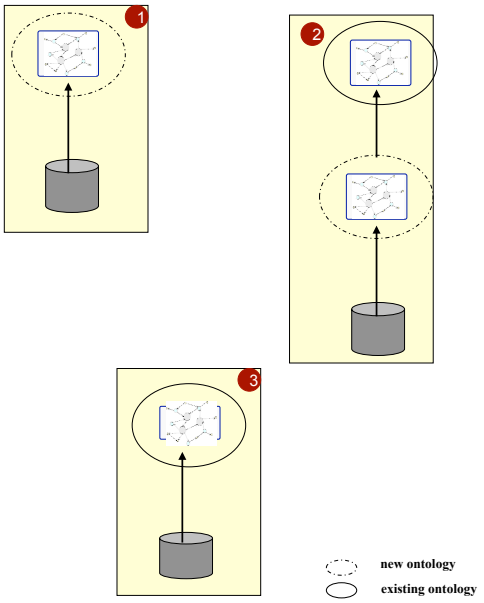
Publishing Linked Data from RDB

Boris Villazón-Terrazas. Oscar Corcho
Facultad de Informática, Universidad Politécnica de Madrid
Campus de Montegancedo sn, 28660 Boadilla del Monte, Madrid
http://www.oeg-upm.net
ocorcho@fi.upm.es
Phone: 34.91.3366605, Fax: 34.91.3524819

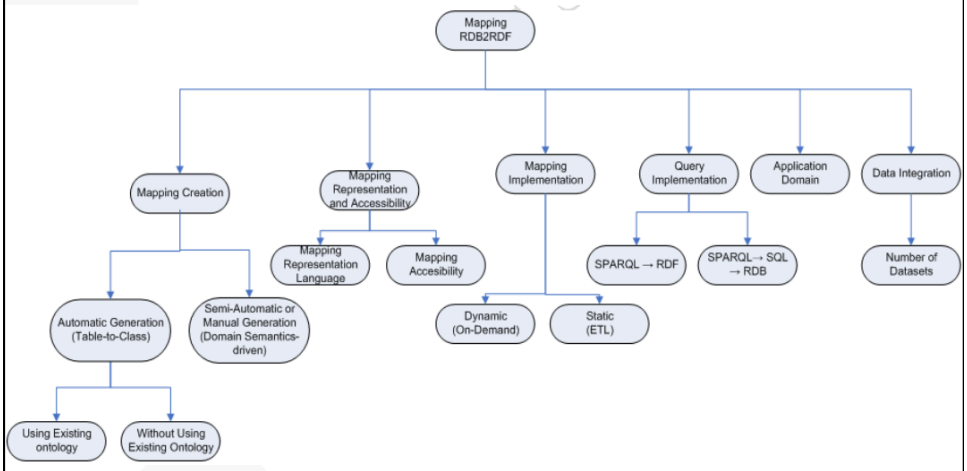
Acknowledgements: Freddy Priyatna, Jan Schulte, Richard Cyganiak and many others that we may have omitted.

RDB2RDF. Existing approaches

1. To build a new ontology from a database schema and content (**direct mappings**)
2. To map the ontology created in approach (1) to a legacy ontology
3. To map an existing DB to a legacy ontology



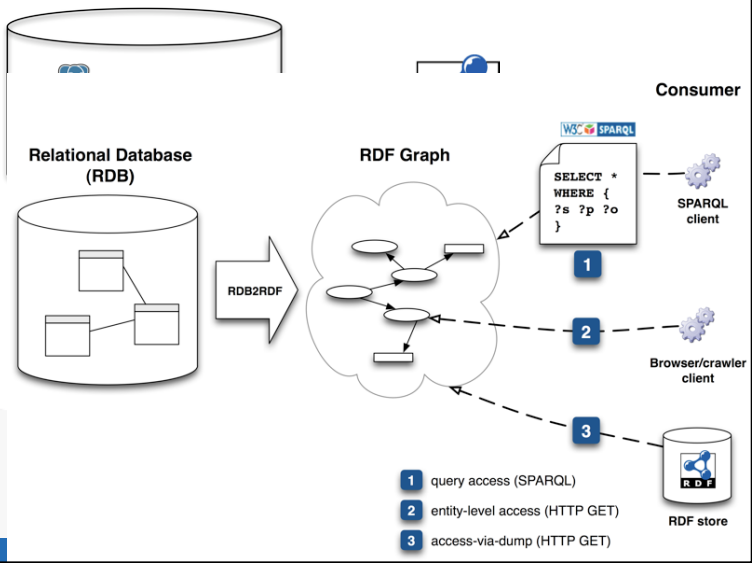
A Survey of Current Approaches for Mapping of Relational Databases to RDF



A Survey of Current Approaches for Mapping of Relational Databases to RDF, Satya S. Sahoo, Wolfgang Halb, Sebastian Hellmann, Kingsley Idehen, Ted Thibodeau Jr, Sören Auer, Juan Sequeda, Ahmed Ezzat, 2009. (See http://www.w3.org/2005/Incubator/rdb2rdf/RDB2RDF_SurveyReport.pdf.)

Transformation – RDB2RDF

- A majority of dynamic Web content is backed by relational databases (RDB), and so are many enterprise systems.



Direct Mapping

- The direct mapping defines an RDF Graph representation of the data in an RDB. The direct mapping takes as input an RDB (data and schema), and generates an RDF graph that is called the direct graph.

```
CREATE TABLE "Addresses" (
        "ID" INT, PRIMARY KEY("ID"),
        "city" CHAR(10),
        "state" CHAR(2)
)

CREATE TABLE "People" (
        "ID" INT, PRIMARY KEY("ID"),
        "fname" CHAR(10),
        "addr" INT,
        FOREIGN KEY("addr") REFERENCES "Addresses"("ID")
)

INSERT INTO "Addresses" ("ID", "city", "state") VALUES (18, 'Cambridge', 'MA')
INSERT INTO "People" ("ID", "fname", "addr") VALUES (7, 'Bob', 18)
INSERT INTO "People" ("ID", "fname", "addr") VALUES (8, 'Sue', NULL)
```


Direct Mapping - example

People

PK		*→ Address(ID)*
ID	**fname**	**addr**
7	Bob	18
8	Sue	**NULL**

Addresses

PK		
ID	**city**	**state**
18	Cambridge	MA

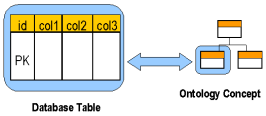
```
@base <http://foo.example/DB/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .


<People/ID=7> rdf:type <People> .
<People/ID=7> <People#ID> 7 .
<People/ID=7> <People#fname> "Bob" .
<People/ID=7> <People#addr> 18 .
<People/ID=7> <People#ref-addr> <Addresses/ID=18> .
<People/ID=8> rdf:type <People> .
<People/ID=8> <People#ID> 8 .
<People/ID=8> <People#fname> "Sue" .

<Addresses/ID=18> rdf:type <Addresses> .
<Addresses/ID=18> <Addresses#ID> 18 .
<Addresses/ID=18> <Addresses#city> "Cambridge" .
<Addresses/ID=18> <Addresses#state> "MA" .
```

The W3C RDB2RDF Working Group

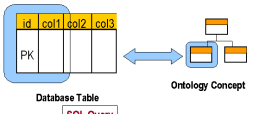
- Created in 2007
- W3C Recommendations in September 2012
  - R2RML: RDB to RDF Mapping Language - http://www.w3.org/TR/r2rml/
  - Direct Mapping - http://www.w3.org/TR/rdb-direct-mapping/
  - R2RML and Direct Mapping Test Cases - http://www.w3.org/2001/sw/rdb2rdf/test-cases/
  - RDB2RDF Implementation Report - http://www.w3.org/2001/sw/rdb2rdf/implementation-report/
- R2RML mappings are themselves expressed as RDF graphs and written down in Turtle syntax.

R2O (Relational-to-Ontology) Language

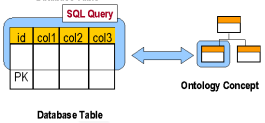
For concepts...



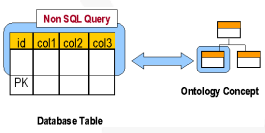
A view maps exactly one concept in the ontology.



A subset of the columns in the view map a concept in the ontology.



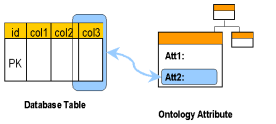
A subset (selection) of the records of a database view map a concept in the ontology.



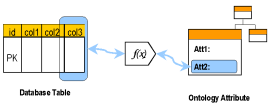
A subset of the records of a database view map a concept in the onto. but the selection cannot be made using SQL.

One or more concepts can be extracted from a single data field (not in 1NF).

For attributes...



A column in a database view maps directly an attribute or a relation.

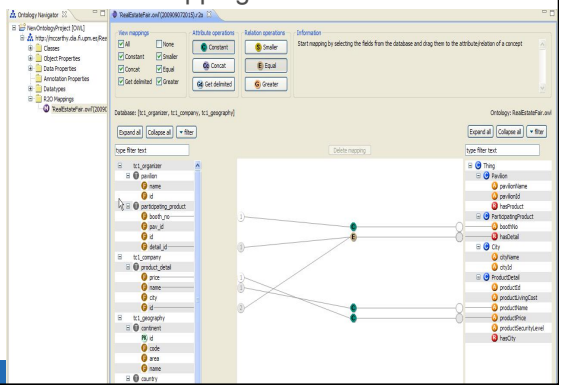


A column in a database view maps an attribute or a relation after some transformation.

A set of columns in a database view map an attribute or a relation.

OEG's background knowledge in RDB2RDF

- R2O and ODEMapster
  - GaV wrapper generation (no mediators)
    - Syntactic sugar for the generation of SQL queries.
  - Simple use of this language and processor in the domains of fund finding, cultural information, and fisheries.
  - NeOn Toolkit plugin for common mappings



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.

R2RML - example I

```
CREATE TABLE "Student" (
"ID" integer,
"Name" varchar(50),
PRIMARY KEY ("ID")
);
INSERT INTO "Student" ("ID", "Name") VALUES(10,'Venus');
```

Student

ID (PK) INTEGER	Name VARCHAR(50)
10	Venus

```
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix ex: <http://example.com/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@base <http://example.com/base/> .

<TriplesMap1>
    a rr:TriplesMap;

    rr:logicalTable [ rr:tableName "\"Student\"" ];

    rr:subjectMap [ rr:template "http://example.com/Student/{\"ID\"}/{\"Name\"}"; ];

    rr:predicateObjectMap
    [
        rr:predicate    rdf:type;
        rr:object       foaf:Person;
    ];
    .
```

Subject	Predicate	Object	Graph
<http://example.com/Student/10/Venus>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://xmlns.com/foaf/0.1/Person>	

R2RML – example II - input

```
CREATE TABLE "Student" (
"ID" integer,
"Name" varchar(50),
PRIMARY KEY ("ID")
);
INSERT INTO "Student" ("ID", "Name") VALUES(10,'Venus');
```

Student

ID (PK) INTEGER	Name VARCHAR(50)
10	Venus

```
<TriplesMap1>
    a rr:TriplesMap;

   rr:logicalTable [ rr:tableName "\"Student\""; ];

   rr:subjectMap [ rr:template "http://example.com/Student/{\"ID\"}/{\"Name\"}";
                   rr:graph ex:PersonGraph;
   ];

   rr:predicateObjectMap
   [
       rr:predicate    rdf:type;
       rr:object       foaf:Person;
   ];

   rr:predicateObjectMap
   [
     rr:predicate      foaf:name;
     rr:objectMap      [ rr:column "\"Name\"" ]
   ]
```

Subject	Predicate	Object	Graph
<http://example.com/Student/10/Venus>	<http://xmlns.com/foaf/0.1/name>	"Venus"	<http://example.com/PersonGraph>
<http://example.com/Student/10/Venus>	<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://xmlns.com/foaf/0.1/Person>	<http://example.com/PersonGraph>

R2RML - example III - input

Student

ID (PK) INTEGER	FirstName VARCHAR(50)	LastName VARCHAR(50)
10	Venus	Williams
11	Fernando	Alonso
12	David	Villa

Student_Sport

ID_Student (PFK) INTEGER	ID_Sport (PFK) INTEGER
10	110
11	111
11	112
12	111

Sport

ID (PK) INTEGER	Description VARCHAR(50)
110	Tennis
111	Football
112	Formula1

```
CREATE TABLE "Student" (
"ID" integer PRIMARY KEY,
"FirstName" varchar(50),
"LastName" varchar(50)
);
CREATE TABLE "Sport" (
"ID" integer PRIMARY KEY,
"Description" varchar(50)
);
CREATE TABLE "Student_Sport" (
"ID_Student" integer,
"ID_Sport" integer,
PRIMARY KEY ("ID_Student","ID_Sport"),
FOREIGN KEY ("ID_Student") REFERENCES "Student"("ID"),
FOREIGN KEY ("ID_Sport") REFERENCES "Sport"("ID")
);

INSERT INTO "Student" ("ID","FirstName","LastName") VALUES (10,'Venus', 'Williams');
INSERT INTO "Student" ("ID","FirstName","LastName") VALUES (11,'Fernando', 'Alonso');
INSERT INTO "Student" ("ID","FirstName","LastName") VALUES (12,'David', 'Villa');

INSERT INTO "Sport" ("ID", "Description") VALUES (110,'Tennis');
INSERT INTO "Sport" ("ID", "Description") VALUES (111,'Football');
INSERT INTO "Sport" ("ID", "Description") VALUES (112,'Formula1');

INSERT INTO "Student_Sport" ("ID_Student", "ID_Sport") VALUES (10,110);
INSERT INTO "Student_Sport" ("ID_Student", "ID_Sport") VALUES (11,111);
INSERT INTO "Student_Sport" ("ID_Student", "ID_Sport") VALUES (11,112);
INSERT INTO "Student_Sport" ("ID_Student", "ID_Sport") VALUES (12,111);
```

R2RML - example III - mapping

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix ex: <http://example.com/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@base <http://example.com/base/> .

<TriplesMap1>
    a rr:TriplesMap;

    rr:logicalTable [ rr:tableName "\"Student\""; ] ;

    rr:subjectMap [ rr:template "http://example.com/student/{\"ID\"}"; ];

    rr:predicateObjectMap
    [
        rr:predicate    ex:firstName ;
        rr:objectMap    [ rr:column "\"FirstName\"" ]
    ];

    rr:predicateObjectMap
    [
        rr:predicate    ex:lastName ;
        rr:objectMap    [ rr:column "\"LastName\"" ]
    ]
```


R2RML - example III - mapping (cont.)

```
<TriplesMap2>
    a rr:TriplesMap;

    rr:logicalTable [ rr:tableName "\"Sport\""; ] ;

    rr:subjectMap [ rr:template "http://example.com/sport/{\"ID\"}"; ];

    rr:predicateObjectMap
    [
        rr:predicate    ex:id ;
        rr:objectMap    [ rr:column "\"ID\""; ]
    ];

    rr:predicateObjectMap
    [
        rr:predicate    ex:description ;
        rr:objectMap    [ rr:column "\"Description\"" ]
    ]
    .

<LinkMap_1_2>
    a rr:TriplesMap;

   rr:logicalTable [ rr:tableName "\"Student_Sport\"" ];

   rr:subjectMap [ rr:template "http://example.com/student/{\"ID_Student\"}" ];

   rr:predicateObjectMap [
        rr:predicate    ex:plays ;
        rr:objectMap    [ rr:template "http://example.com/sport/{\"ID_Sport\"}" ];
   ].
```

R2RML - example III - output

Subject	Predicate	Object	Graph
<http://example.com/student/10>	<http://example.com/lastName>	"Williams"	
<http://example.com/student/10>	<http://example.com/firstName>	"Venus"	
<http://example.com/student/12>	<http://example.com/lastName>	"Villa"	
<http://example.com/student/12>	<http://example.com/firstName>	"David"	
<http://example.com/student/11>	<http://example.com/lastName>	"Alonso"	
<http://example.com/student/11>	<http://example.com/firstName>	"Fernando"	
<http://example.com/sport/110>	<http://example.com/description>	"Tennis"	
<http://example.com/sport/110>	<http://example.com/id>	"110"^^<http://www.w3.org/2001/XMLSchema#integer>	
<http://example.com/sport/111>	<http://example.com/description>	"Football"	
<http://example.com/sport/111>	<http://example.com/id>	"111"^^<http://www.w3.org/2001/XMLSchema#integer>	
<http://example.com/sport/112>	<http://example.com/description>	"Formula1"	
<http://example.com/sport/112>	<http://example.com/id>	"112"^^<http://www.w3.org/2001/XMLSchema#integer>	
<http://example.com/student/10>	<http://example.com/plays>	<http://example.com/sport/110>	
<http://example.com/student/12>	<http://example.com/plays>	<http://example.com/sport/111>	
<http://example.com/student/11>	<http://example.com/plays>	<http://example.com/sport/112>	
<http://example.com/student/11>	<http://example.com/plays>	<http://example.com/sport/111>	

R2RML - example IV - input

Student

ID (PK) INTEGER	FirstName VARCHAR(50)	LastName VARCHAR(50)
10	Venus	Williams
11	Fernando	Alonso
12	David	Villa

Student_Sport

ID_Student (PFK) INTEGER	ID_Sport (PFK) INTEGER
10	110
11	111
11	112
12	111

Sport

ID (PK) INTEGER	Description VARCHAR(50)
110	Tennis
111	Football
112	Formula1

```
CREATE TABLE "Student" (
"ID" integer PRIMARY KEY,
"FirstName" varchar(50),
"LastName" varchar(50)
);
CREATE TABLE "Sport" (
"ID" integer PRIMARY KEY,
"Description" varchar(50)
);
CREATE TABLE "Student_Sport" (
"ID_Student" integer,
"ID_Sport" integer,
PRIMARY KEY ("ID_Student","ID_Sport"),
FOREIGN KEY ("ID_Student") REFERENCES "Student"("ID"),
FOREIGN KEY ("ID_Sport") REFERENCES "Sport"("ID")
);

INSERT INTO "Student" ("ID","FirstName","LastName") VALUES (10,'Venus', 'Williams');
INSERT INTO "Student" ("ID","FirstName","LastName") VALUES (11,'Fernando', 'Alonso');
INSERT INTO "Student" ("ID","FirstName","LastName") VALUES (12,'David', 'Villa');

INSERT INTO "Sport" ("ID", "Description") VALUES (110,'Tennis');
INSERT INTO "Sport" ("ID", "Description") VALUES (111,'Football');
INSERT INTO "Sport" ("ID", "Description") VALUES (112,'Formula1');

INSERT INTO "Student_Sport" ("ID_Student", "ID_Sport") VALUES (10,110);
INSERT INTO "Student_Sport" ("ID_Student", "ID_Sport") VALUES (11,111);
INSERT INTO "Student_Sport" ("ID_Student", "ID_Sport") VALUES (11,112);
INSERT INTO "Student_Sport" ("ID_Student", "ID_Sport") VALUES (12,111);
```

R2RML - example IV - mapping

```
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix ex: <http://example.com/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@base <http://example.com/base/> .

<TriplesMap1>
    a rr:TriplesMap;
    rr:logicalTable [ rr:sqlQuery """
       SELECT "Student"."ID" as ID,
              "Student"."FirstName" as FirstName,
              "Student"."LastName" as LastName,
              "Sport"."Description" as Description,
              "Sport"."ID" as Sport_ID
       FROM "Student","Sport","Student_Sport"
       WHERE "Student"."ID" = "Student_Sport"."ID_Student"
       AND "Sport"."ID" = "Student_Sport"."ID_Sport";
       """; ];
    rr:subjectMap [ rr:template "http://example.com/{ID}/{FirstName};{LastName}" ];
    rr:predicateObjectMap
    [
      rr:predicate  ex:id ;
      rr:objectMap  [ rr:column "ID"; ] ];
    rr:predicateObjectMap
    [
      rr:predicate  ex:firstName ;
      rr:objectMap  [ rr:column "FirstName" ] ];
    rr:predicateObjectMap
    [
      rr:predicate  ex:lastName ;
      rr:objectMap  [ rr:column "LastName" ] ];
    rr:predicateObjectMap
    [
      rr:predicate  ex:plays ;
      rr:objectMap  [ rr:template "http://example.com/{Sport_ID}/{Description}" ] ]
    .
```


R2RML - example IV - mapping (cont.)

```
<TriplesMap2>
    a rr:TriplesMap;

    rr:logicalTable [ rr:tableName  "\"Sport\"" ];

    rr:subjectMap [ rr:template "http://example.com/{\"ID\"}/{\"Description\"}"; ];

    rr:predicateObjectMap
    [
      rr:predicate  ex:id ;
      rr:objectMap  [ rr:column "\"ID\""; ]
    ];

    rr:predicateObjectMap
    [
      rr:predicate  ex:description ;
      rr:objectMap  [ rr:column "\"Description\"" ]
    ]
    .
```

R2RML - example IV - output

Subject	Predicate	Object	Graph
<http://example.com/110/Tennis>	<http://example.com/description>	"Tennis"	
<http://example.com/110/Tennis>	<http://example.com/id>	"110"^^<http://www.w3.org/2001/XMLSchema#integer>	
<http://example.com/10/Venus;Williams>	<http://example.com/plays>	<http://example.com/110/Tennis>	
<http://example.com/10/Venus;Williams>	<http://example.com/lastName>	"Williams"	
<http://example.com/10/Venus;Williams>	<http://example.com/firstName>	"Venus"	
<http://example.com/10/Venus;Williams>	<http://example.com/id>	"10"^^<http://www.w3.org/2001/XMLSchema#integer>	
<http://example.com/111/Football>	<http://example.com/description>	"Football"	
<http://example.com/111/Football>	<http://example.com/id>	"111"^^<http://www.w3.org/2001/XMLSchema#integer>	
<http://example.com/12/David;Villa>	<http://example.com/plays>	<http://example.com/111/Football>	
<http://example.com/12/David;Villa>	<http://example.com/lastName>	"Villa"	
<http://example.com/12/David;Villa>	<http://example.com/firstName>	"David"	
<http://example.com/12/David;Villa>	<http://example.com/id>	"12"^^<http://www.w3.org/2001/XMLSchema#integer>	
<http://example.com/112/Formula1>	<http://example.com/description>	"Formula1"	
<http://example.com/112/Formula1>	<http://example.com/id>	"112"^^<http://www.w3.org/2001/XMLSchema#integer>	
<http://example.com/11/Fernando;Alonso>	<http://example.com/lastName>	"Alonso"	
<http://example.com/11/Fernando;Alonso>	<http://example.com/firstName>	"Fernando"	
<http://example.com/11/Fernando;Alonso>	<http://example.com/id>	"11"^^<http://www.w3.org/2001/XMLSchema#integer>	
<http://example.com/11/Fernando;Alonso>	<http://example.com/plays>	<http://example.com/111/Football>	
<http://example.com/11/Fernando;Alonso>	<http://example.com/plays>	<http://example.com/112/Formula1>	

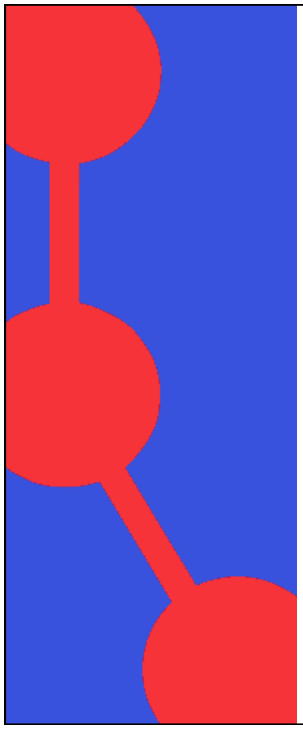
R2RML processors

- http://code.google.com/p/oeg-obdi/
- https://github.com/jpcik/morph
- https://github.com/boricles/morph

Table 2.: R2RML Processors.

Name	Contact	Home
OpenLink Virtuoso	Ivan Mikhailov	http://virtuoso.openlinksw.com
RDF-RDB2RDF	Toby Inkster	https://metacpan.org/release/RDF-RDB2RDF
XSPARQL	Nuno Lopes	http://xsparql.deri.org
morph	Jean-Paul Calbimonte	https://github.com/jpcik/morph
ultrawrap	Juan Sequeda	http://www.capsenta.com/
db2triples	Julien Homo and Laurent Mazuel	https://github.com/antidot/db2triples

RDB2RDF Implementation Report. Boris Villazón-Terrazas, Michael Hausenblas.
http://www.w3.org/2001/sw/rdb2rdf/implementation-report/



Publishing Linked Data from RDB

Boris Villazón-Terrazas. Oscar Corcho
Facultad de Informática, Universidad Politécnica de Madrid
Campus de Montegancedo sn, 28660 Boadilla del Monte, Madrid
http://www.oeg-upm.net
ocorcho@fi.upm.es
Phone: 34.91.3366605, Fax: 34.91.3524819

Acknowledgements: Freddy Priyatna, Jan Schulte, Richard Cyganiak and many others that we may have omitted.