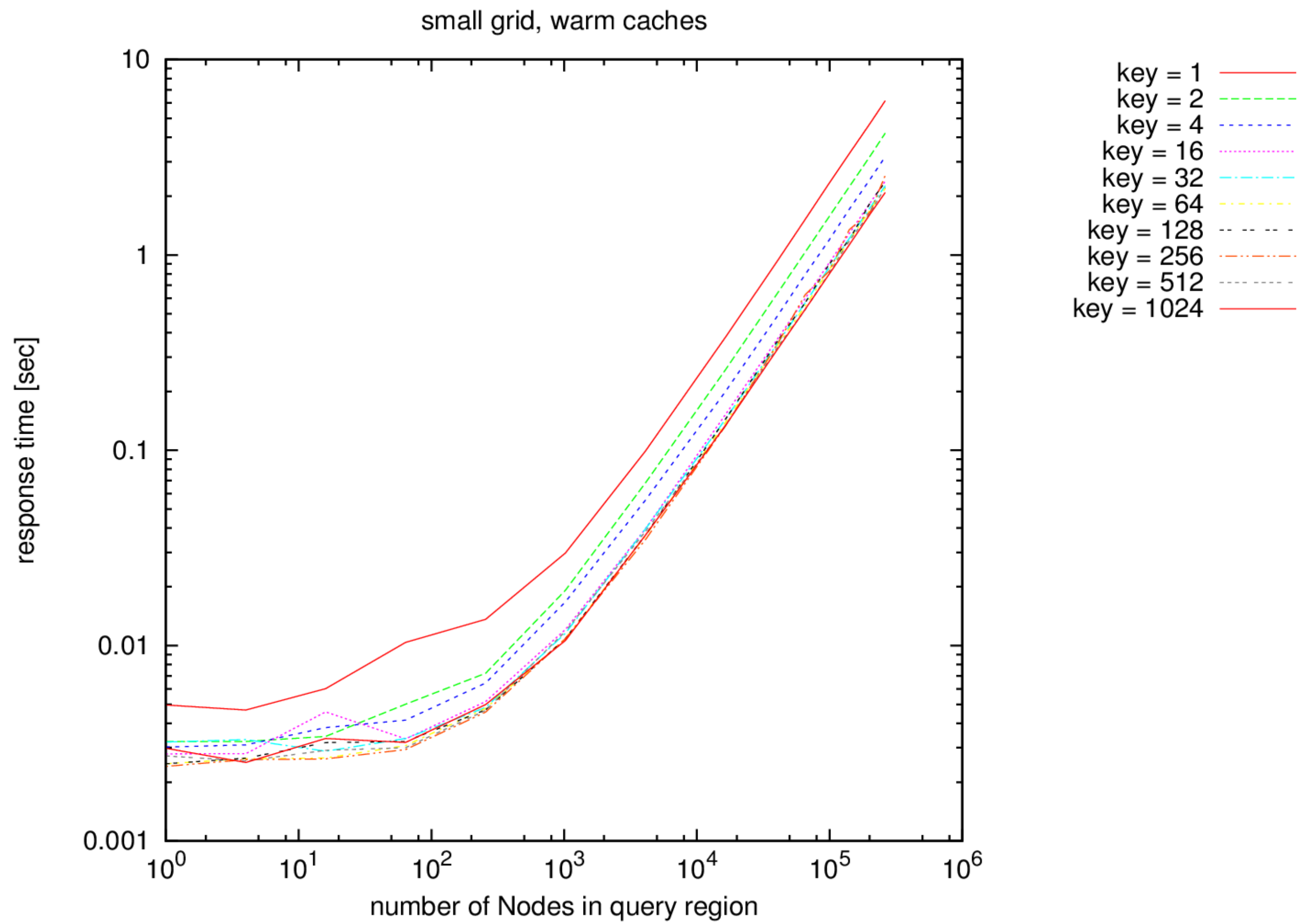
- We used synthetic datasets based on OpenStreetMaps
  - Dataset size: 10 million triples (2GB) - 1 billion triples (110GB)
  - Triples with spatial literals: 1.4 million - 140 million triples
- We measured the response time of queries with various thematic and spatial selectivities

Sample data

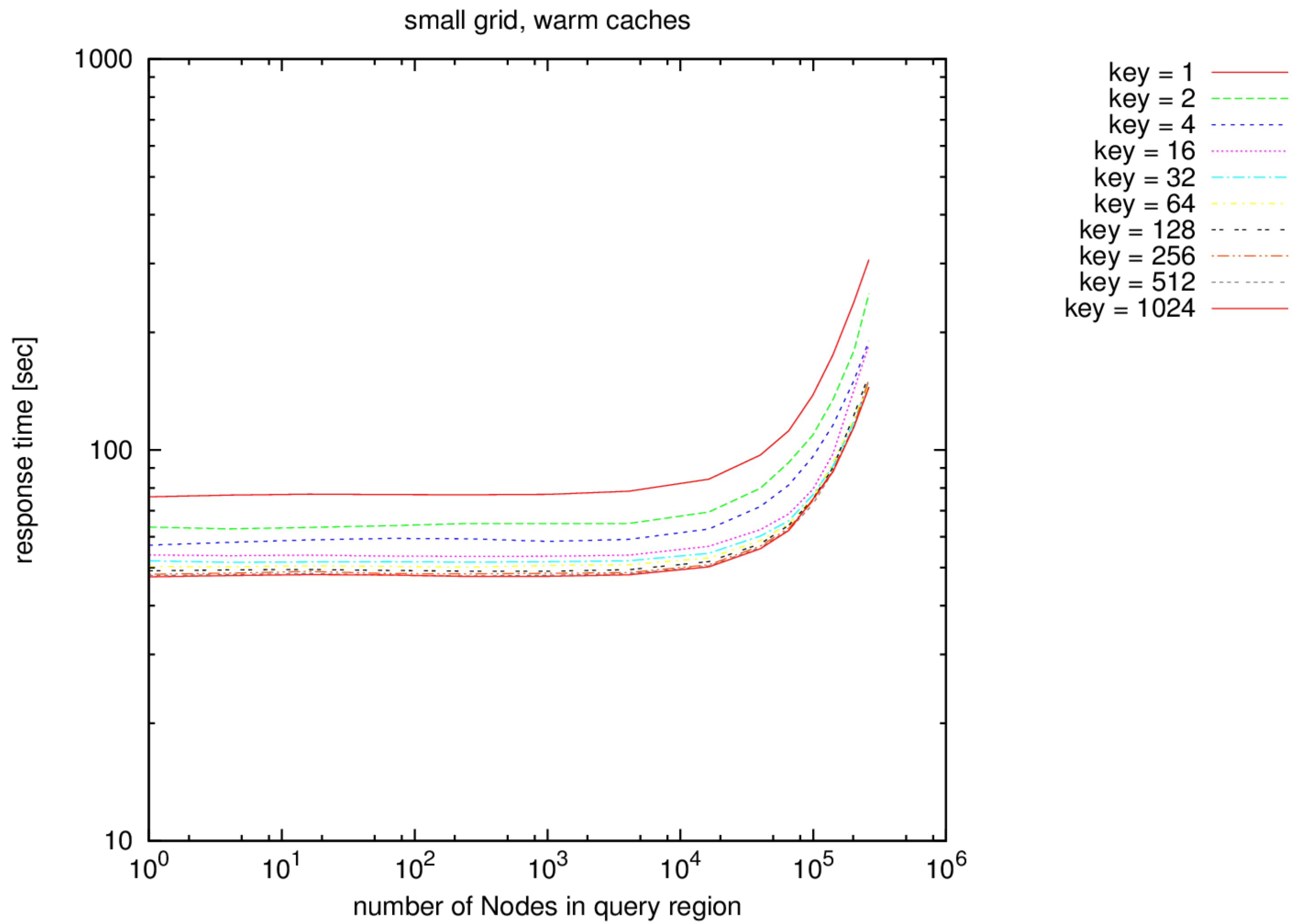


"1"
osm:key
osm:Tag
rdf:type
osm:Tag1
"1"
osm:value
osm:hasTag
osm:Node
rdf:type
osm:Node23
"POINT(2.00 3.00)"
osm:id
geordf:hasGeometry1
"14"

LGD dataset – 10 million triples



LGD dataset – 1 billion triples



Outline

1. Introduction
2. The Data Model stRDF
3. The Query Language stSPARQL
4. The System Strabon
5. The Semantic Registry
6. Experimental evaluation
7. **Future Work**

Future Work

- Evaluate our current implementation.
- Port our current implementation to MonetDB.
- Study the query processing of stRDF/stSPARQL in the MonetDB case.
- Use our implementation to publish public spatial datasets as Linked Open Data.

Thank you for your attention!

Ερωτήσεις;