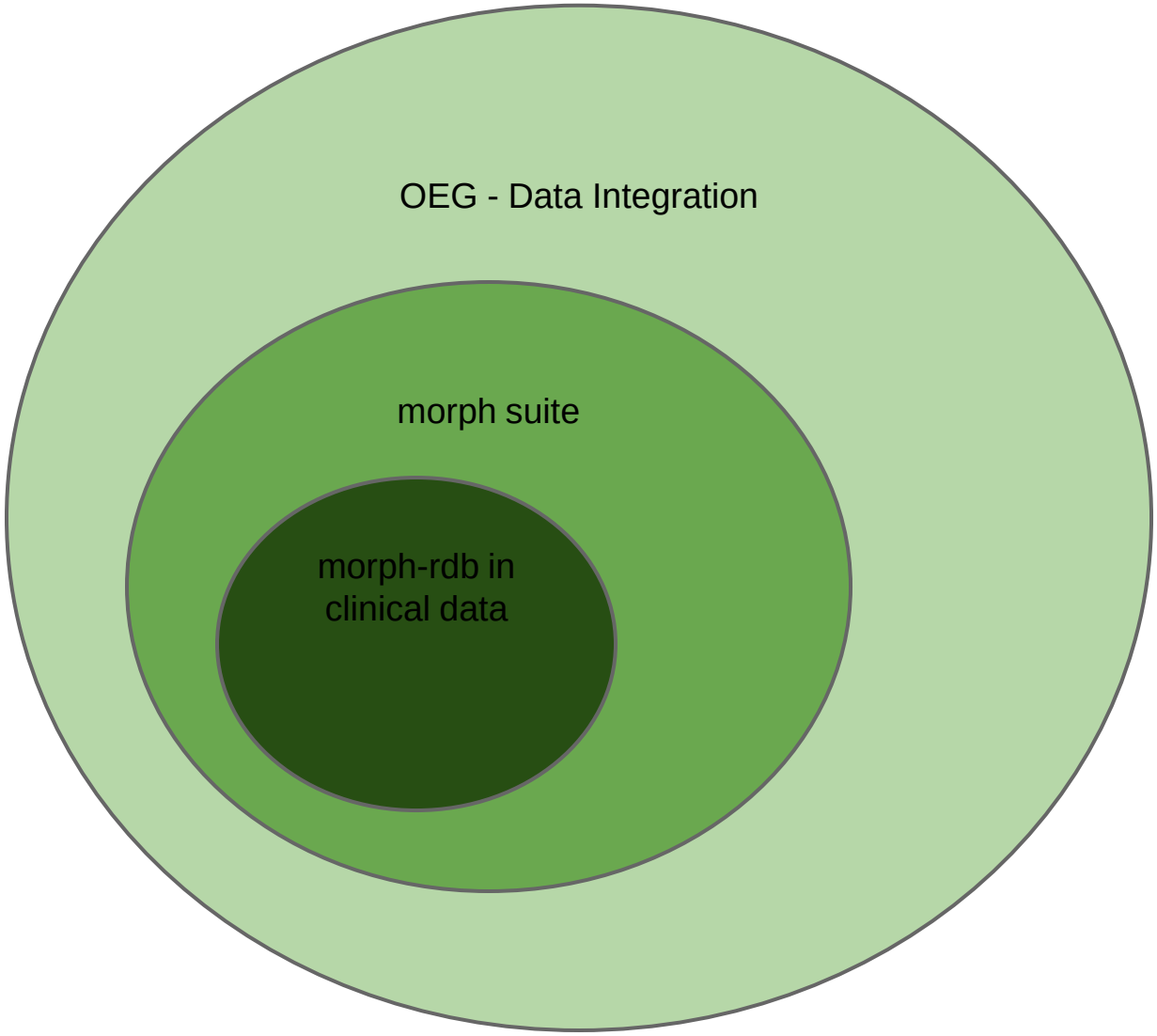
An Overview of the Research Carried Out at the Data Integration Group - OEG

CrEDIBLE Workshop, October 9th, 2014

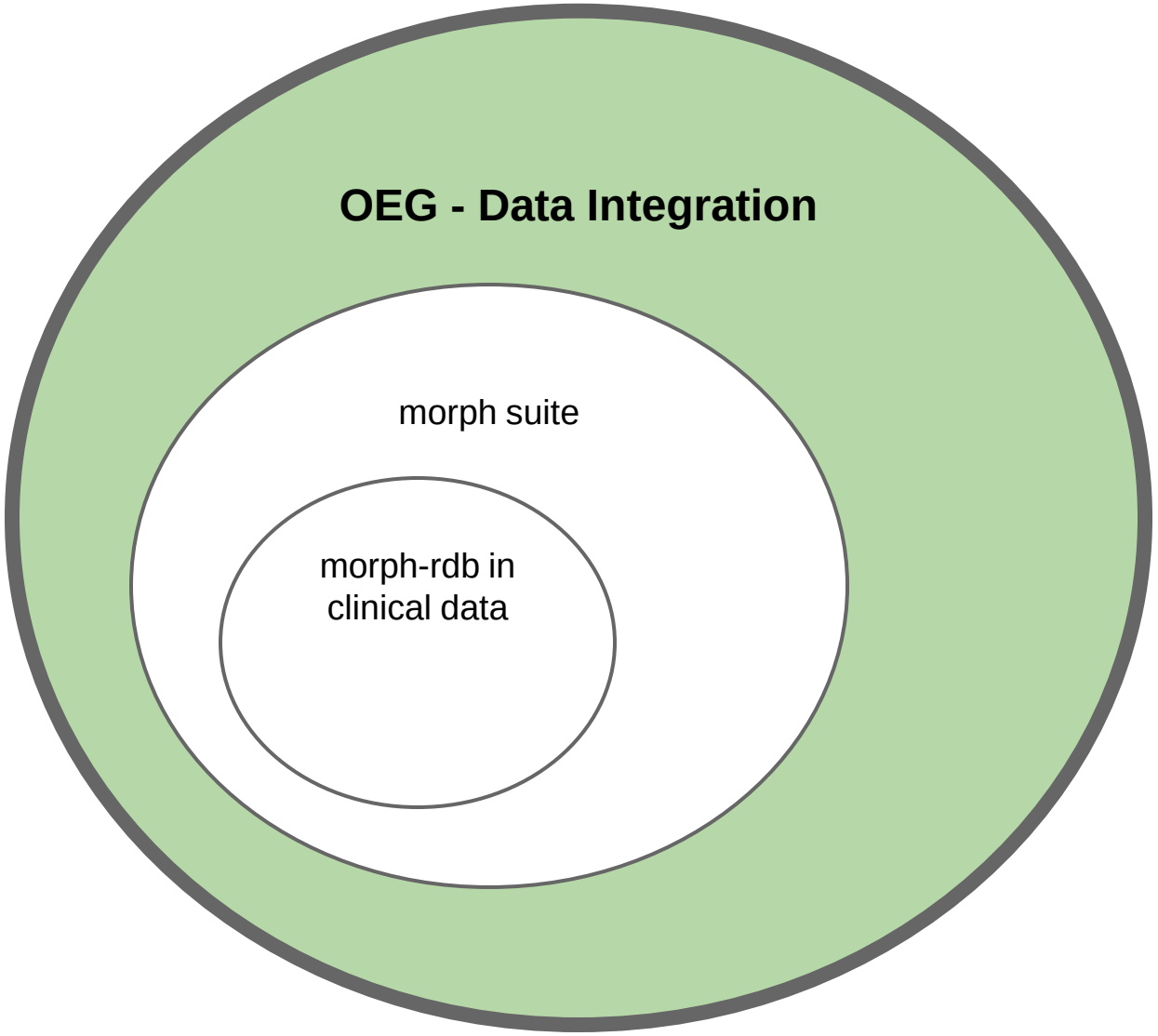
Freddy Priyatna
fpriyatna@fi.upm.es @freddy_priyatna

With contributions from:
Oscar Corcho, Jose Mora (now at Università di Roma - Sapienza), Carlos Buil-Aranda (now at Pontificia Universidad Católica de Chile), Jean Paul Calbimonte (now at École Polytechnique Fédérale de Lausanne), Nandana Mihindukulasooriya, Alejandro Llaves

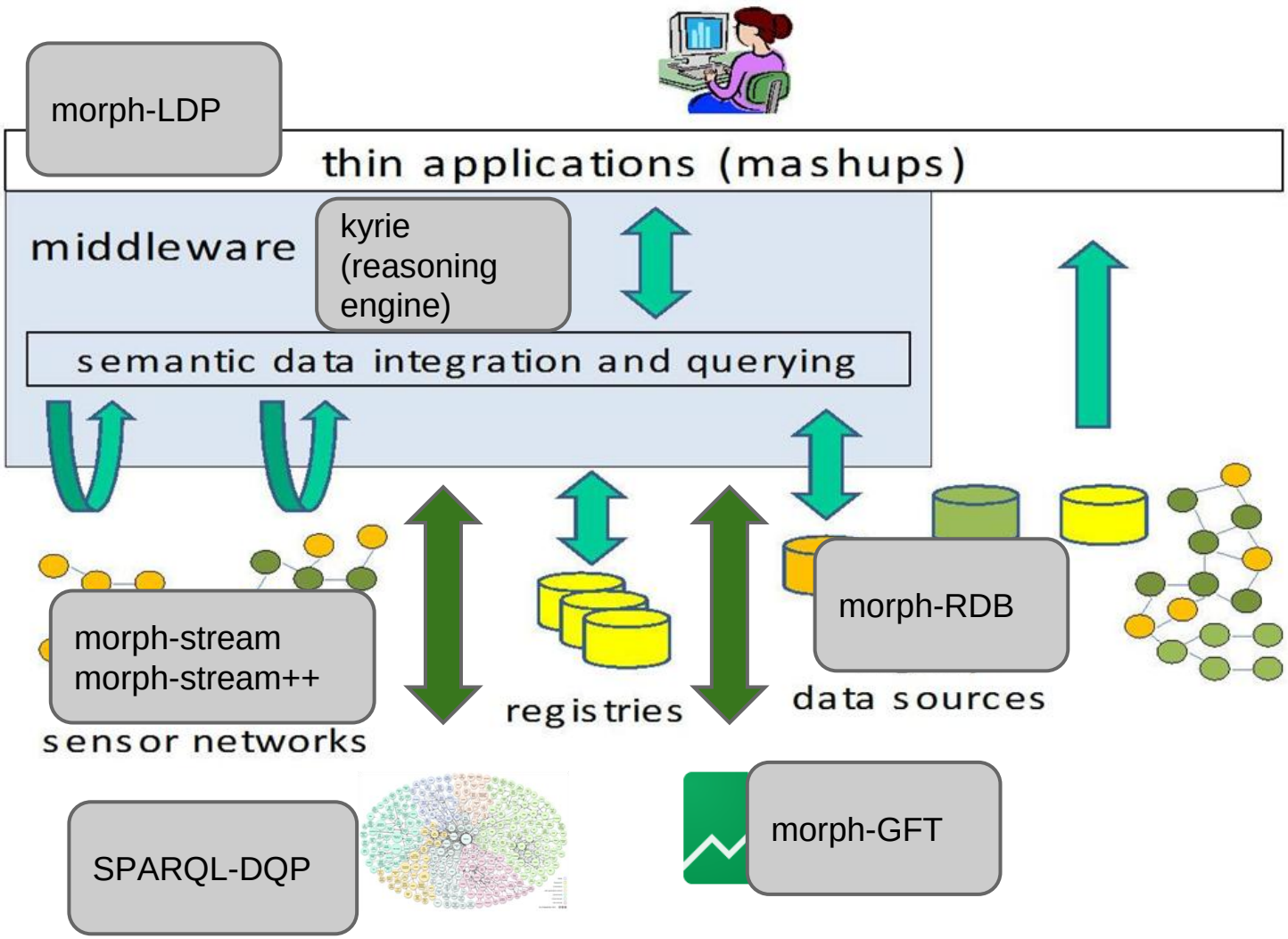
Outline


OEG - Data Integration
morph suite
morph-rdb in
clinical data

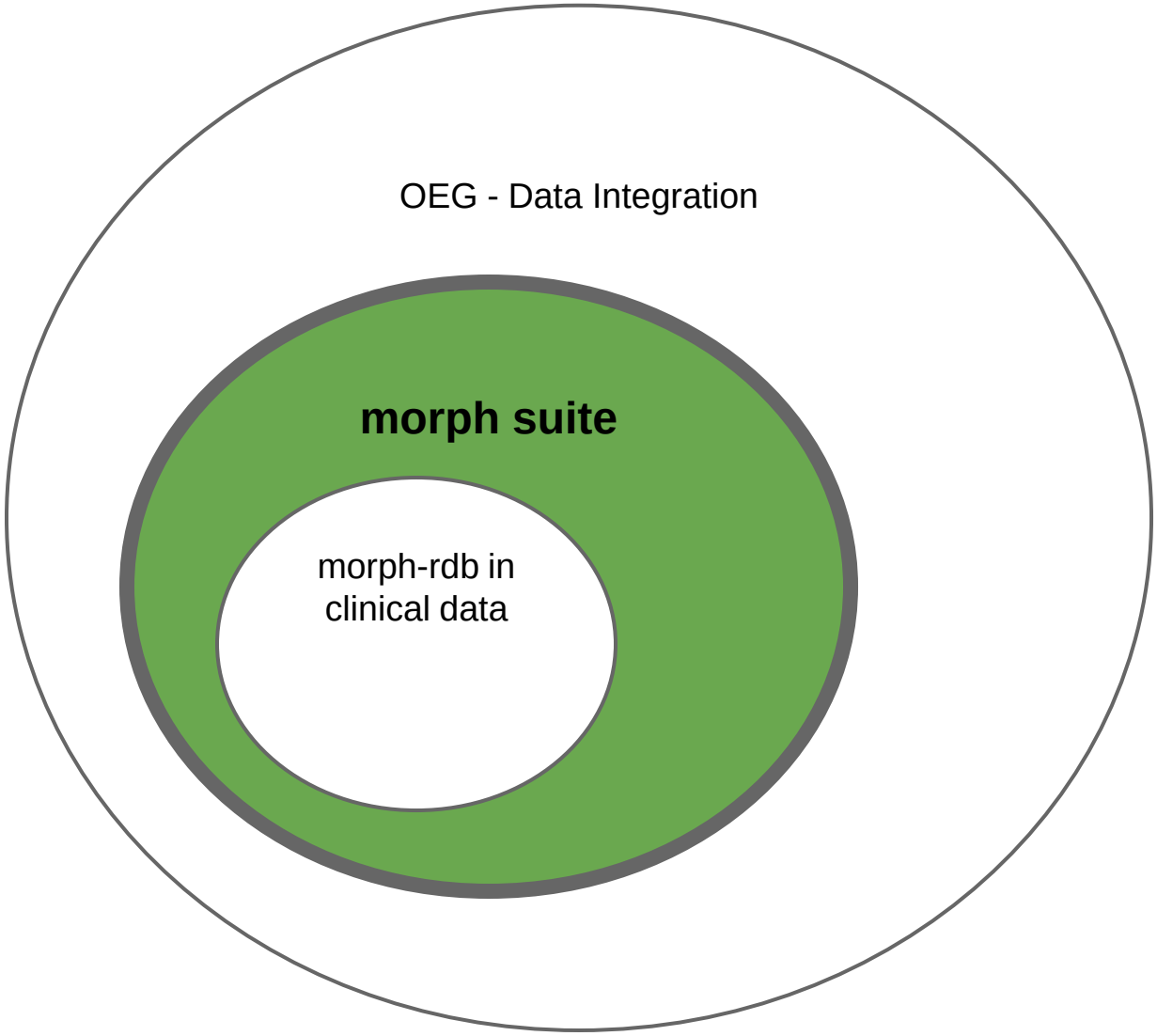
Outline


OEG - Data Integration
morph suite
morph-rdb in
clinical data

Ingredients

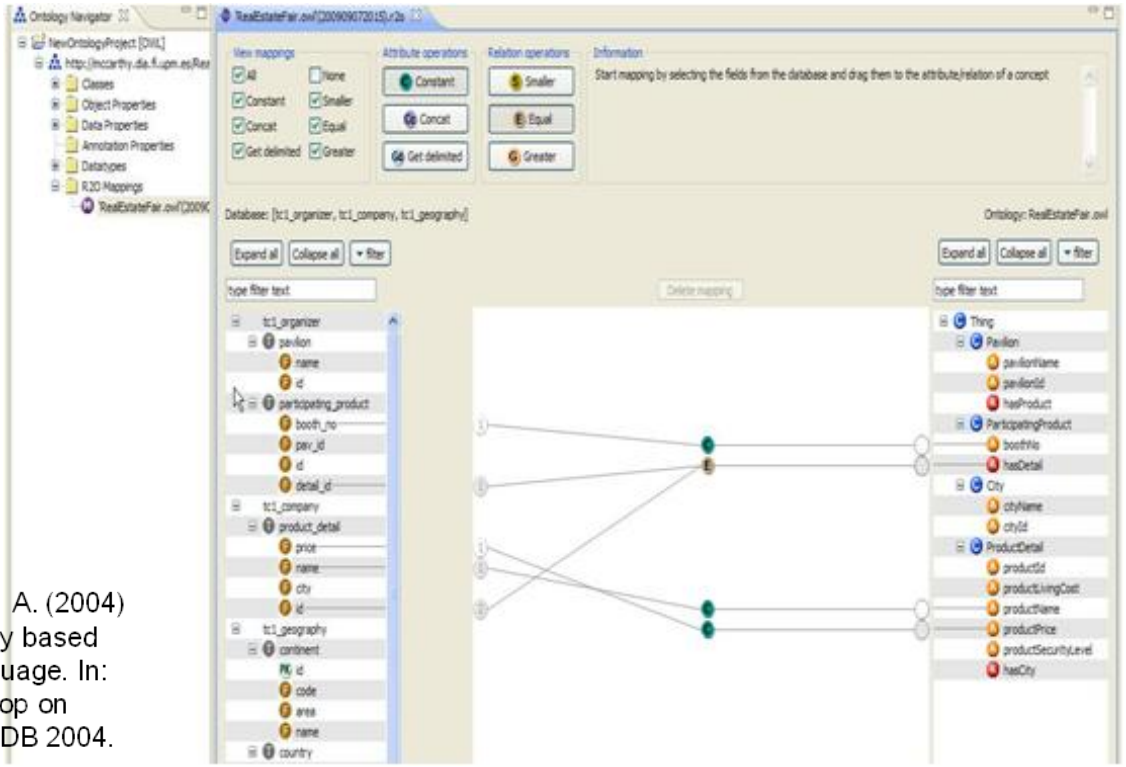


Outline


OEG - Data Integration
morph suite
morph-rdb in
clinical data

RDB2RDF:
Our old system

- R2O and ODEMapster
- NeON Toolkit plug-in
- Domains:
  - fund finding
  - cultural
  - fisheries



Barrasa J, Corcho O, Gómez-Pérez A. (2004) R2O, an extensible and semantically based database-to-ontology mapping language. In: Proceedings of the Second Workshop on Semantic Web and Databases, SWDB 2004.

RDB2RDF
Current days

	Before	**Current**
Language	R2O	R2RML
Engine	ODEMapster	morph-RDB
Focus	GUI	Optimisation in Query Translation
Goodies	NeOn Toolkit Plugin	morph-GFT morph-LDP

R2RML

www.w3.org/TR/r2rml/

W3C Recommendation

R2RML: RDB to RDF Mapping Language

W3C Recom

This version:
http://www.v

Latest version:
http://www.v

Previous versior
http://www.v

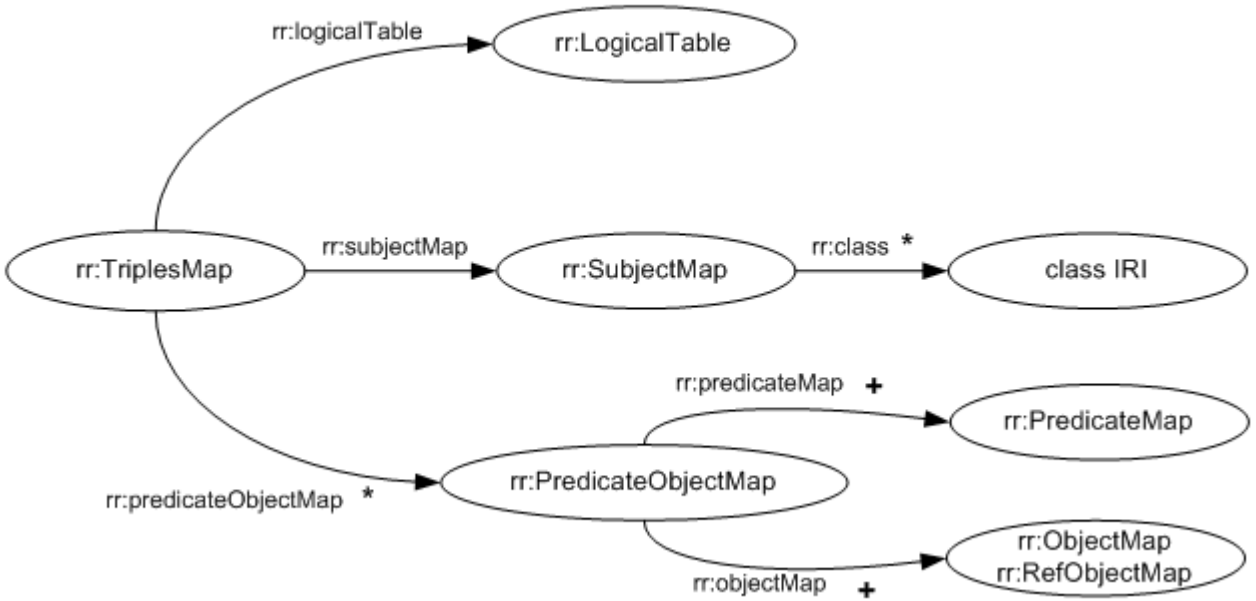
Editors:
Souripriya D
Seema Sun
Richard Cyg

Please refer to the

See also **translat**

Abstract

This document de
datasets. Such m
structure and targ
written down in T
offer a virtual SPA
interface.



Federated Query Processing

W3C Recommendation

W3C

SPARQL 1.1 Federated Query

W3C Recommendation 21 March 2013

This version:
http://www.w3.org/TR/2013/REC-sparql11-federated-query-20130321/
Latest version:
http://www.w3.org/TR/sparql11-federated-query/
Previous version:
http://www.w3.org/TR/2012/PR-sparql11-federated-query-20121108/
Editors:
Eric Prud'hommeaux, W3C <eric@w3.org>
Carlos Buil-Aranda, Ontology Engineering Group, UPM, Spain; currently at Universidad Pontificia Católica de Chile
Contributors:
Andy Seaborne, The Apache Software Foundation
Axel Polleres, Siemens AG <axel.polleres@siemens.com>
Lee Feigenbaum, Cambridge Semantics <lee@thefigtrees.net>
Gregory Todd Williams, Rensselaer Polytechnic Institute <greg@evilfunhouse.com>

Please refer to the **errata** for this document, which may include some normative corrections.

SPARQL-DQP

- Federated SPARQL Engine based on OGSA-DAI

Buil-Aranda, Carlos and Arenas, Marcelo and Corcho, Oscar. Semantics and optimization of the SPARQL 1.1 federation extension. The Semanic Web: Research and Applications. 2011

morph-GFT

- Accessing Google Fusion Tables (GFT) content via R2RML mappings, and integrating it with external information sources
  - morph-RDB (our R2RML engine)
  - SPARQL-DQP

Priyatna, Freddy and Aranda, Carlos Buil and Corcho, Oscar. Applying SPARQL-DQP for federated SPARQL querying over google fusion tables. ESWC 2013 Demo

morph-GFT

"Give me all the members of the Ontology Engineering Group coming from a country whose capital is Madrid"

```
SELECT ?n ?c
WHERE {
  SERVICE <http://mappingpedia.linkeddata.es/mappings/fusiontables/1pQBGUqR_g-j1WQavu-F1lwGS7jsdRxomGc0DxMI/oegmembers.ttl>
Q1{
    ?m rdf:type foaf:Person.
    ?m foaf:name ?n.
    ?m ex:hasCountry ?c.
  }
  SERVICE <http://DBpedia.org/sparql>
Q2{
    ?c dbpedia:property/capital dbpedia:resource/Madrid.
  }
}
```

```
SELECT name,
FROM 1pQBGUqR
WHERE name NC
```


(Carlos, Spain)


(Jean-Paul, Bolivia)


(Freddy, Indonesia)


(Oscar, Spain)

```
SELECT ?c
WHERE {
  ?c dbpedia:property/capital dbpedia:resource/Madrid.
}
```

Spain

Raul Garcia-Castro	Post-doc	Spain	

```
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant <http://dbpedia.org/ontology/Country> ];
  rr:objectMap    [ rr:template "http://dbpedia.org/resource/{Country}"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant ex:hasCountry ];
  rr:objectMap    [ rr:template "http://dbpedia.org/resource/{Country}"; ];
];
```


morph-LDP http://oeg-dev.dia.fi.upm.es/morph-ldp/

A marriage between:

- read-write morph-RDB
- LDP4j (our LDP implementation)

What it does:

1. Translate HTTP request into SPARQL
2. Translate SPARQL into SQL using R2RML mappings
3. Translate SQL Result into HTTP Response

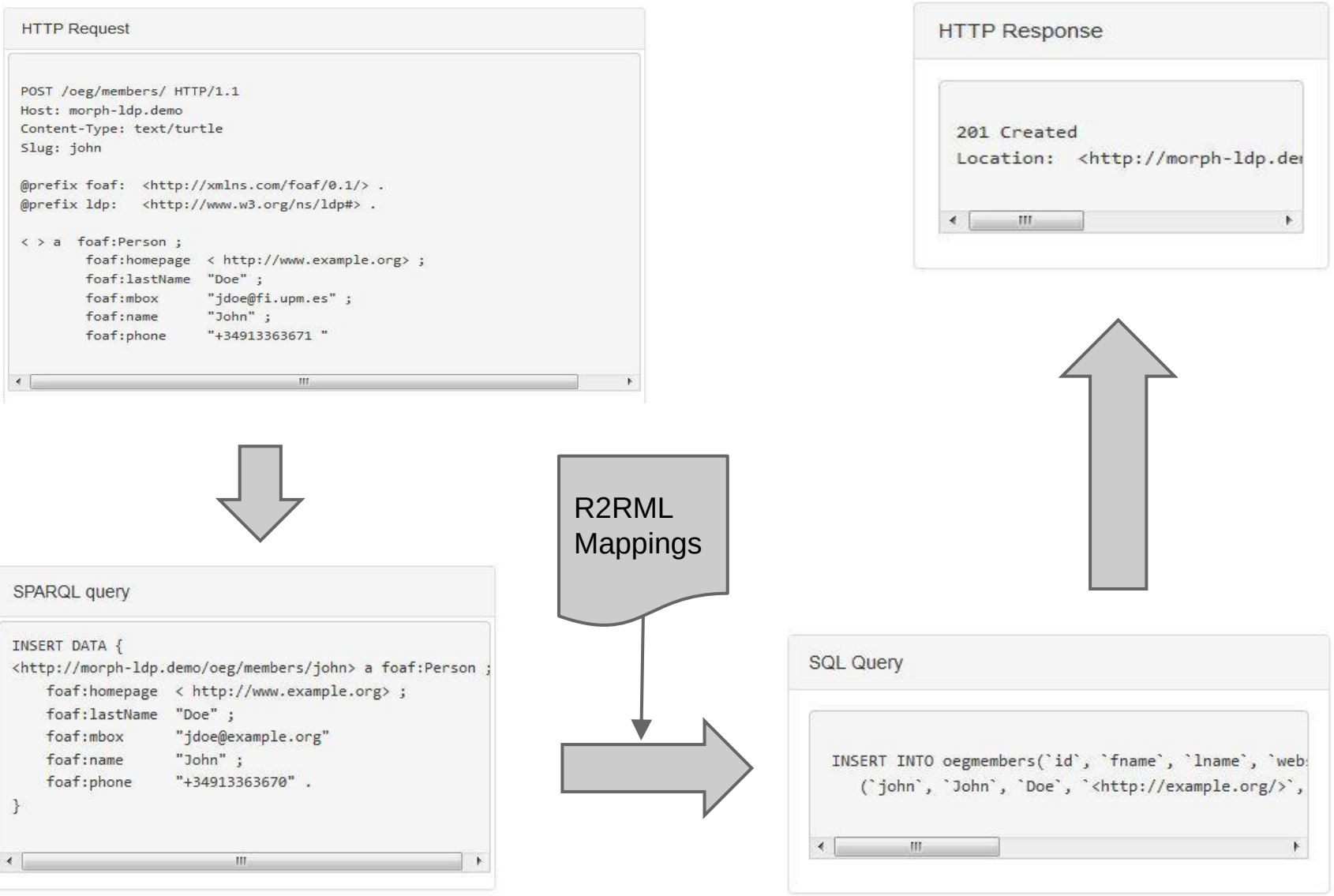
Mihindukulasooriya, Nandana and Priyatna, Freddy and Corcho, Oscar and Garcia-Castro, Raul and Esteban-Gutierrez, Miguel. morph-LDP: An R2RML-based Linked Data Platform implementation. ESWC 2014 Demo

morph-LDP
Motivations

As a Linked Data application developer, I want to:

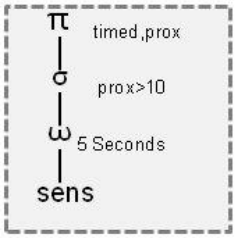
- retrieve the list of research group members
  - retrieve an LDP Container.
- retrieve details of a certain group member
  - retrieve an LDP Resource.
- update the details of a certain group member
  - update an LDP Resource.
- create a new member record of the group
  - create a new LDP Resource.

morph-LDP
Create Resource Example

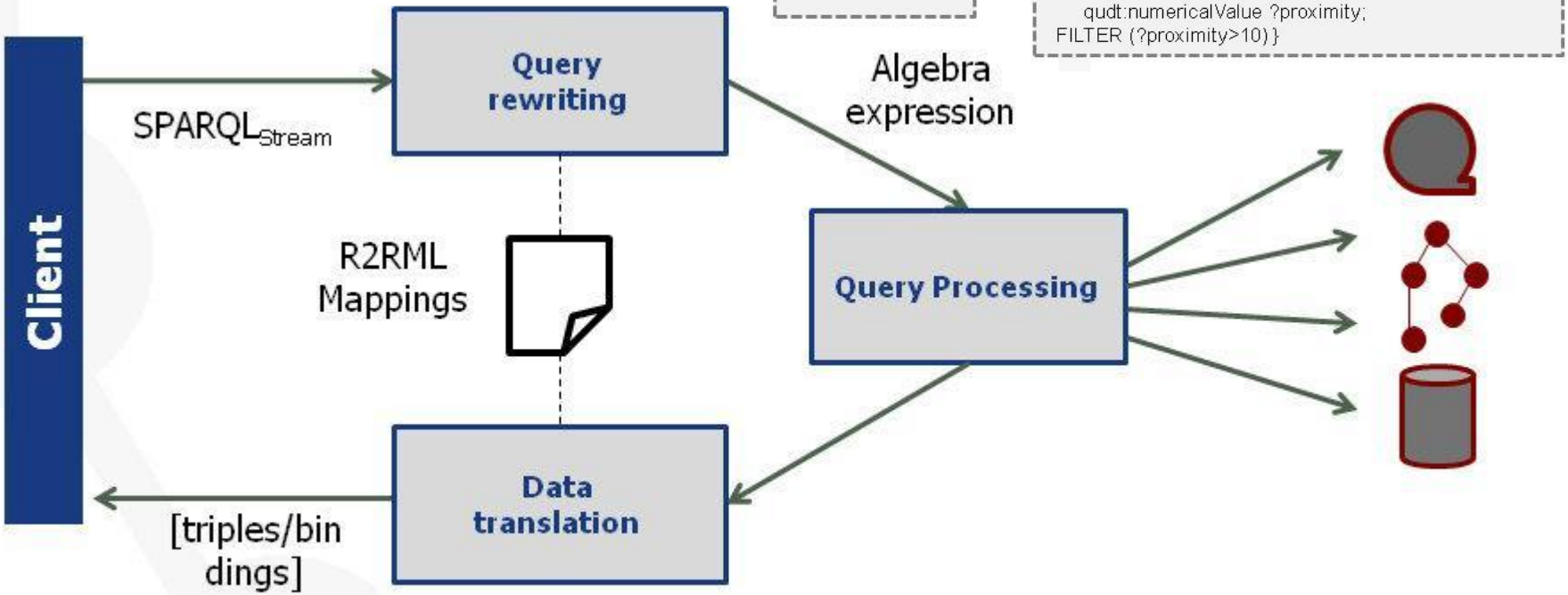


morph-stream

```
SELECT prox
FROM sens.win:time(5 sec)
WHERE prox >10
```




```
SELECT ?proximity
FROM STREAM
<http://streamreasoning.org/SensorReadings.srdf> [NOW–5 S]
WHERE {
 ?obs a ssn:ObservationValue;
      qudt:numericalValue ?proximity;
 FILTER (?proximity>10) }
```

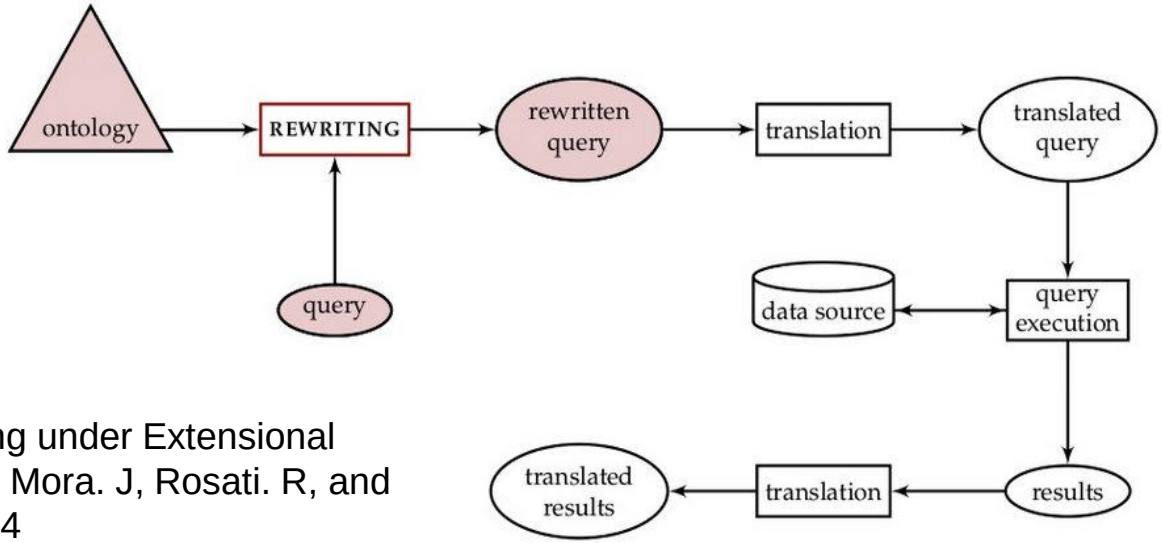


Morph-streams processing SPARQL$_{Stream}$ queries

Calbimonte, Jean-Paul and Corcho, Oscar and Gray, Alasdair JG.
Enabling ontology-based access to streaming data sources. ISWC 2010

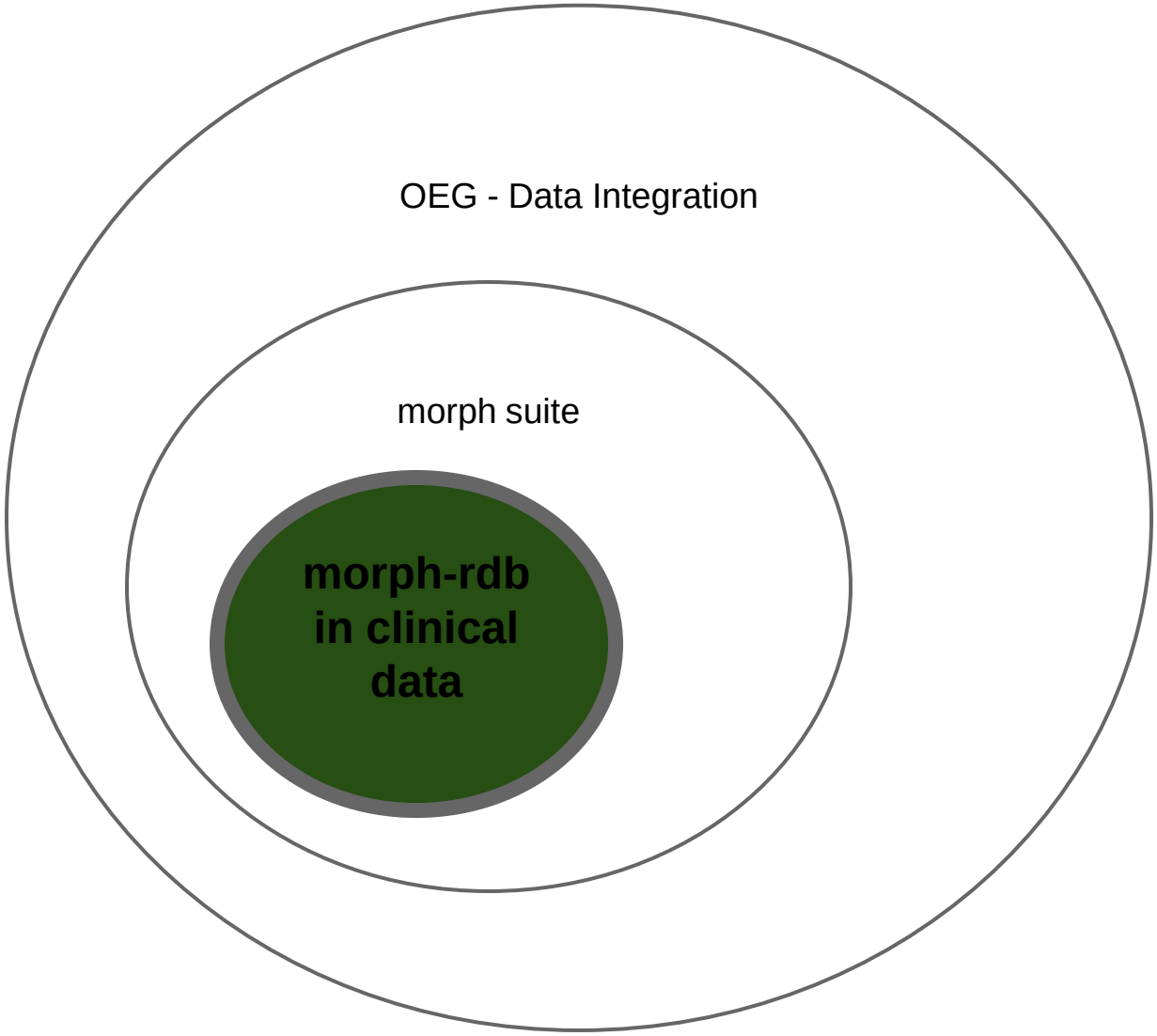
Reasoning Engine (kyrie)

- The TBox allows adding (intensional) facts to those in the ABox
- A reasoning engine allows doing this in query time, by extending the query, with no modification to the data sources and no materialization
- Query answering: Answers are extended with those that can be inferred (using the TBox) from data in the ABox (certain answers)
- Query rewriting: a form of query answering, produces a rewritten query to obtain all certain answers
- kyrie is a system that implements query rewriting for ELHIO TBoxes, including engineering optimisations



kyrie2: Query Rewriting under Extensional Constraints in ELHIO. Mora. J, Rosati. R, and Corcho. O. ISWC 2014

Outline


OEG - Data Integration
morph suite
morph-rdb
in clinical
data

morph-RDB

- Open source R2RML engine

$$trans(tp, \alpha, \beta) =$$

$$\texttt{Select Distinct } genPR\text{-}SQL(tp, \beta, name) \texttt{ From } \alpha(tp) \texttt{ Where } genCond\text{-}SQL(tp, \beta); \quad (13)$$

$$trans(gp_1\ AND\ gp_2, \alpha, \beta) =$$
$$\texttt{Select Distinct } name(a)_{[a|a\in(terms(gp_1)-terms(gp_2))]}\ name(b)_{[b|b\in(terms(gp_2)-terms(gp_1))]}$$
$$\texttt{Coalesce}(r_1.name(c), r_2.name(c))\ \texttt{As}\ name(c)_{[c|c\in(terms(gp_1)\cap terms(gp_2))]}$$
$$\texttt{From } (\ trans(gp_1, \alpha, \beta)\ )\ r_1\ \texttt{Inner Join } (\ trans(gp_2, \alpha, \beta)\ )\ r_2 \quad (14)$$
$$\texttt{On (True And}_{[c|c\in(terms(gp_1)\cap terms(gp_2))]}$$
$$(r_1.name(c)=r_2.name(c)\ \texttt{Or}\ r_1.name(c)\ \texttt{Is Null Or}\ r_2.name(c)\ \texttt{Is Null}));$$
where $r_1 = alias()$ and $r_2 = alias()$.

$$trans(gp_1\ OPT\ gp_2, \alpha, \beta) =$$
$$\texttt{Select Distinct } name(a)_{[a|a\in(terms(gp_1)-terms(gp_2))]}\ name(b)_{[b|b\in(terms(gp_2)-terms(gp_1))]}$$
$$\texttt{Coalesce}(r_1.name(c), r_2.name(c))\ \texttt{As}\ name(c)_{[c|c\in(terms(gp_1)\cap terms(gp_2))]}$$
$$\texttt{From } (\ trans(gp_1, \alpha, \beta)\ )\ r_1\ \texttt{Left Outer Join } (\ trans(gp_2, \alpha, \beta)\ )\ r_2 \quad (15)$$
$$\texttt{On (True And}_{[c|c\in(terms(gp_1)\cap terms(gp_2))]}$$
$$(r_1.name(c)=r_2.name(c)\ \texttt{Or}\ r_1.name(c)\ \texttt{Is Null Or}\ r_2.name(c)\ \texttt{Is Null}));$$
where $r_1 = alias()$ and $r_2 = alias()$.

$$trans(gp_1\ UNION\ gp_2, \alpha, \beta) =$$
$$\texttt{Select } name(a)_{[a|a\in A]},\ name(b)_{[b|b\in B]},\ r_1.name(c)_{[c|c\in C]}\ \texttt{As}\ name(c)$$
$$\texttt{From } (trans(gp_1, \alpha, \beta))\ r_1\ \texttt{Left Outer Join } (trans(gp_2, \alpha, \beta))\ r_2\ \texttt{On (False)} \quad (16)$$
$$\texttt{Union}$$
$$\texttt{Select } name(a)_{[a|a\in A]},\ name(b)_{[b|b\in B]},\ r_3.name(c)_{[c|c\in C]}\ \texttt{As}\ name(c)$$
$$\texttt{From } (trans(gp_2, \alpha, \beta))\ r_3\ \texttt{Left Outer Join } (trans(gp_1, \alpha, \beta))\ r_4\ \texttt{On (False)};$$
where r_1, r_2, r_3, and $r_4 = alias()$; A, B, and C are ordered sets $(terms(gp_1) - terms(gp_2))$, $(terms(gp_2) - terms(gp_1))$, and $(terms(gp_1) \cap terms(gp_2))$, respectively.

$$trans(gp\ FILTER\ expr, \alpha, \beta) =$$
$$\texttt{Select * From } (\ trans(gp, \alpha, \beta)\ )\ alias()\ \texttt{Where}\ transexpr(expr); \quad (17)$$

$$trans(SELECT\ (v_1, v_2, ..., v_n)\ WHERE(gp), \alpha, \beta) =$$
$$\texttt{Select Distinct } name(v_1), name(v_2), ..., name(v_n)\ \texttt{From } (\ trans(gp, \alpha, \beta)\ )\ alias(); \quad (18)$$

Priyatna. F, Corcho. O, Sequeda. J. Formalisation and Experiences of R2RML-based SPARQL to SQL Query Translation using Morph. WWW 2014

Clinical Data

- Data model (HL7 v3)
  - Relational schema implementation
  - Ontology implementation

Clinical Queries

- 4

q14.rq

q45.rq

```
# Obtener los pacientes a los que se les ha detectado un tumor de categoría T2 en el pecho

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX hl7rim: <http://hl7rim.GIB-UPM.org/common-data-model#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rr: <http://www.w3.org/ns/r2rml#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX rev: <http://purl.org/stuff/rev#>

SELECT DISTINCT ?id ?code ?patientId ?birthTime ?effectiveTime ?targetSiteCode
WHERE {
   ?instPerson          hl7rim:person_id ?patientId.
   ?instPerson          hl7rim:person_code '337915000'.
   OPTIONAL{   ?instPerson hl7rim:person_birthTime ?birthTime }
   ?instPerson          hl7rim:person_role ?instRole2.
   ?instRole2           hl7rim:role_entityId ?patientId.
   ?instRole2           hl7rim:role_participation ?instPart2.
   ?instPart2           hl7rim:participation_entityId ?patientId.
   ?instPart2           hl7rim:participation_act ?instAct.
   ?instAct          hl7rim:act_code ?code;
                     hl7rim:act_id  ?id.
   OPTIONAL {  ?instAct      hl7rim:act_effectiveTime ?effectiveTime}
   FILTER (?code IN ('67673008'))
   OPTIONAL{
      ?instAct          hl7rim:act_observationAct ?instObs.
      ?instObs                hl7rim:observationAct_actTargetSiteCode ?instTarget.
      OPTIONAL{
         ?instTarget          hl7rim:actTargetSiteCode_code ?targetSiteCode
         FILTER (?targetSiteCode IN ('76752008'))
      }
   }
}
```

Q01d

Q34 List all patients who were administered chemotherapy.

Q45 List patients who have been detected a category the

breast.

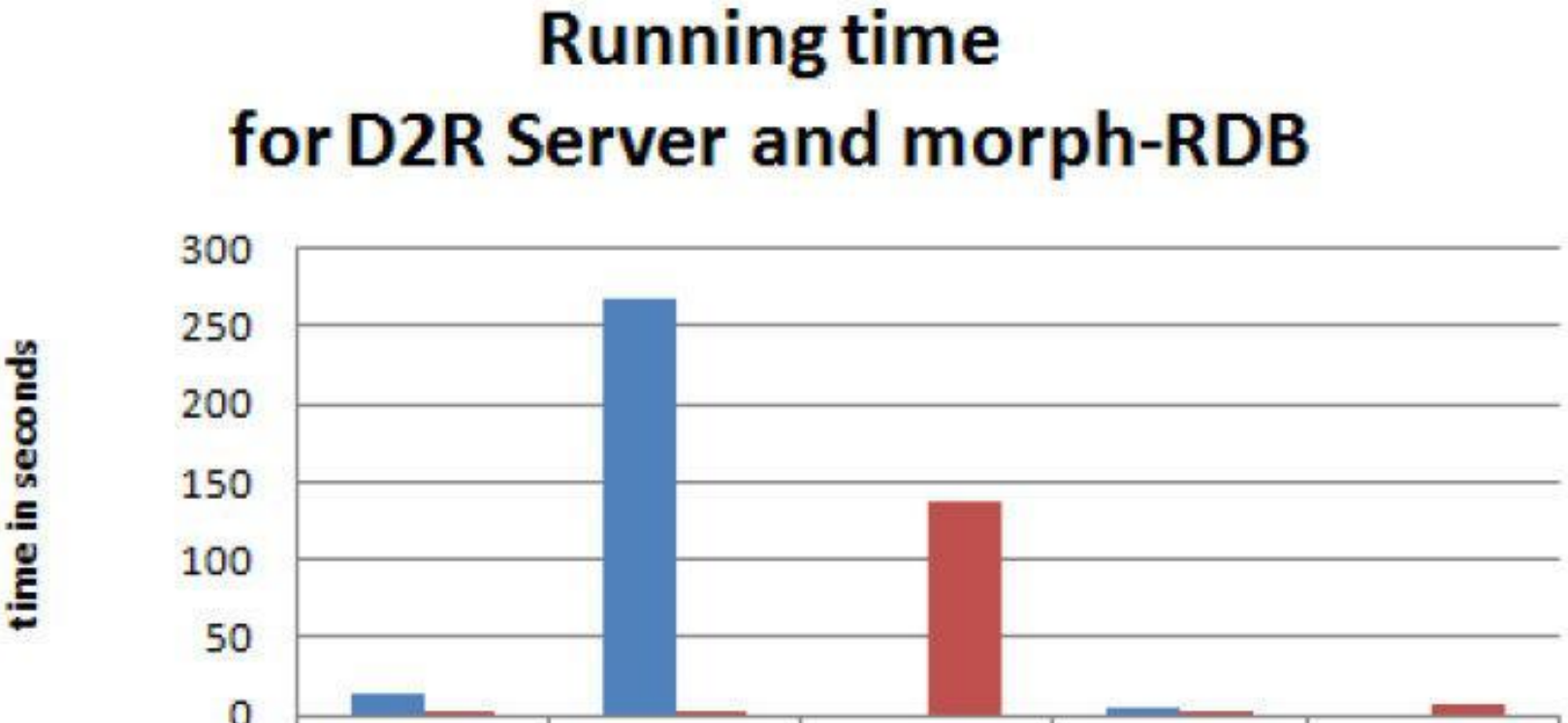
Table 3. Grouping of all queries a

Table 2. Summary of the properties of our five most representative queries

Our attempts

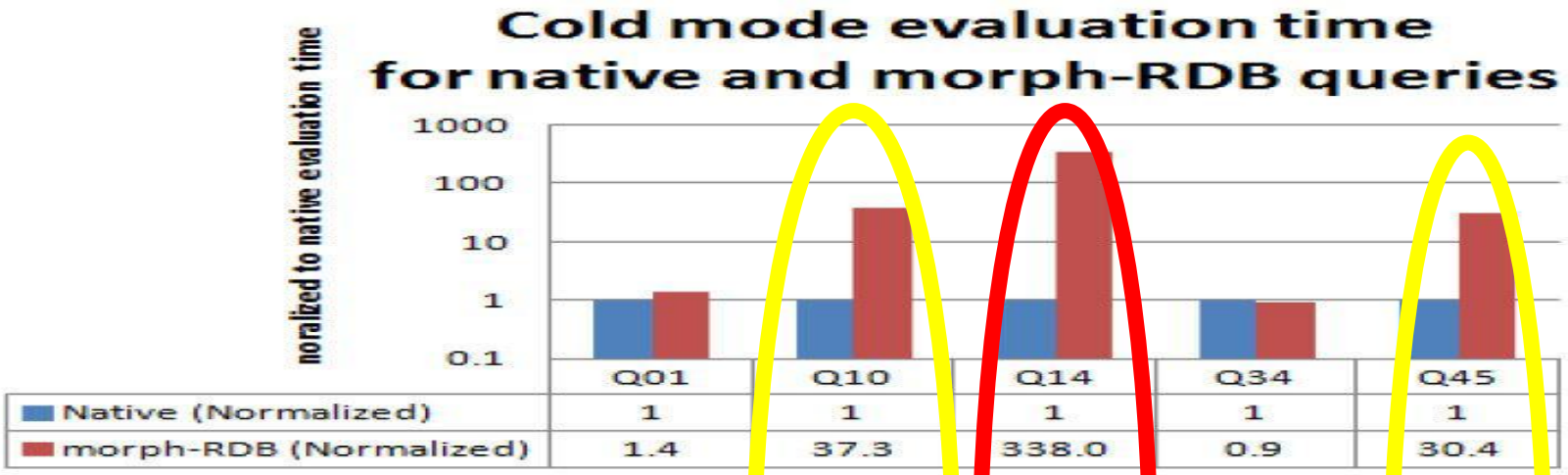
- First attempt
  - D2R + D2RQ
  - not applicable for various reasons
    - queries taking too long
    - too many joins
- Second attempt
  - R2RML + morph-RDB
  - 20 Triples Maps
    - 6 mapped to views
  - 364 Predicate Object Maps
    - 56 rr:refObjectMaps

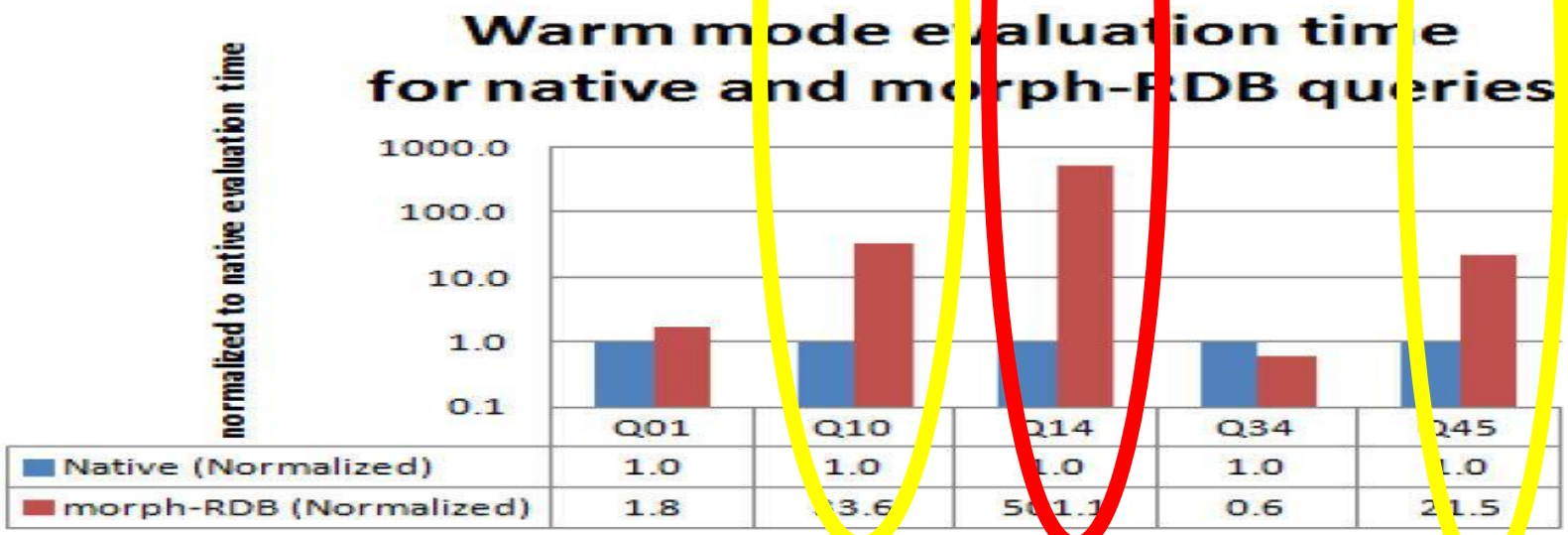
Evaluations



	Q01	Q10	Q14	Q34	Q45
D2R Server	13.766	268.092	0	4.308	0
morph-RDB	1.045	2.098	136.431	1.681	6.402

Evaluations



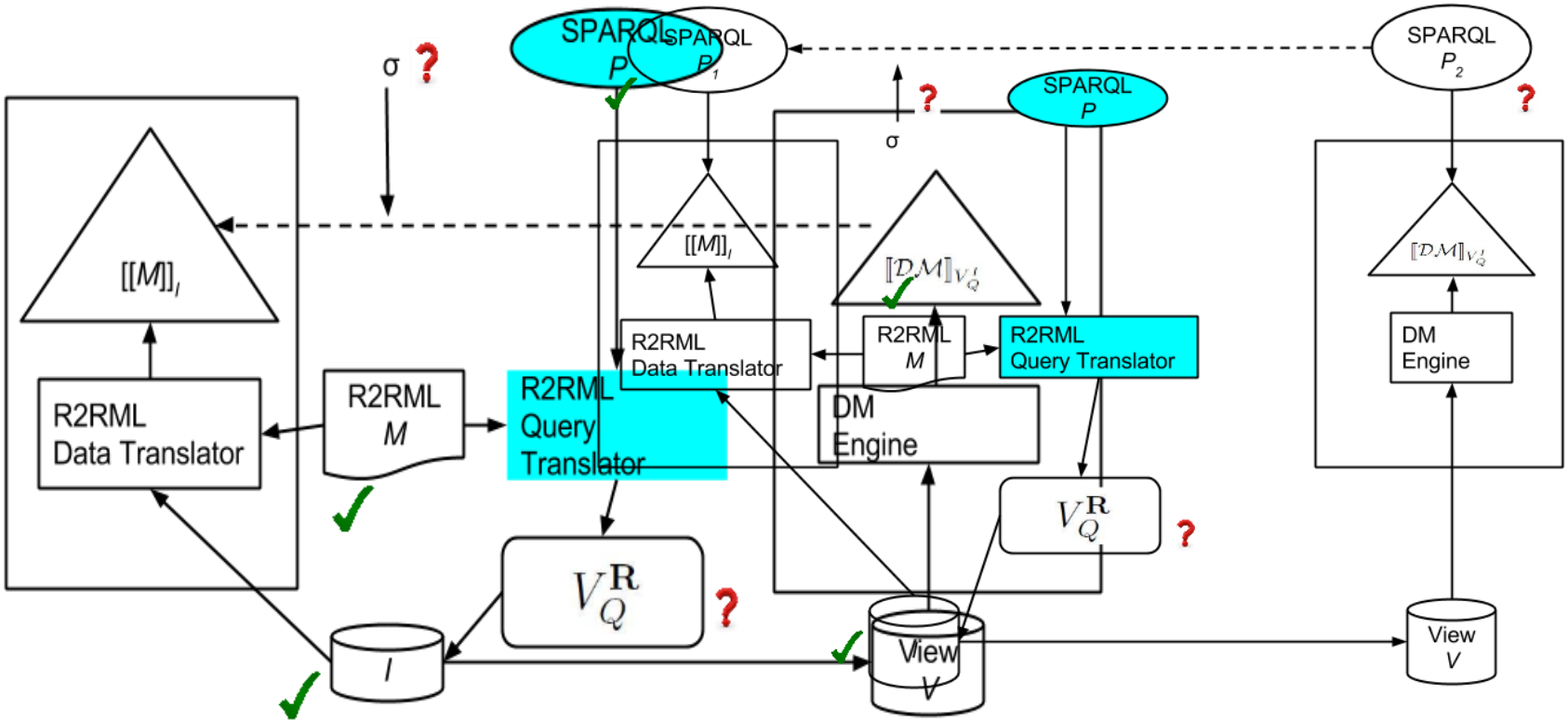


Conclusion

- We have seen the overview of work done in OEG's Data Integration group
  - Possibility/call for collaboration
- morph-RDB makes it possible to run clinical queries
  - Some still need additional work (Q14)
- We have also applied morph-RDB in other real-world domains

Ongoing/Future Work (1)
R2RML <-> Direct Mapping

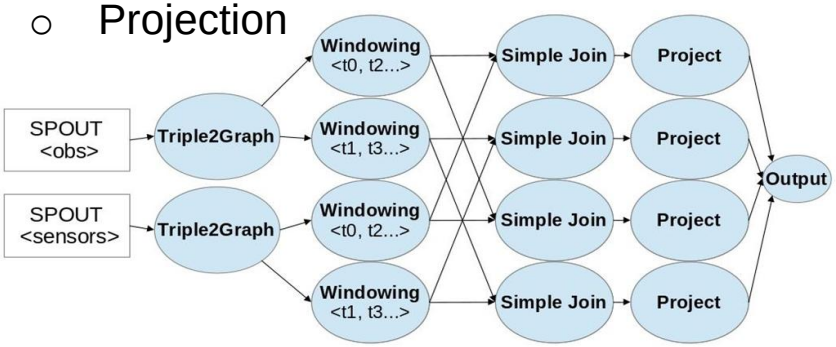
- Identifying the essential fragment of R2RML mapping language
- Studying the expressive power of Direct Mapping and its relationship with R2RML



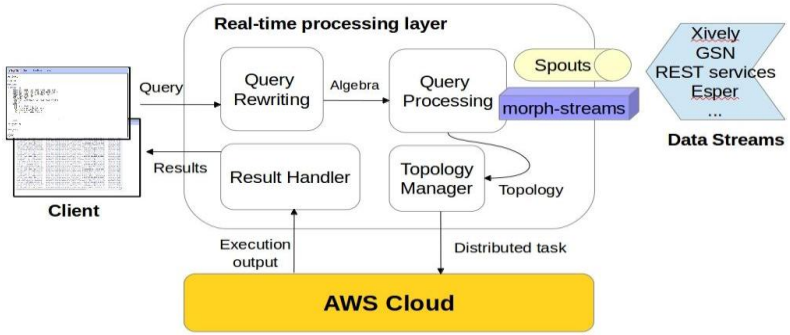
Ongoing/Future Work (2) morph-stream++

Towards a scalable RDF stream processing engine

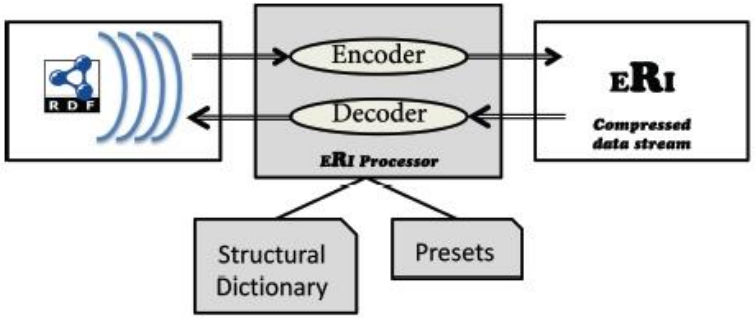
- Parallelization: Storm query operators
  - Triple2Graph
  - Time-windowing
  - Simple Join
  - Projection



- Modularity: distributed real-time layer



- Data compression: Efficient RDF Interchange (ERI)
  - Based on Efficient XML Interchange (EXI)
  - Main assumption: RDF streams have regular structure and are redundant
  - ERI processing model



  - Information encoded at two levels
    - Structural dictionary
    - Presets (redundant values)