



Semantic Integration of Streaming Data Sources

Jean-Paul Calbimonte and Oscar Corcho

Ontology Engineering Group. Departamento de Inteligencia Artificial.

Facultad de Informática, Universidad Politécnica de Madrid.

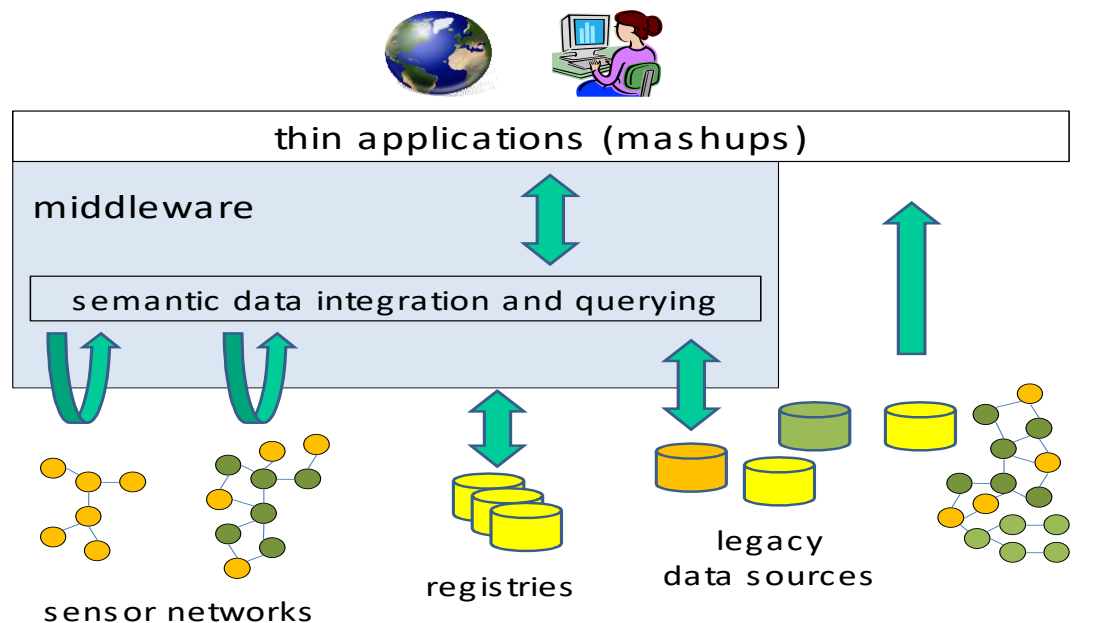
Campus de Montegancedo s/n.

28660 Boadilla del Monte. Madrid. Spain

{jpcalbimonte, ocorcho}@fi.upm.es

Development of an integrated information space where new sensor networks can be easily discovered and integrated with existing ones and possibly other data sources (e.g., historical databases)

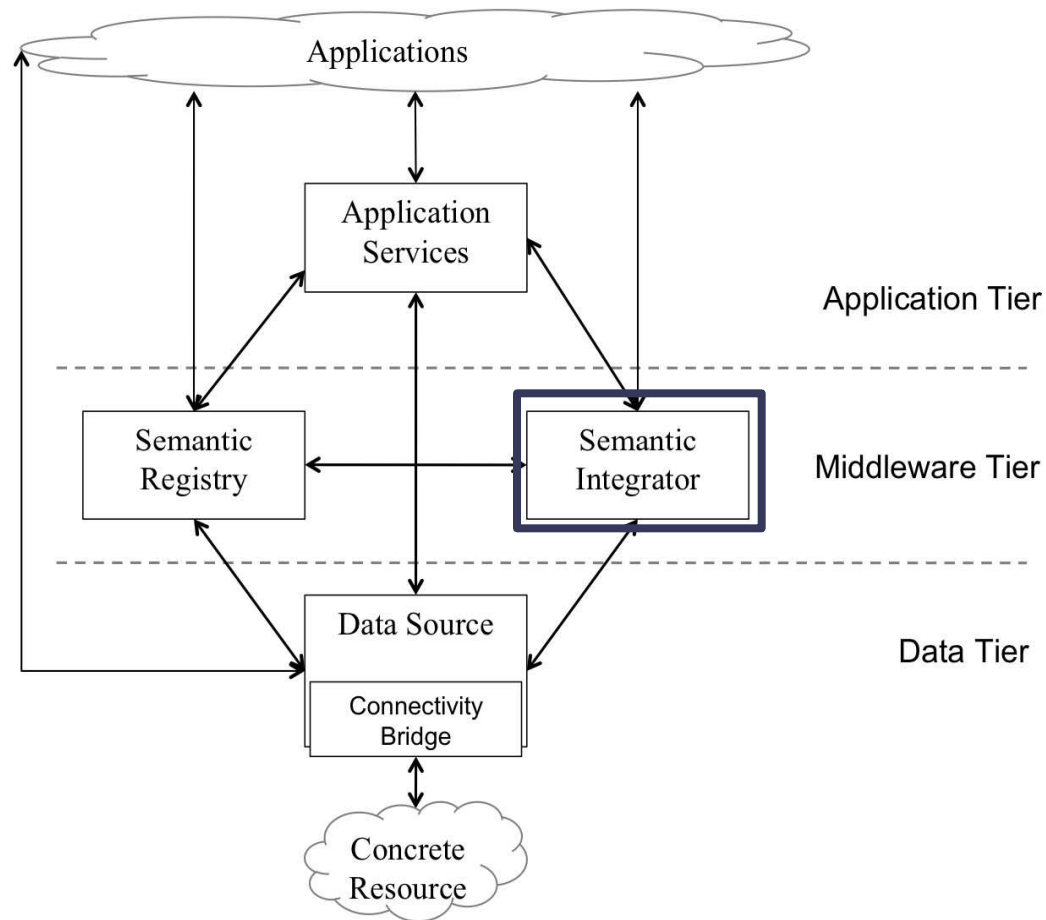
Rapid development of flexible and user-centric decision support systems that use data from multiple autonomous independently deployed sensor networks and other applications.



Semantic Integrator provides data access to an integrated virtual resource that integrates:

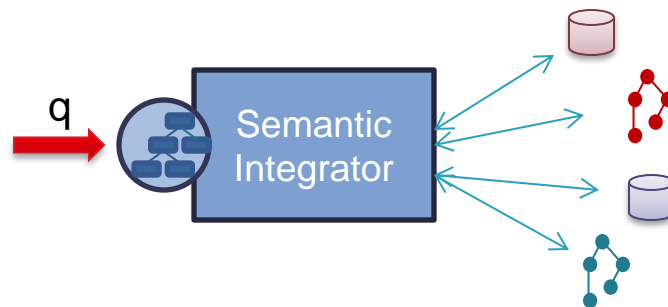
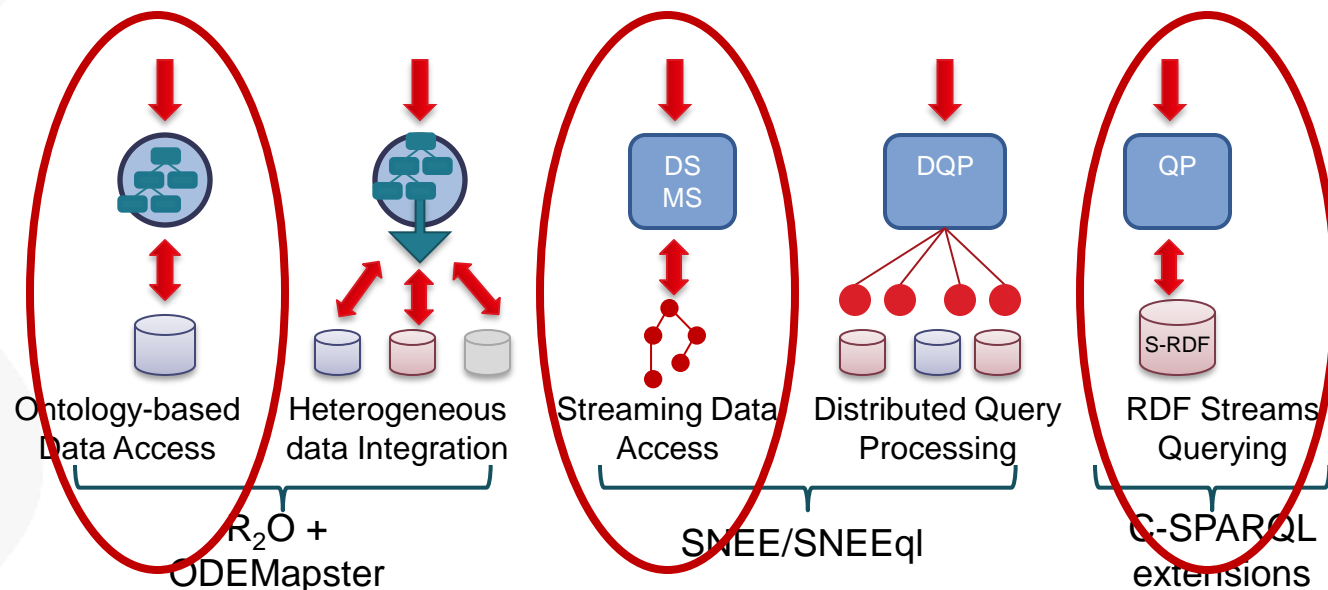
- Streaming data sources
- Stored data sources

It provides a unified ontological view accessible through declarative queries.

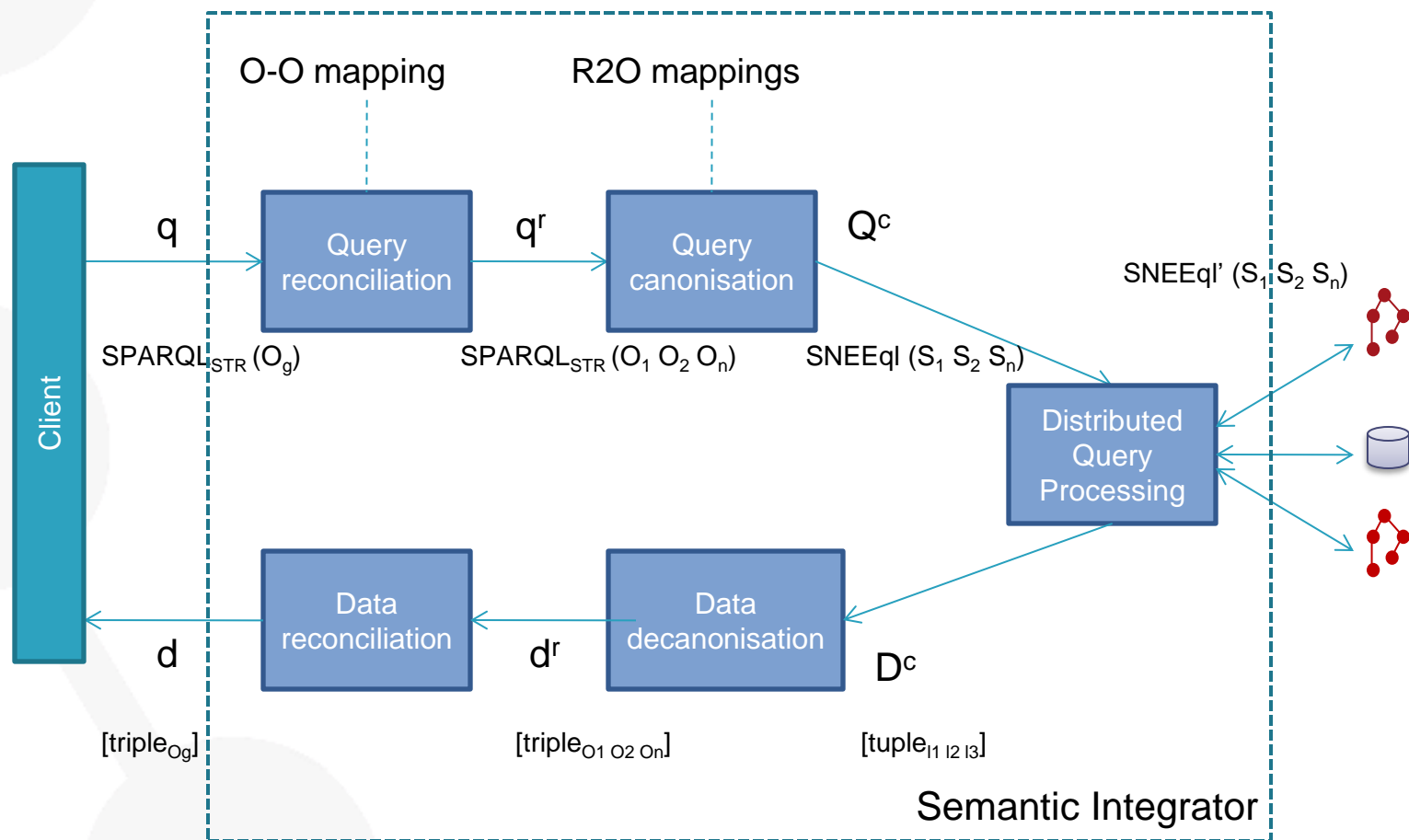


- To design, implement and deploy a *Semantic Integration Service* for the SemSorGrid4Env software architecture
 - To extend existing ontology-based data integration models to take into account sensor networks streaming data, semantic heterogeneity and quality of service
 - To implement and deploy the Semantic Integrator

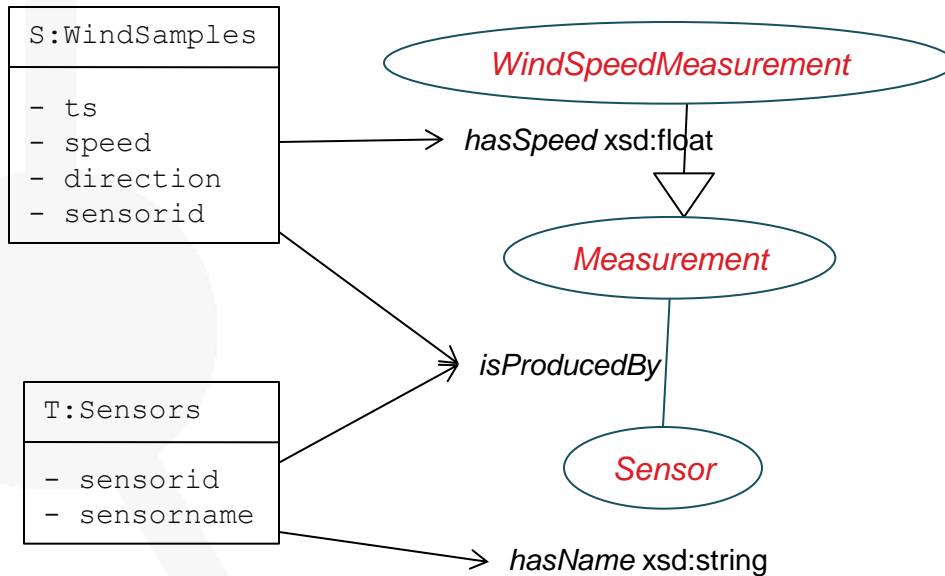
Background: Approaches & Technologies



Ontology-based Streaming Data Access



S2O: Mapping Streams to Ontologies



conceptmap-def *WindSpeedMeasurement*

uri-as

```
concat('ssg4env:WindSpeedMeasurement_',  
        windsamples.sensorid,windsamples.ts)
```

described-by

attributemap-def *hasSpeed*

operation "constant"

has-column windsamples.speed

dbrelationmap-def *isProducedBy* toConcept *Sensor*

joins-via

condition "equals"

has-column sensors.sensorid

has-column windsamples.sensorid

conceptmap-def *Sensor*

uri-as

```
concat('ssg4env:Sensor_',sensors.sensorid)
```

described-by

attributemap-def *hasName*

operation "constant"

has-column sensors.sensorname



```
PREFIX fire: http://www.semsorgrid4env.eu#  
PREFIX rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#  
SELECT ?speed ?name  
FROM STREAM <http://www.ssg4env.eu/Readings.srdf>  
[RANGE 10 MINUTE STEP 1  
MINUTE]  
WHERE {  
  ?WindSpeed a fire:WindSpeedMeasurement;  
  fire:hasSpeed ?speed;  
  fire:isProducedBy ?sensor;  
  fire:hasTimestamp ?time.  
  ?sensor a fire:Sensor;  
  fire:hasName ?name.  
}
```

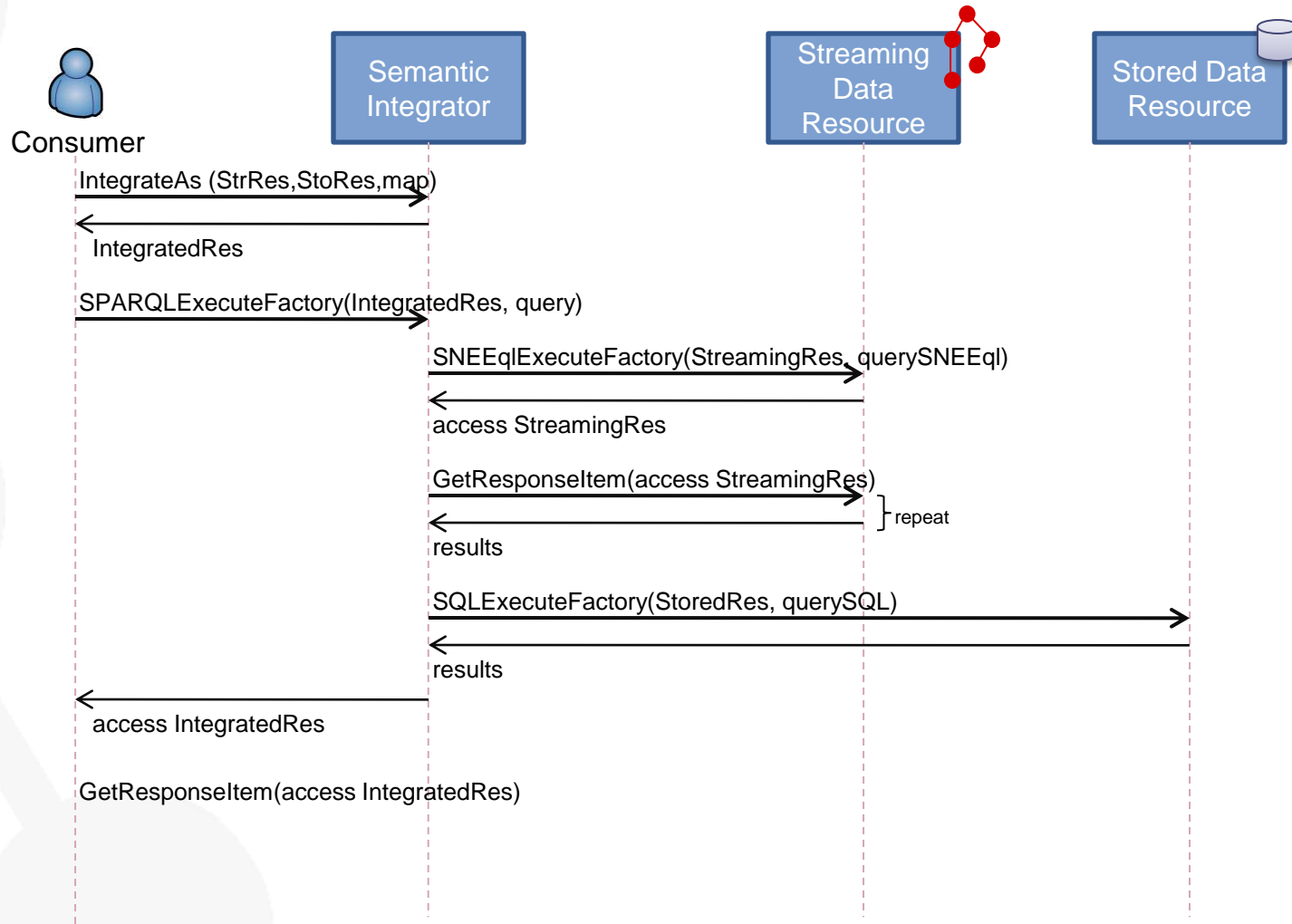
```
SELECT concat( 'ssg4env.eu#Sensor' , sensors.sensorid ) as a1 ,  
          ( sensors.sensorname ) as name  
FROM sensors
```

```
SELECT concat( 'ssg4env.eu#WindSpeedMeasurement' ,  
               windsensor.id , windsensor.ts ) as a1 ,  
          ( windsensor.speed ) as speed  
FROM windsensor[ FROM NOW - 10 TO NOW MIN]
```

```
SELECT concat( 'ssg4env.eu#WindSpeedMeasurement' ,  
               windsensor.id , windsensor.ts ) as a1 ,  
          concat( 'ssg4env.eu#Sensor' , sensors.sensorid ) as a2  
FROM sensors, windsensor[ FROM NOW - 10 TO NOW MIN]  
WHERE ( sensors.sensorid = windsensor.id )
```

Work in progress: removing redundant queries,
basic optimisations, more complex scenarios

Semantic Integration Interactions



- Ontology-based streaming data access
 - Continue implementation of SPARQL stream extensions
 - Multiple graph pattern matching, aggregates, projections
 - Continue implementation of SNEEqL translation
 - Eliminate redundant queries, basic optimizations
 - Support for tuple-based windows, slide parameters
 - Provision as a Service in the SemSorGrid4Env Architecture
- Ontology-based streaming data integration
 - Continue implementation of R2O support for streams
 - Support for push and pull based streams
 - Add mapping of multiple virtual RDF streams
 - Support for quality of service parameters