CC7220-1
La Web de Datos
Primavera 2019

Lecture 10: RDB2RDF

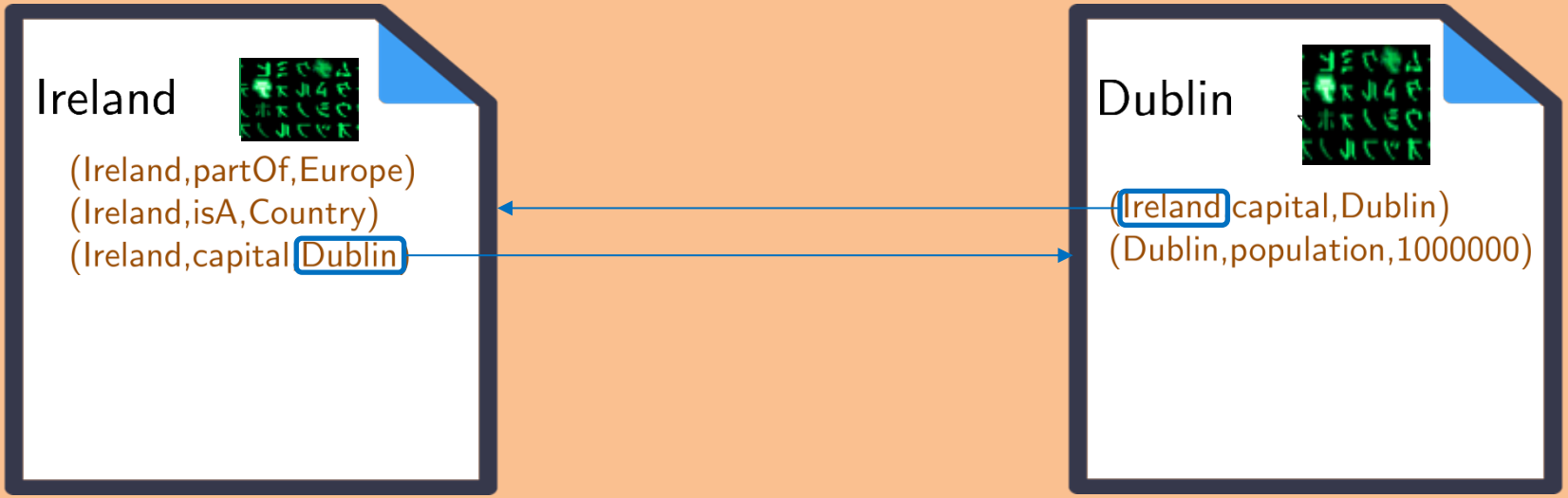
Aidan Hogan
aidhog@gmail.com

Previously …

* More or less

Semantic Web: Data, Logic, Query, Links

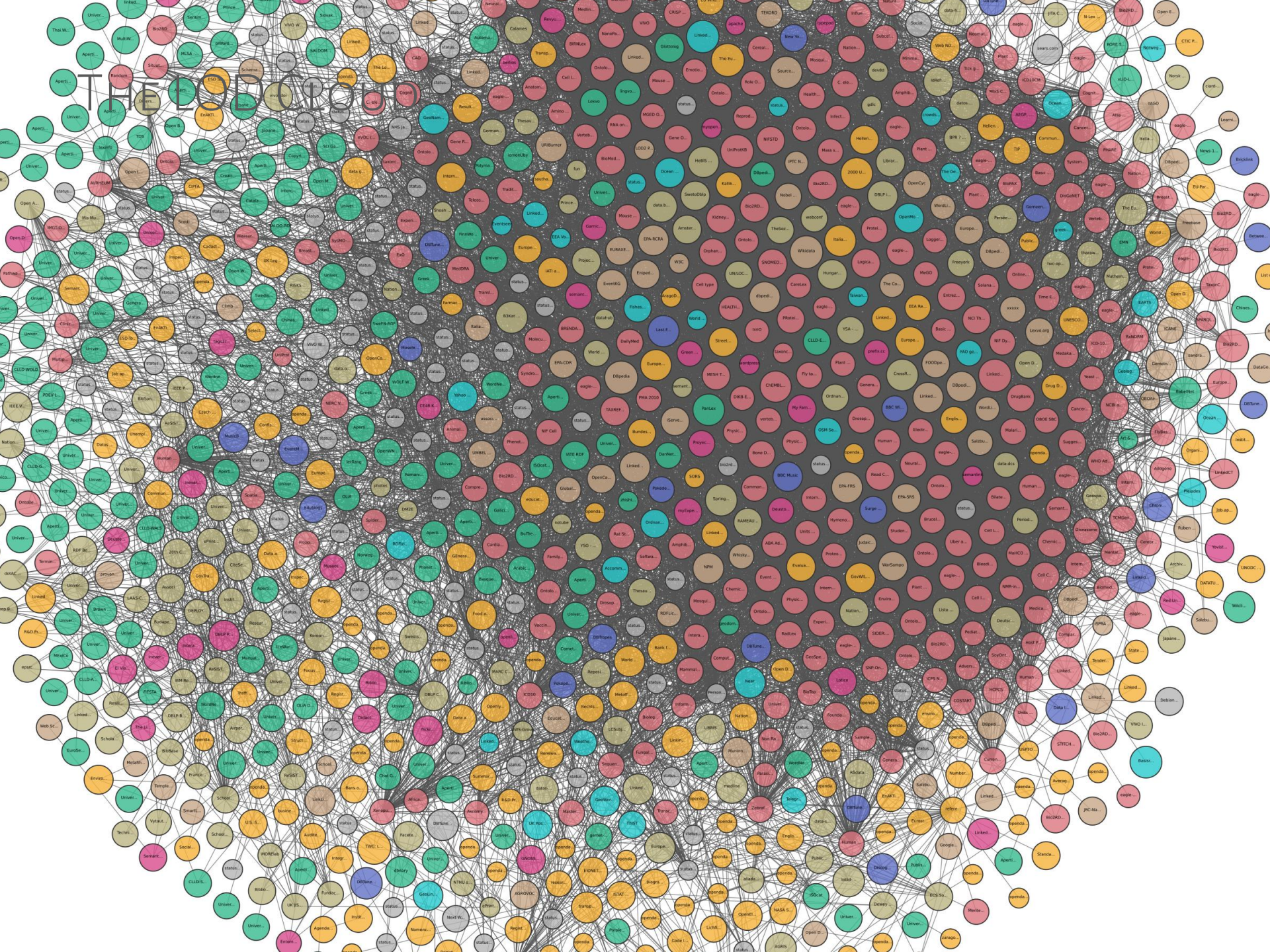
Data:



Logic: "$(b$,capital,$a) \rightarrow (a$,partOf,$b)$"
"$(a$,partOf,$b)$, $(b$,partOf,$c) \rightarrow (a$,partOf,$c)$"

Query: "$(x$,partOf,$y)$?"

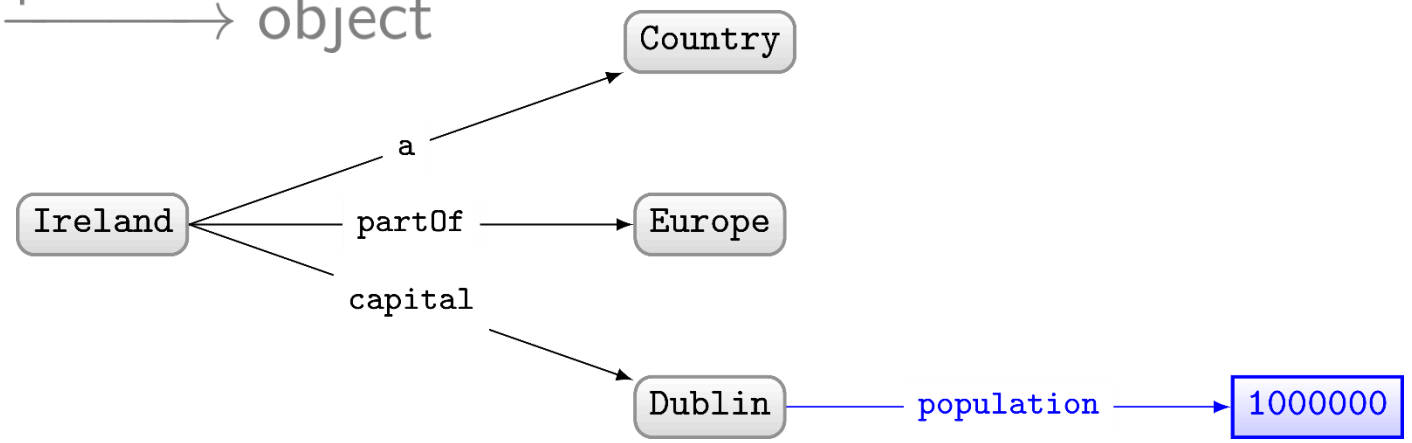
Output: $\{(x \mapsto$ Ireland, $y \mapsto$ Europe),
$(x \mapsto$ Dublin, $y \mapsto$ Ireland),
$(x \mapsto$ Dublin, $y \mapsto$ Europe$)\}$


THE DOD CLOUD

RDF: Proposed model for a Web of Data

subject	*predicate*	*object*
Ireland	partOf	Europe
Ireland	a	Country
Ireland	capital	Dublin
Dublin	population	1,000,000

subject —predicate→ object

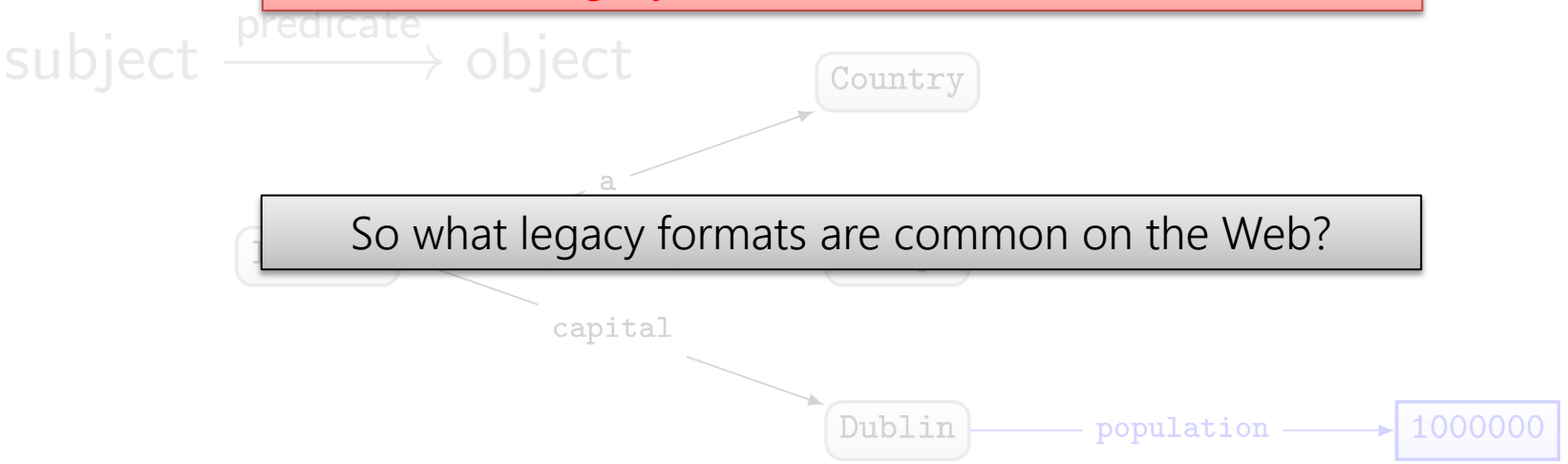


RDF: Proposed model for a Web of Data

subject	predicate	object
Ireland	partOf	Europe
Ireland	capital	Dublin
Dublin	population	1,000,000

But where should this RDF come from?

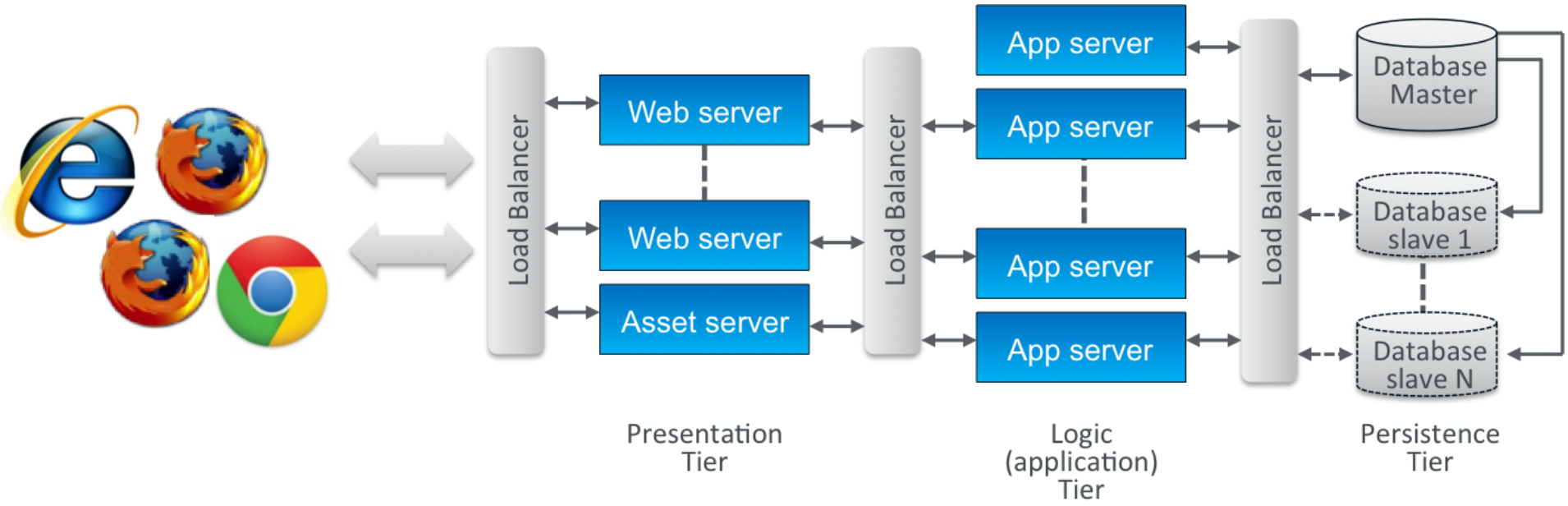
Most legacy data on the Web not in RDF!



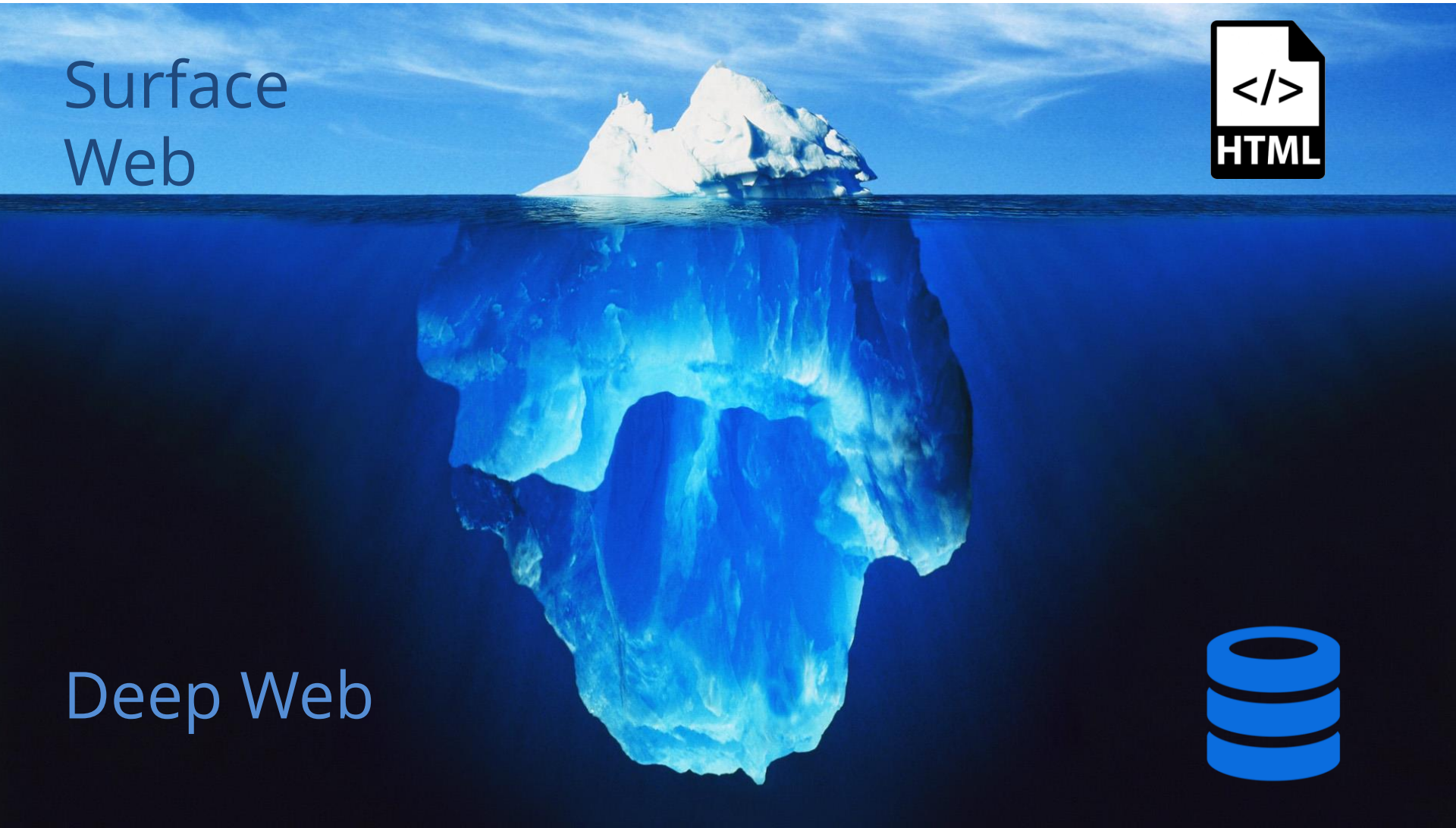
So what legacy formats are common on the Web?

LAMP: Linux, Apache, MySQL, PHP

3-Tier Web Applications



Deep Web



Lots of data in relational databases …

Ranking > Complete Ranking

RSS RSS Feed

DB-Engines Ranking

The DB-Engines Ranking ranks database management systems according to their popularity. The ranking is updated monthly.

Read more about the method of calculating the scores.

trend chart

348 systems in ranking, November 2018

Rank Nov 2018	Rank Oct 2018	Rank Nov 2017	DBMS	Database Model	Score Nov 2018	Score Oct 2018	Score Nov 2017
1.	1.	1.	Oracle	Relational DBMS	1301.11	-18.16	-58.94
2.	2.	2.	MySQL	Relational DBMS	1159.89	-18.22	-162.14
3.	3.	3.	Microsoft SQL Server	Relational DBMS	1051.55	-6.78	-163.53
4.	4.	4.	PostgreSQL	Relational DBMS	440.24	+20.85	+60.33
5.	5.	5.	MongoDB	Document store	369.48	+6.30	+39.01
6.	6.	6.	IBM Db2	Relational DBMS	179.87	+0.19	-14.19
7.	7.	↑ 9.	Redis	Key-value store	144.17	-1.12	+22.99
8.	8.	↑ 10.	Elasticsearch	Search engine	143.46	+1.13	+24.05
9.	9.	↓ 7.	Microsoft Access	Relational DBMS	138.44	+1.64	+5.12
10.	↑ 11.	↑ 11.	SQLite	Relational DBMS	122.71	+5.96	+9.95

http://db-engines.com/en/ranking

Lots of data in relational databases …

Ranking > Complete Ranking

RSS Feed

DB-Engines Ranking

The DB-Engines Ranking ranks database management systems according to their popularity. The ranking is updated monthly.

Read more about the method of calculating the scores.

trend chart

Out of top 10 databases, 7 are relational databases.

November 2018

Rank Nov 2018	Rank Oct 2018	Rank Nov 2017	DBMS	Database Model	Score Nov 2018	Score Oct 2018	Score Nov 2017
1.	1.	1.	Oracle	Relational DBMS	1301.11	-18.16	-58.94
2.	2.	2.	MySQL	Relational DBMS	1159.89	-18.22	-162.14
3.	3.	3.	Microsoft SQL Server	Relational DBMS	1051.55	-6.78	-163.53
4.	4.	4.	PostgreSQL	Relational DBMS	440.24	+20.85	+60.33
5.	5.	5.	MongoDB	Document store	369.48	+6.30	+39.01
6.	6.					+0.19	-14.19
7.	7.					-1.12	+22.99
8.	8.					+1.13	+24.05
9.	9.	7.	Microsoft Access	Relational DBMS	136.44	+1.64	+5.12
10.	↑ 11.	↑ 11.	SQLite	Relational DBMS	122.71	+5.96	+9.95

Out of 348 databases, any idea in what position the first SPARQL engine would be?

http://db-engines.com/en/ranking

Top SPARQL engine is …

81.	79.	73.	Apache Drill	Multi-model	2.85	-0.07	-0.07
82.	84.	75.	Graphite	Time Series DBMS	2.85	+0.04	-0.01
83.	74.	86.	Amazon CloudSearch	Search engine	2.75	-0.33	+0.39
84.	83.	74.	PouchDB	Document store	2.75	-0.06	-0.16
85.	85.	69.	RRDtool	Time Series DBMS	2.73	+0.05	-0.47
86.	95.	84.	TimesTen	Relational DBMS	2.62	+0.35	+0.04
87.	86.	95.	LevelDB	Key-value store	2.59	+0.01	+0.57
88.	88.	88.	Apache Jena - TDB	RDF store	2.45	-0.05	+0.16
89.	92.	81.	SAP Advantage Database Server	Relational DBMS	2.40	+0.07	-0.27
90.	89.	90.	OmniSci	Relational DBMS	2.40	-0.04	+0.17
91.	97.	97.	Virtuoso	Multi-model	2.37	+0.18	+0.49
92.	87.	87.	Infinispan	Key-value store	2.36	-0.20	+0.04
93.	90						
94.	94						
95.	91						
96.	93						
97.	96						
98.	100						
99.	99						
100.	104						

RDB2RDF:
Relational DataBases to RDF

Some relational tables about planets ...

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

Moon

name	pname	discoverer	year
Luna	Earth	$\perp$	$\perp$
Ganimedes	Jupiter	Galileo Galilei	1610
Calisto	Jupiter	Galileo Galilei	1610
Europa	Jupiter	Galileo Galilei	1610
Io	Jupiter	Galileo Galilei	1610
Titan	Saturn	Christiaan Huygens	1655
Triton	Neptune	William Lassell	1846

Landing

ship	pname	country	year
Messenger	Mercury	US	2015
Venera 3	Venus	USSR	1966
Pioneer	Venus	US	1978
Mars 2 lander	Ma	USSR	1971
Viking 1	Mars	US	1976
Beagle 2	Mars	EU	2003
Galileo	Jupiter	US	2003

Meanwhile on Pluto ...

RDB2RDF?

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

How might we automatically convert this table to RDF?

Direct Mapping: Automatic RDB2RDF mapping

W3C Recommendation

A Direct Mapping of Relational Data to RDF

W3C Recommendation 27 September 2012

This version:
http://www.w3.org/TR/2012/REC-rdb-direct-mapping-20120927/
Latest version:
http://www.w3.org/TR/rdb-direct-mapping/
Previous version:
http://www.w3.org/TR/2012/PR-rdb-direct-mapping-20120814/
Editors:
Marcelo Arenas, Pontificia Universidad Católica de Chile <marenas@ing.puc.cl>
Alexandre Bertails, W3C <bertails@w3.org>
Eric Prud'hommeaux, W3C <eric@w3.org>
Juan Sequeda, University of Texas at Austin <jsequeda@cs.utexas.edu>

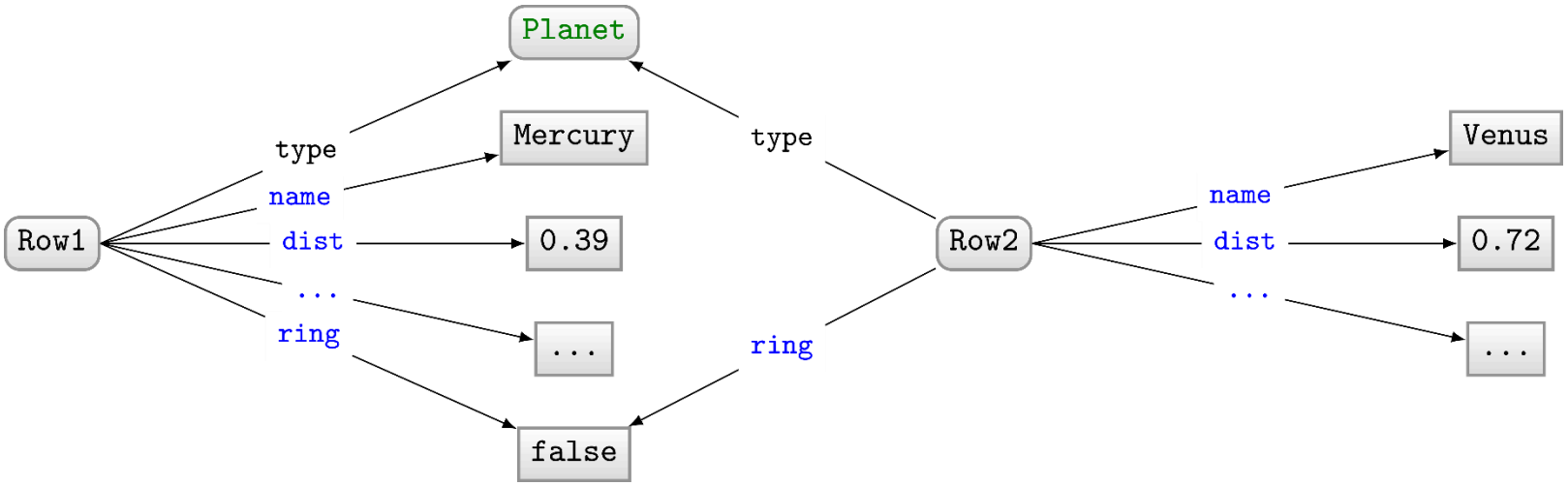
Please refer to the **errata** for this document, which may include some normative corrections.

See also **translations**.

Direct Mapping

Planet

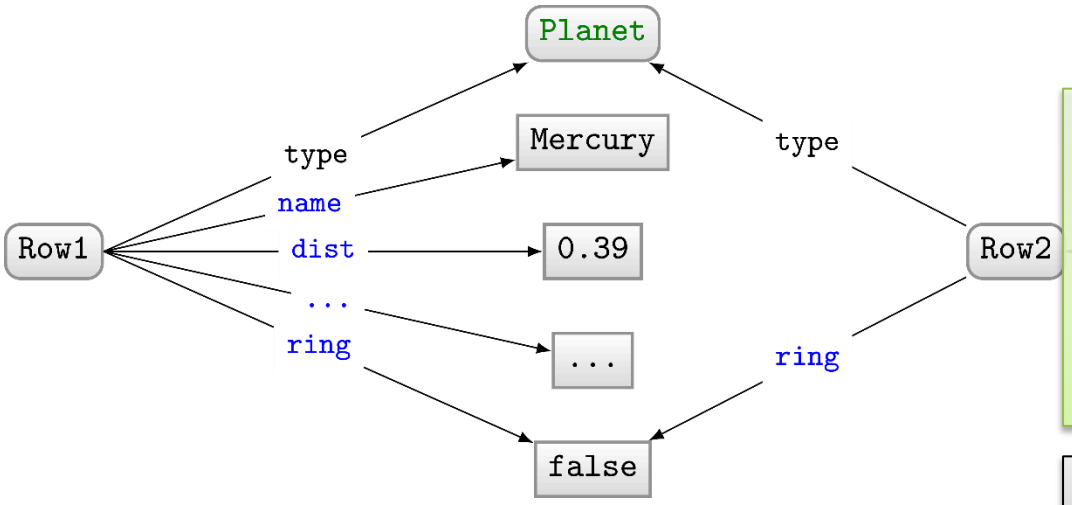
name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true



Direct Mapping

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true



Each table name is a type

Each row is a subject

Each attribute a predicate

Each value an object

But what about RDF terms (IRIs/literals/blank nodes)?

Direct Mapping: Identifying Rows

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

Row1

How can we identify Row1?

- If the table has a primary key ($\underline{pk_1}$,…,$\underline{pk_n}$):
  - `http://ex.org/TableName/pk1=v1;...;pkn=vn`
  - (Base IRI `http://ex.org/` given as input)
- If not: use a fresh blank node.

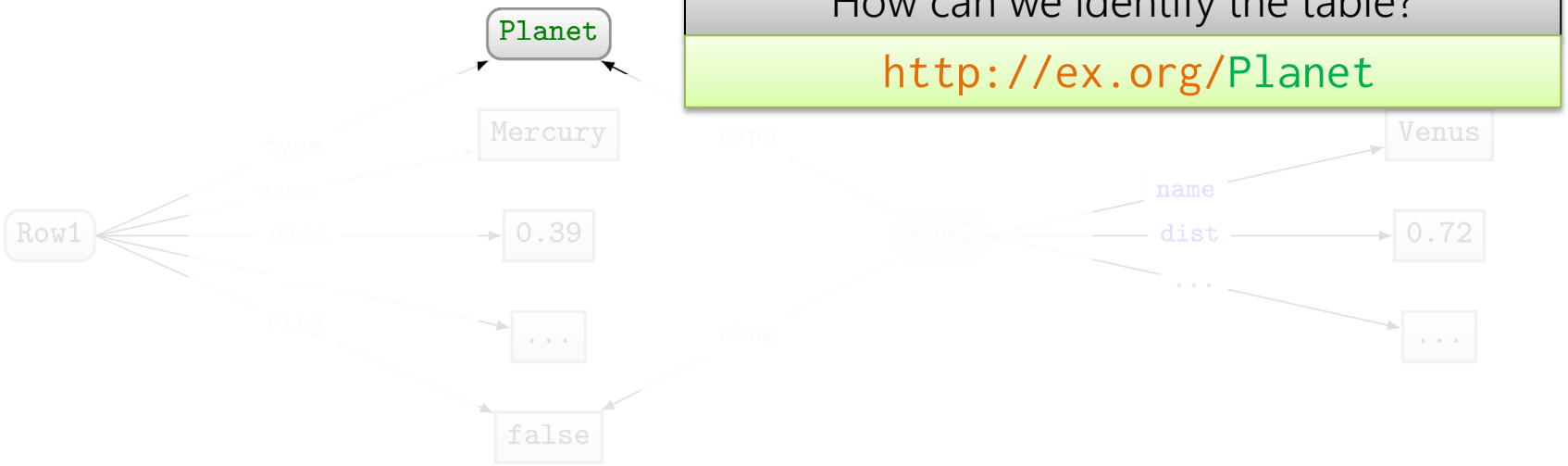
So Row1 would be …?

`http://ex.org/Planet/name=Mercury`

Direct Mapping: Identifying Tables

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true



How can we identify the table?

http://ex.org/Planet

Direct Mapping: Identifying Columns

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

How can we identify the attributes?

http://ex.org/Planet#ring

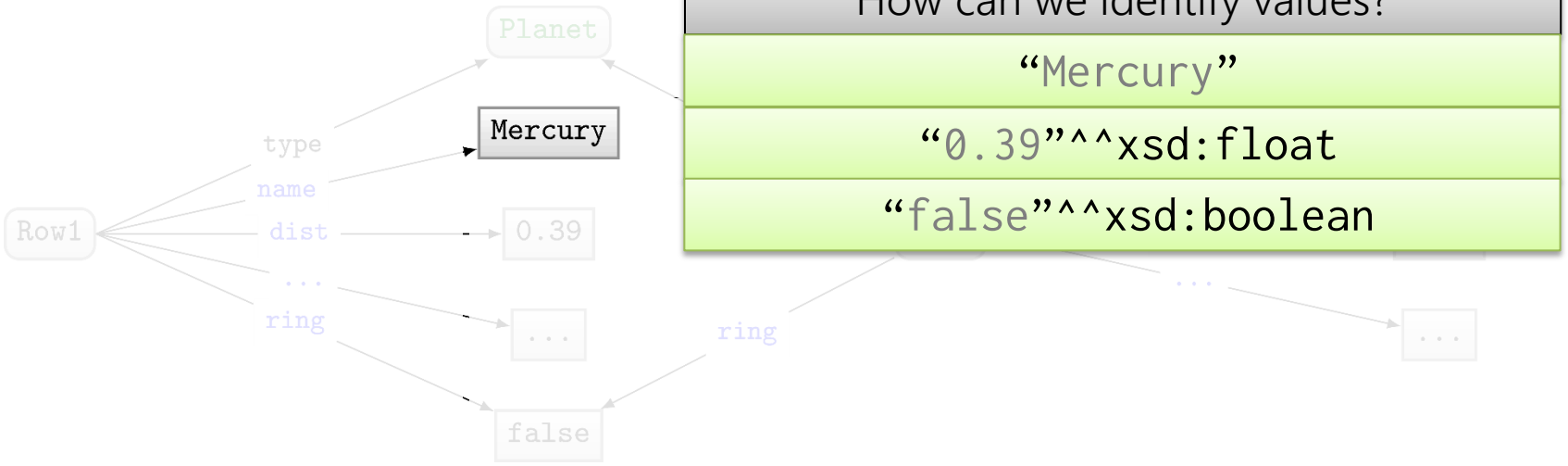
http://ex.org/Planet#dist

http://ex.org/Planet#...

Direct Mapping: Identifying Values

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true



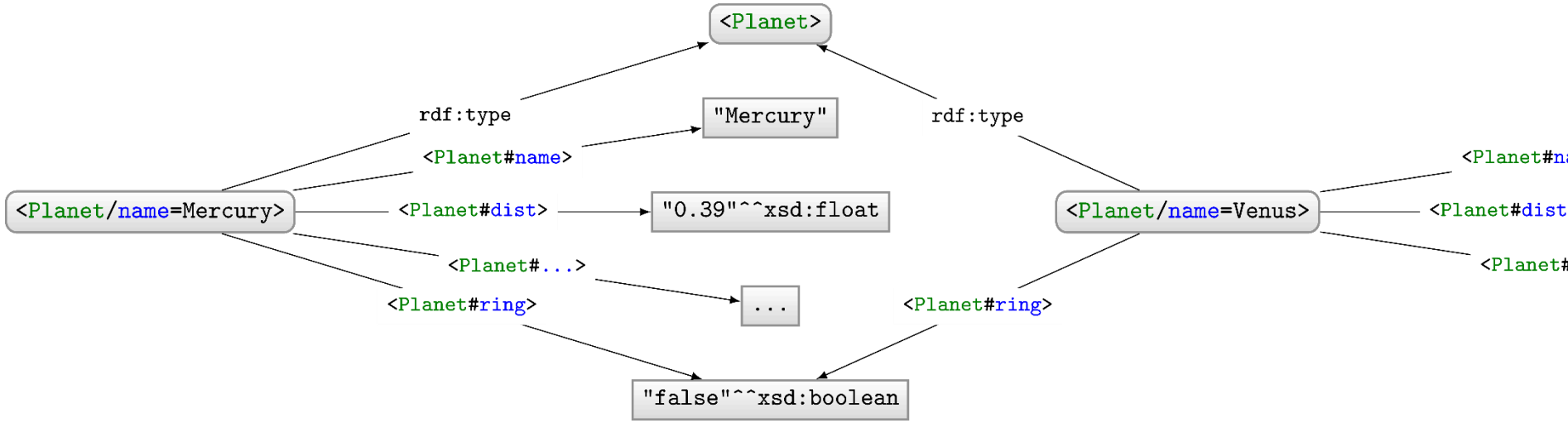
How can we identify values?

- “Mercury”
- “0.39”^^xsd:float
- “false”^^xsd:boolean

Direct Mapping: Final RDF

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true




```
@base : <http://ex.org/>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.
```

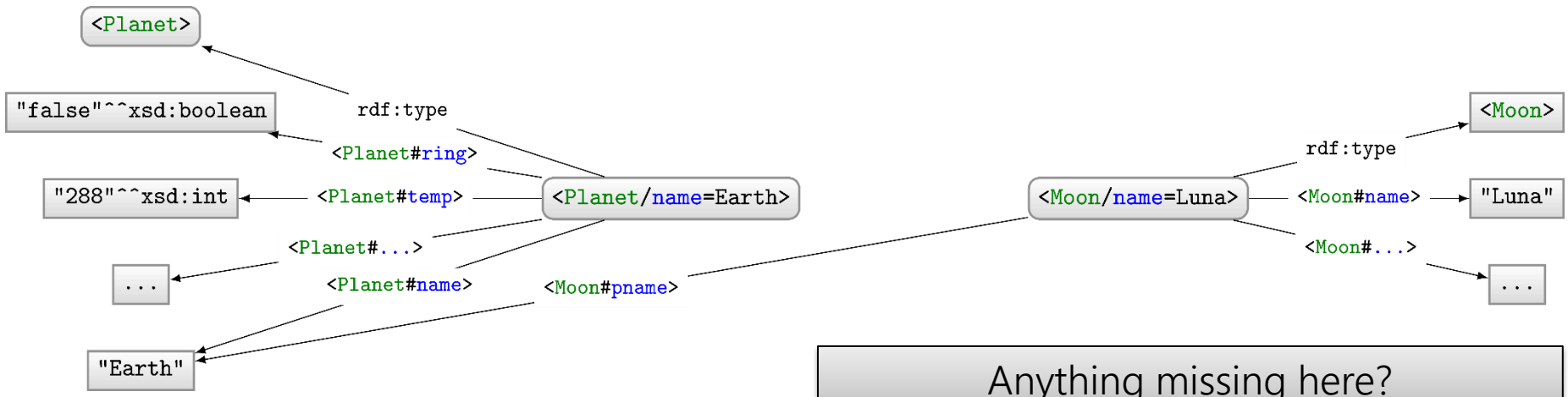
Direct Mapping: Multiple Tables

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

Moon

name	pname	discoverer	year
Luna	Earth	$\perp$	$\perp$
Ganimedes	Jupiter	Galileo Galilei	1610
Calisto	Jupiter	Galileo Galilei	1610
Europa	Jupiter	Galileo Galilei	1610
Io	Jupiter	Galileo Galilei	1610
Titan	Saturn	Christiaan Huygens	1655
Triton	Neptune	William Lassell	1846



Anything missing here?

If we assume that `pname` is a foreign key for `Planet.name`, then ...

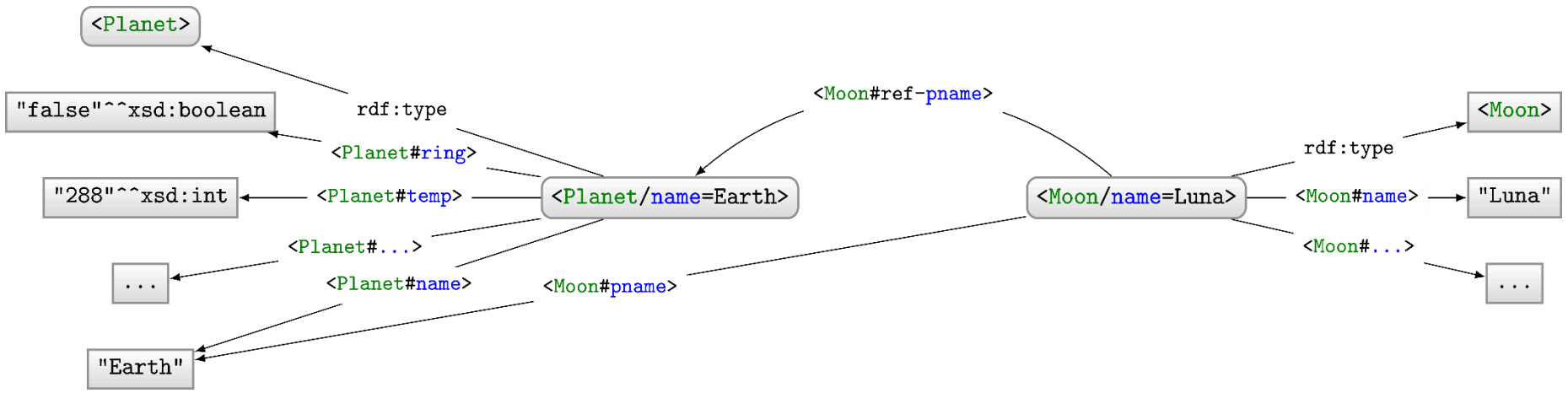
Direct Mapping: Foreign Key References

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

Moon

name	pname	discoverer	year
Luna	Earth	$\perp$	$\perp$
Ganimedes	Jupiter	Galileo Galilei	1610
Calisto	Jupiter	Galileo Galilei	1610
Europa	Jupiter	Galileo Galilei	1610
Io	Jupiter	Galileo Galilei	1610
Titan	Saturn	Christiaan Huygens	1655
Triton	Neptune	William Lassell	1846



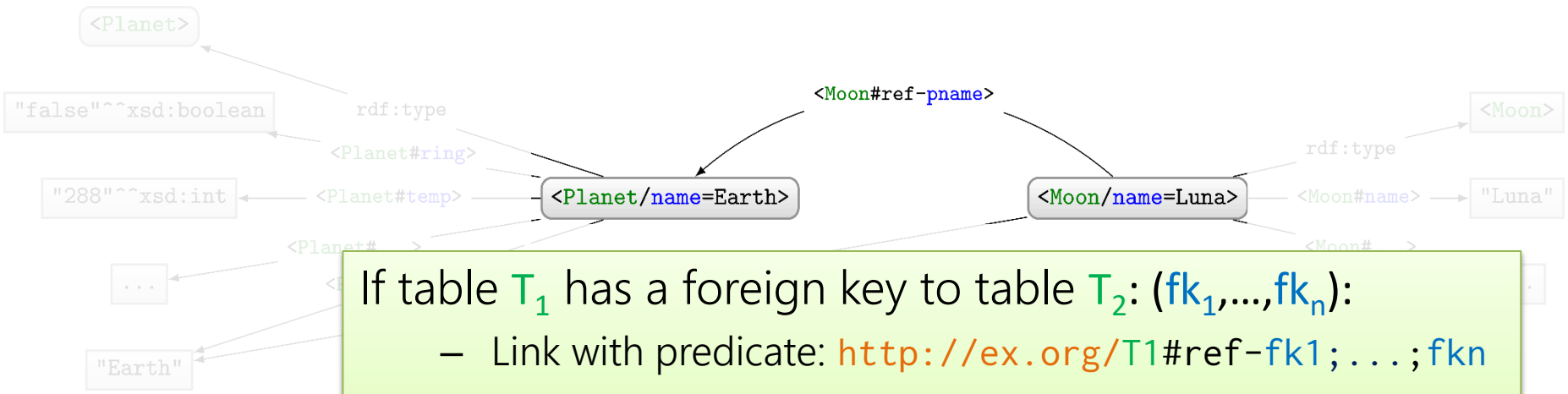
Direct Mapping: Foreign Key References

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

Moon

name	pname	discoverer	year
Luna	Earth	⊥	⊥
Ganimedes	Jupiter	Galileo Galilei	1610
Calisto	Jupiter	Galileo Galilei	1610
Europa	Jupiter	Galileo Galilei	1610
Io	Jupiter	Galileo Galilei	1610
Titan	Saturn	Christiaan Huygens	1655
Triton	Neptune	William Lassell	1846



If table T_1 has a foreign key to table T_2: (fk_1,...,fk_n):

- Link with predicate: `http://ex.org/T1#ref-fk1;...;fkn`
- (Base IRI `http://ex.org/` given as input)

Example: `http://ex.org/Moon#ref-pname`

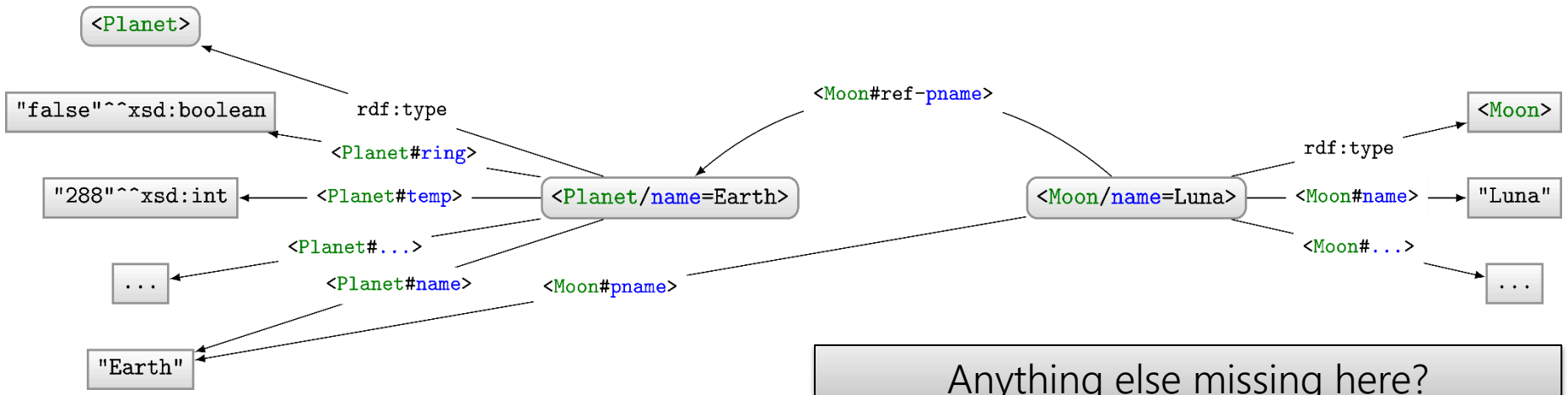
Direct Mapping

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

Moon

name	pname	discoverer	year
Luna	Earth	⊥	⊥
Ganimedes	Jupiter	Galileo Galilei	1610
Calisto	Jupiter	Galileo Galilei	1610
Europa	Jupiter	Galileo Galilei	1610
Io	Jupiter	Galileo Galilei	1610
Titan	Saturn	Christiaan Huygens	1655
Triton	Neptune	William Lassell	1846



Anything else missing here?

What should we do with NULL (⊥)??

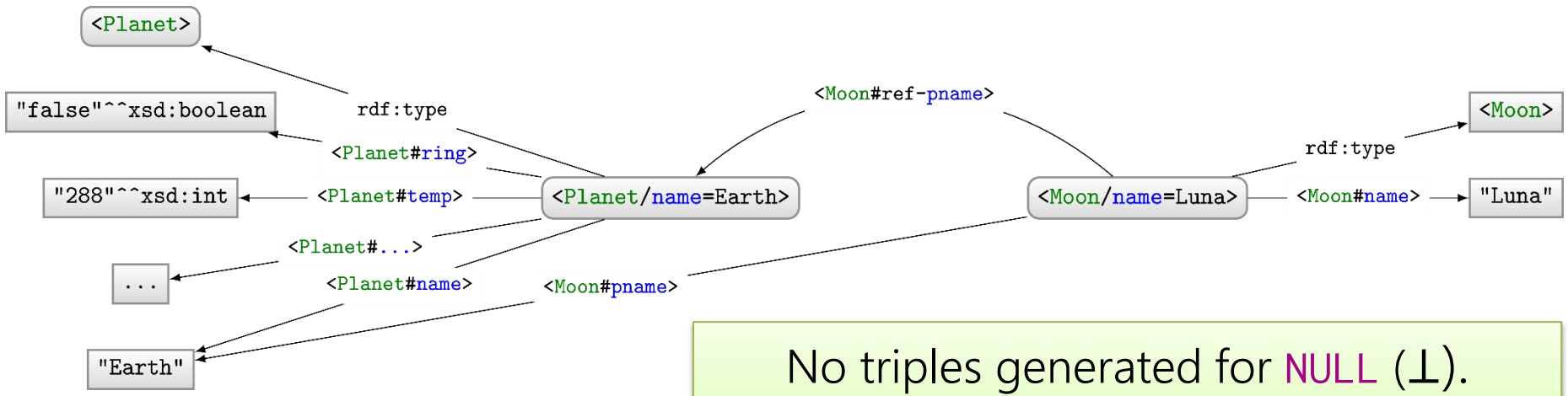
Direct Mapping: NULL (⊥)

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

Moon

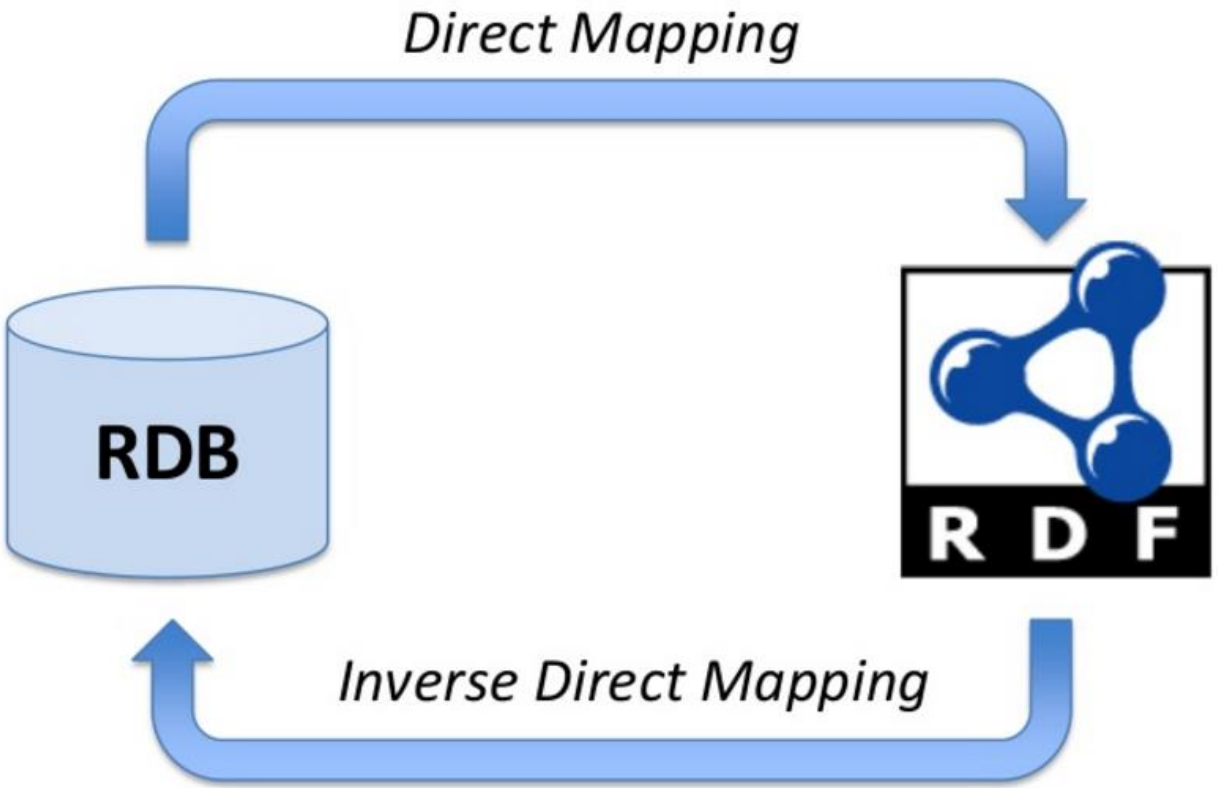
name	pname	discoverer	year
Luna	Earth	⊥	⊥
Ganimedes	Jupiter	Galileo Galilei	1610
Calisto	Jupiter	Galileo Galilei	1610
Europa	Jupiter	Galileo Galilei	1610
Io	Jupiter	Galileo Galilei	1610
Titan	Saturn	Christiaan Huygens	1655
Triton	Neptune	William Lassell	1846



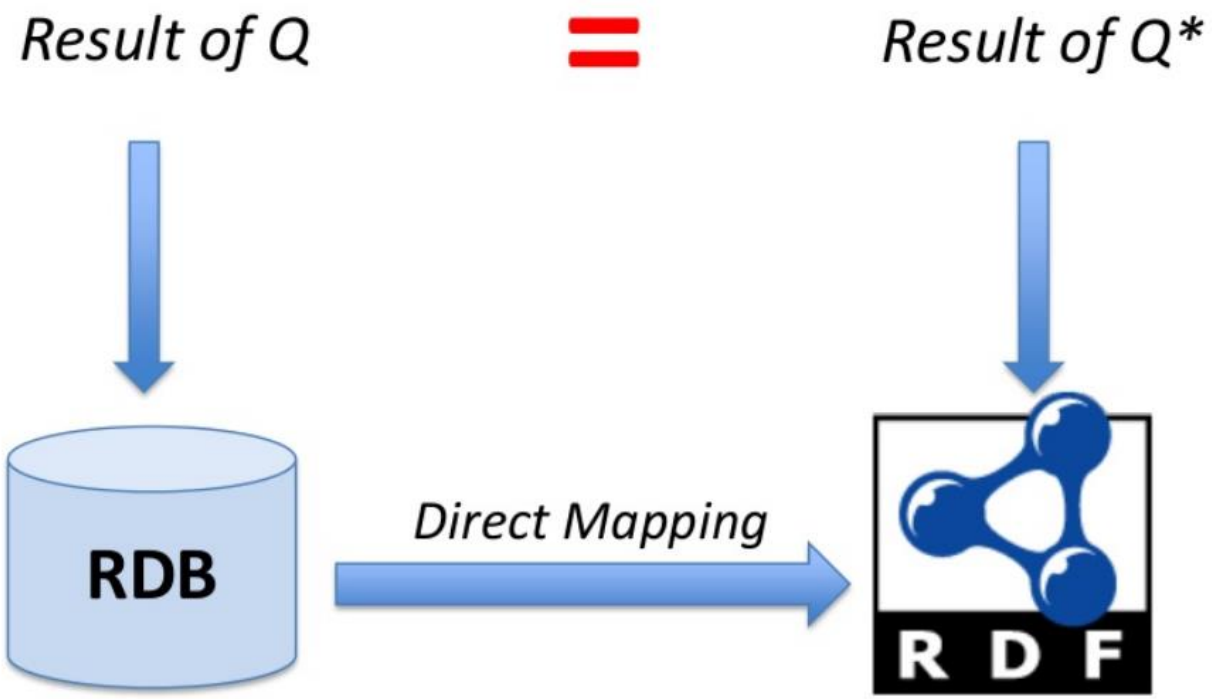
No triples generated for NULL (⊥).

(Semantics of NULL often not clear)

Direct Mapping: Information Preservation



Direct Mapping: Query Preservation



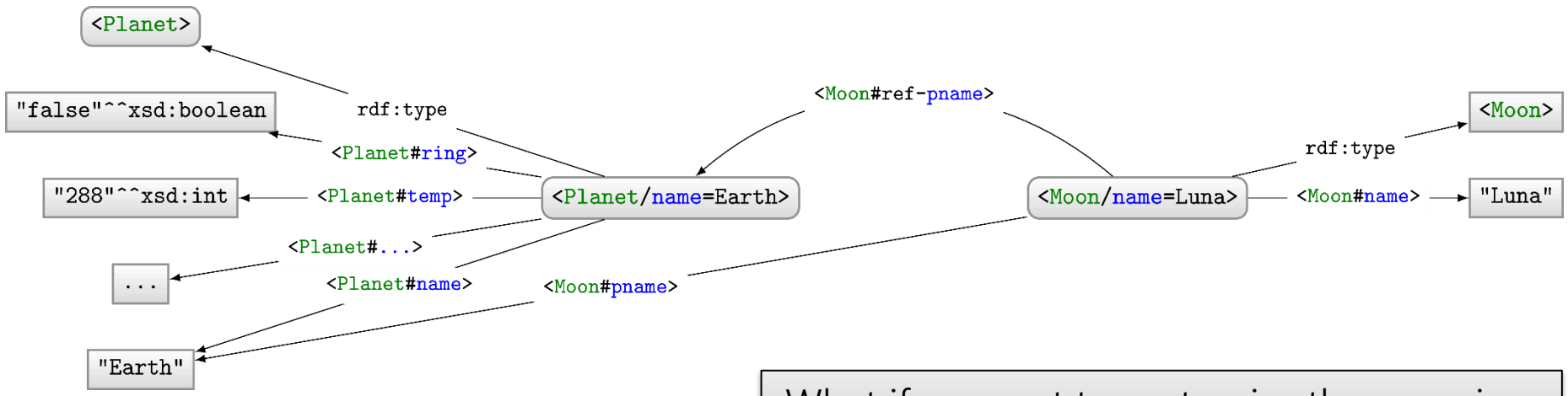
Direct Mapping: Customisation?

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

Moon

name	pname	discoverer	year
Luna	Earth	$\perp$	$\perp$
Ganimedes	Jupiter	Galileo Galilei	1610
Calisto	Jupiter	Galileo Galilei	1610
Europa	Jupiter	Galileo Galilei	1610
Io	Jupiter	Galileo Galilei	1610
Titan	Saturn	Christiaan Huygens	1655
Triton	Neptune	William Lassell	1846



What if we want to customise the mapping, e.g., to create a node for people?

R2RML:
Custom RDB2RDF mappings

W3C Recommendation

R2RML: RDB to RDF Mapping Language

W3C Recommendation 27 September 2012

This version:
http://www.w3.org/TR/2012/REC-r2rml-20120927/

Latest version:
http://www.w3.org/TR/r2rml/

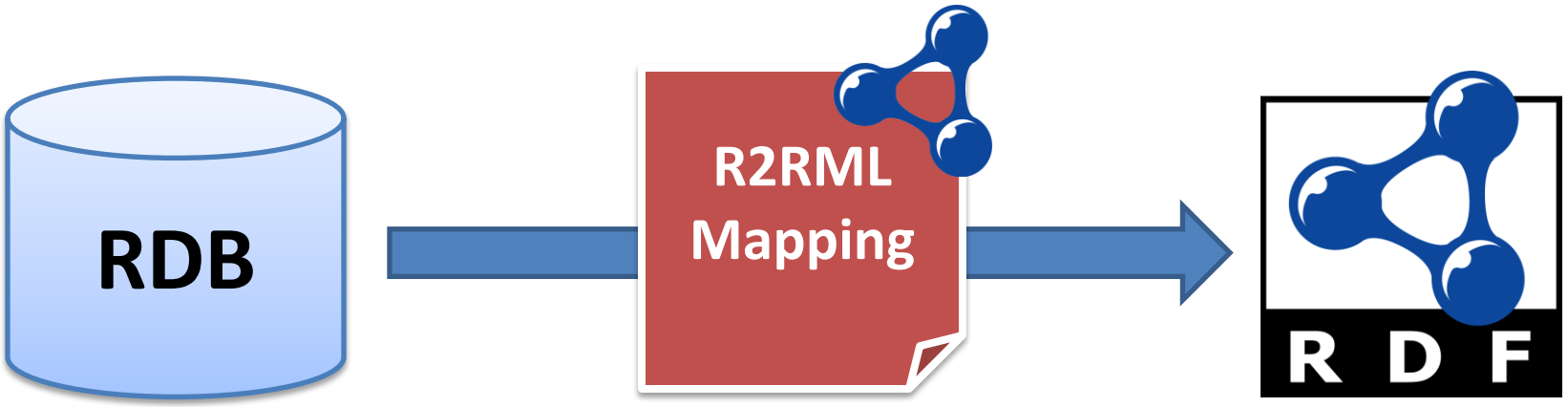
Previous version:
http://www.w3.org/TR/2012/PR-r2rml-20120814/

Editors:
Souripriya Das, Oracle
Seema Sundara, Oracle
Richard Cyganiak, DERI, National University of Ireland, Galway

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

See also **translations**.

R2RML: In a nutshell

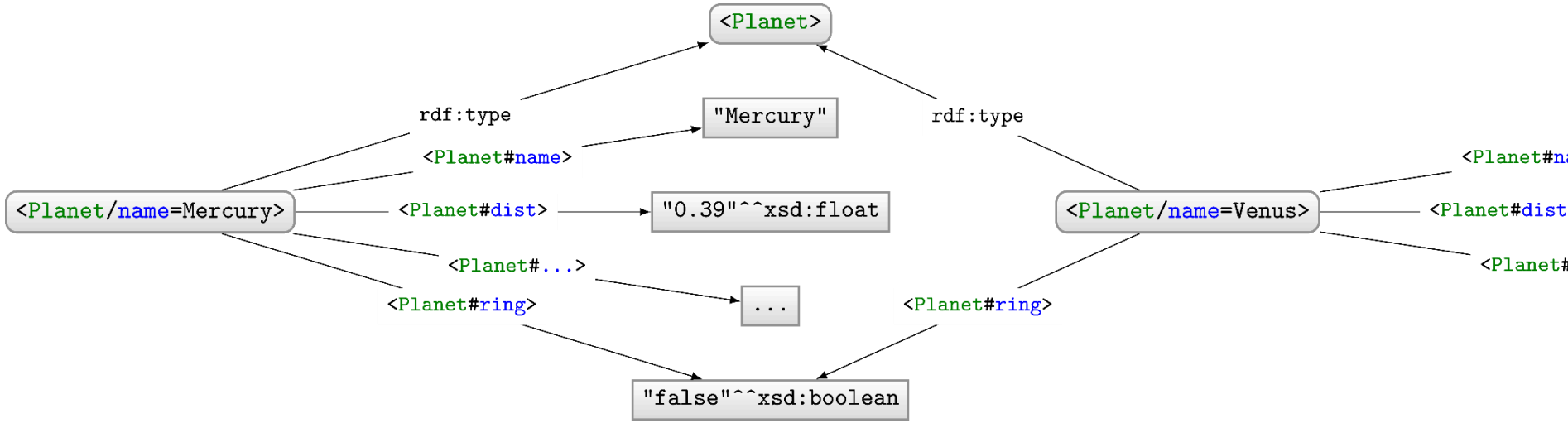


What should we use to specify this R2RML mapping?

R2RML Example: The Direct Mapping

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true




```
@base : <http://ex.org/>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.
```

R2RML Example: The Direct Mapping

```
@prefix rr: <http://www.w3.org/ns/r2rml#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.
@base <http://rr.org/astro#>.

<PlanetMap> a rr:TriplesMap ;

  rr:logicalTable [ rr:tableName "Planet" ] ;

  rr:subjectMap [
     rr:template "http://ex.org/Planet/name={name}" ;
     rr:class <http://ex.org/Planet>
  ] ;

  rr:predicateObjectMap [
     rr:predicate <http://ex.org/Planet#name> ;
     rr:objectMap [ rr:column "name" ]
  ] ;

  rr:predicateObjectMap [
     rr:predicate <http://ex.org/Planet#dist> ;
     rr:objectMap [ rr:column "dist" ; rr:datatype xsd:float ]
  ] ;
  ...
```

Triple Mapping

Source Table

Subject Mapping

Predicate-Object Mapping

Predicate-Object Mapping

...

R2RML: Selecting a Logical Table

```
@prefix rr: <http://www.w3.org/ns/r2rml#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.
@base <http://rr.org/astro#>.

<PlanetMap> a rr:TriplesMap ;

  rr:logicalTable [ rr:tableName "Planet" ] ;
```

Triple Mapping

Source Table

Option 1: Specify table name:

```
  ...
  rr:logicalTable [ rr:tableName "Planet" ] ;
  ...
```

Option 2: Specify SQL query:

```
  ...
  rr:logicalTable [
      rr:sqlQuery "SELECT * FROM Planet WHERE dist>1" ;
      rr:sqlVersion rr:SQL2008
  ] ;
  ...
```

(`rr:sqlVersion` is optional)

R2RML: Example with SQL Query

```
@prefix ...

<OuterPlanetMap> a rr:TriplesMap ;

  rr:logicalTable [
      rr:sqlQuery "SELECT * FROM Planet WHERE dist>1" ;
      rr:sqlVersion rr:SQL2008
  ] ;

  rr:subjectMap [
     rr:template "http://ex.org/Planet/name={name}" ;
     rr:class <http://ex.org/OuterPlanet>
  ] .
```

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

```
@base <http://ex.org/>
@prefix ...

<Planet/name=Mars> a <OuterPlanet> .
<Planet/name=Jupiter> a <OuterPlanet> .
<Planet/name=Saturn> a <OuterPlanet> .
<Planet/name=Neptune> a <OuterPlanet> .
```

R2RML Term Maps: Creating RDF Terms

```
@prefix rr: <http://www.w3.org/ns/r2rml#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.
@base <http://rr.org/astro#>.

<PlanetMap> a rr:TriplesMap ;

  rr:logicalTable [ rr:tableName "Planet" ] ;

  rr:subjectMap [
     rr:template "http://ex.org/Planet/name={name}" ;
     rr:class <http://ex.org/Planet>
  ] ;

  rr:predicateObjectMap [
     rr:predicate <http://ex.org/Planet#name> ;
     rr:objectMap [ rr:column "name" ]
  ] ;

  rr:predicateObjectMap [
     rr:predicate <http://ex.org/Planet#dist> ;
     rr:objectMap [ rr:column "dist" ; rr:datatype xsd:float ]
  ] ;
  ...
```

Triple Mapping

Source Table

Subject Mapping

Predicate-Object Mapping

Predicate-Object Mapping

...

R2RML Term Maps: Creating RDF Terms

Option 1: Specify a constant:

```
...
rr:objectMap [ rr:constant "Solar System" ] ;
...
```

Option 2: Select from a table column:

```
...
rr:objectMap [ rr:column "dist" ] ;
...
```

Option 3: Template using table columns:

```
...
rr:objectMap [ rr:template "http://ex.org/Moon/{name}_({pname})" ] ;
...
```

R2RML Term Maps: Constants

Option 1: Specify a constant:

```
...
rr:objectMap [ rr:constant "Solar System" ] ;
...
```

Or use the shortcut form:

```
...
rr:object "Solar System" ;
...
```

Can also use for IRIs:

```
...
rr:object <http://ex.org/Solar_System> ;
...
```

R2RML Term Maps: Columns

Option 2: Select from a table column:

```
...
rr:objectMap [ rr:column "dist" ] ;
...
```

By default generates ...
... literals for obj. (datatype based on RDB), IRIs for sub. or pred.
... but can use `rr:termType` to override:

`rr:IRI`, `rr:BlankNode` or `rr:Literal`

```
...
rr:objectMap [ rr:column "homepage" ; rr:termType rr:IRI ] ;
...
```

If a literal, can specify `rr:datatype` or `rr:language`

```
...
rr:objectMap [ rr:column "dist" ; rr:datatype xsd:float ] ;
...
```

```
...
rr:objectMap [ rr:column "name" ; rr:language "en" ] ;
...
```

R2RML Term Maps: Templates

Option 3: Template using table columns:

```
...
rr:objectMap [ rr:column "http://ex.org/Moon/{name}_({pname})" ] ;
...
```

By default generates IRIs ...
... but can use `rr:termType` to specify:

`rr:IRI`, `rr:BlankNode` or `rr:Literal`

```
...
rr:objectMap [
    rr:template "{name}_({pname})" ;
    rr:termType rr:Literal
] ;
...
```

If a literal, can (again) specify `rr:datatype` or `rr:language`

R2RML Term Maps

Term map should not break restrictions on positions:

subject	*predicate*	*object*
[IRI, Blank Node]	[IRI]	[IRI, Blank Node, Literal]

R2RML Term Maps: Example

```
@prefix ...
<PlanetMap> a rr:TriplesMap ;
  rr:logicalTable [ rr:tableName "Planet" ] ;

  rr:subjectMap [
     rr:template "http://ex.org/p/{name}"
  ] .

  rr:predicateObjectMap [
     rr:predicate <http://ex.org/v/dist> ;
     rr:objectMap [ rr:column "dist" ; rr:datatype xsd:float ]
  ] .
```

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

```
@base <http://ex.org/>
@prefix ...

<p/Mercury> <v/dist> "0.39"^^xsd:float
<p/Venus> <v/dist> "0.72"^^xsd:float .
<p/Earth> <v/dist> "1.00^^xsd:float .
...
```

R2RML: Subject Map

```
@prefix rr: <http://www.w3.org/ns/r2rml#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.
@base <http://rr.org/astro#>.

<PlanetMap> a rr:TriplesMap ;

  rr:logicalTable [ rr:tableName "Planet" ] ;

  rr:subjectMap [
     rr:template "http://ex.org/Planet/name={name}" ;
     rr:class <http://ex.org/Planet>
  ] ;
```

Triple Mapping

Source Table

Subject Mapping

One per triple map

Specifies one term map (IRI/Blank Node)

Specifies zero or many types (`rr:class`)

Predicate-Object Mapping

Predicate-Object Mapping

...

R2RML: Example with multiple types

```
@prefix ...

<OuterPlanetMap> a rr:TriplesMap ;

  rr:logicalTable [
      rr:sqlQuery "SELECT * FROM Planet WHERE dist>1" ;
      rr:sqlVersion rr:SQL2008
  ] ;

  rr:subjectMap [
     rr:template "http://ex.org/Planet/name={name}" ;
     rr:class <OuterPlanet> , <Planet>
  ] .
```

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

```
@base <http://ex.org/>
@prefix ...

<Planet/name=Mars> a
    <OuterPlanet> , <Planet> .
<Planet/name=Jupiter> a
    <OuterPlanet> , <Planet> .
...
```

R2RML: Predicate–Object Map

```
@prefix rr: <http://www.w3.org/ns/r2rml#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.
@base <http://rr.org/astro#>.

<PlanetMap> a rr:TriplesMap ;

  rr:logicalTable [ rr:tableName "Planet" ] ;

  rr:subjectMap [
     rr:template "http://ex.org/Planet/name={name}" ;
     rr:class <http://ex.org/Planet>
  ] ;

  rr:predicateObjectMap [
     rr:predicate <http://ex.org/Planet#name> ;
     rr:objectMap [ rr:column "name" ]
  ] ;

  rr:predicateObjectMap [
     rr:predicate <http://ex.org/Planet#dist> ;
     rr:objectMap [ rr:column "dist" ; rr:datatype xsd:float ]
  ] ;
  ...
```

Triple Mapping

Source Table

Subject Mapping

Predicate–Object Mapping

Predicate–Object Mapping

...

R2RML: Predicate–Object Map

Zero or more per triple map

Each has one or more predicate term map, one or more object term map

```
rr:predicateObjectMap [
   rr:predicate <http://ex.org/Planet#name> ;
   rr:objectMap [ rr:column "name" ]
] ;
```

Predicate–Object Mapping

```
rr:predicateObjectMap [
   rr:predicate <http://ex.org/Planet#dist> ;
   rr:objectMap [ rr:column "dist" ; rr:datatype xsd:float ]
] ;
...
```

Predicate–Object Mapping

...

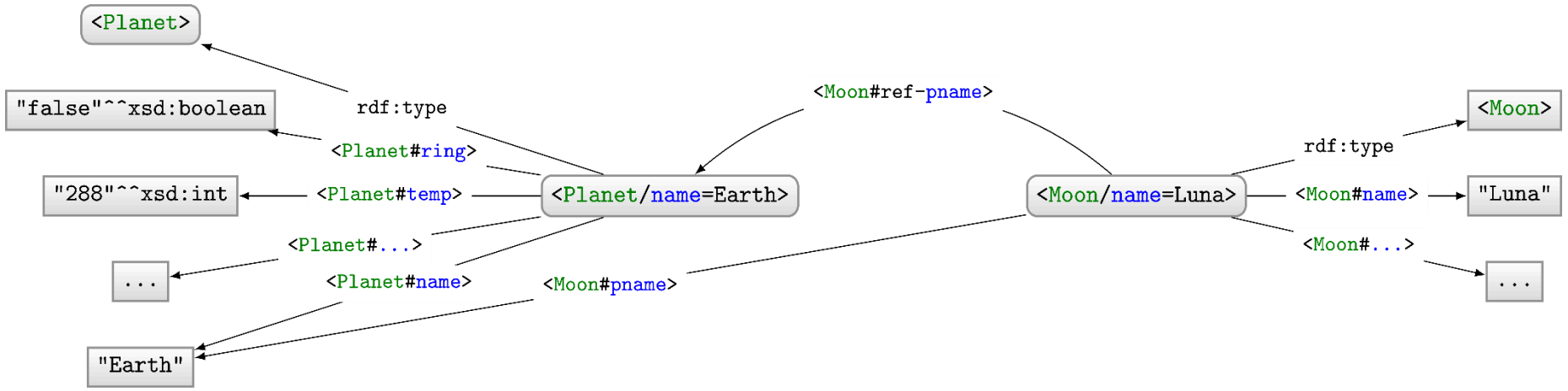
R2RML: Reference Object Map

Planet

name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

Moon

name	pname	discoverer	year
Luna	Earth	$\perp$	$\perp$
Ganimedes	Jupiter	Galileo Galilei	1610
Calisto	Jupiter	Galileo Galilei	1610
Europa	Jupiter	Galileo Galilei	1610
Io	Jupiter	Galileo Galilei	1610
Titan	Saturn	Christiaan Huygens	1655
Triton	Neptune	William Lassell	1846



R2RML: Reference Object Map

```
@prefix ...
<PlanetMap> a rr:TriplesMap ;
  rr:logicalTable [ rr:tableName "Planet" ] ;
  rr:subjectMap [
     rr:template "http://ex.org/Planet/name={name}" ;
     rr:class <http://ex.org/Planet>
  ] ; ...

<MoonMap> a rr:TriplesMap ;
  rr:logicalTable [ rr:tableName "Moon" ] ;
  rr:subjectMap [
     rr:template "http://ex.org/Moon/name={name}" ;
     rr:class <http://ex.org/Moon>
  ] ;
  rr:predicateObjectMap [
     rr:predicate <http://ex.org/Moon#ref-pname> ;
     rr:objectMap [
        rr:parentTriplesMap <PlanetMap> ;
        rr:joinCondition [
           rr:child "pname" ; rr:parent "name"
        ]
     ]
  ] ; ...
```

Zero or more join conditions
(zero when parent and child share logical table)

Child object will be parent subject ...

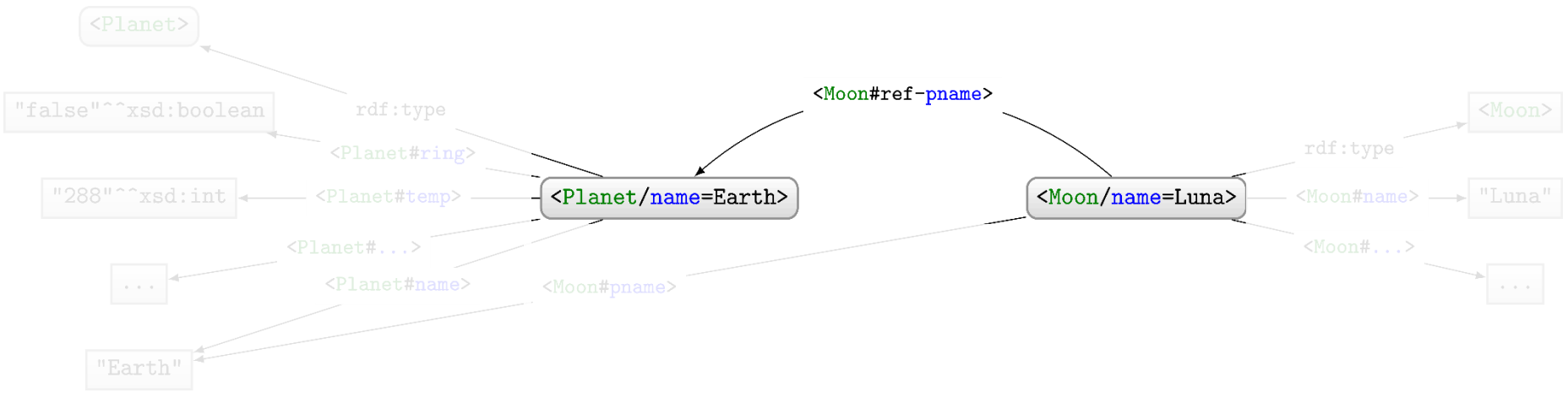
R2RML: Reference Object Map

Planet

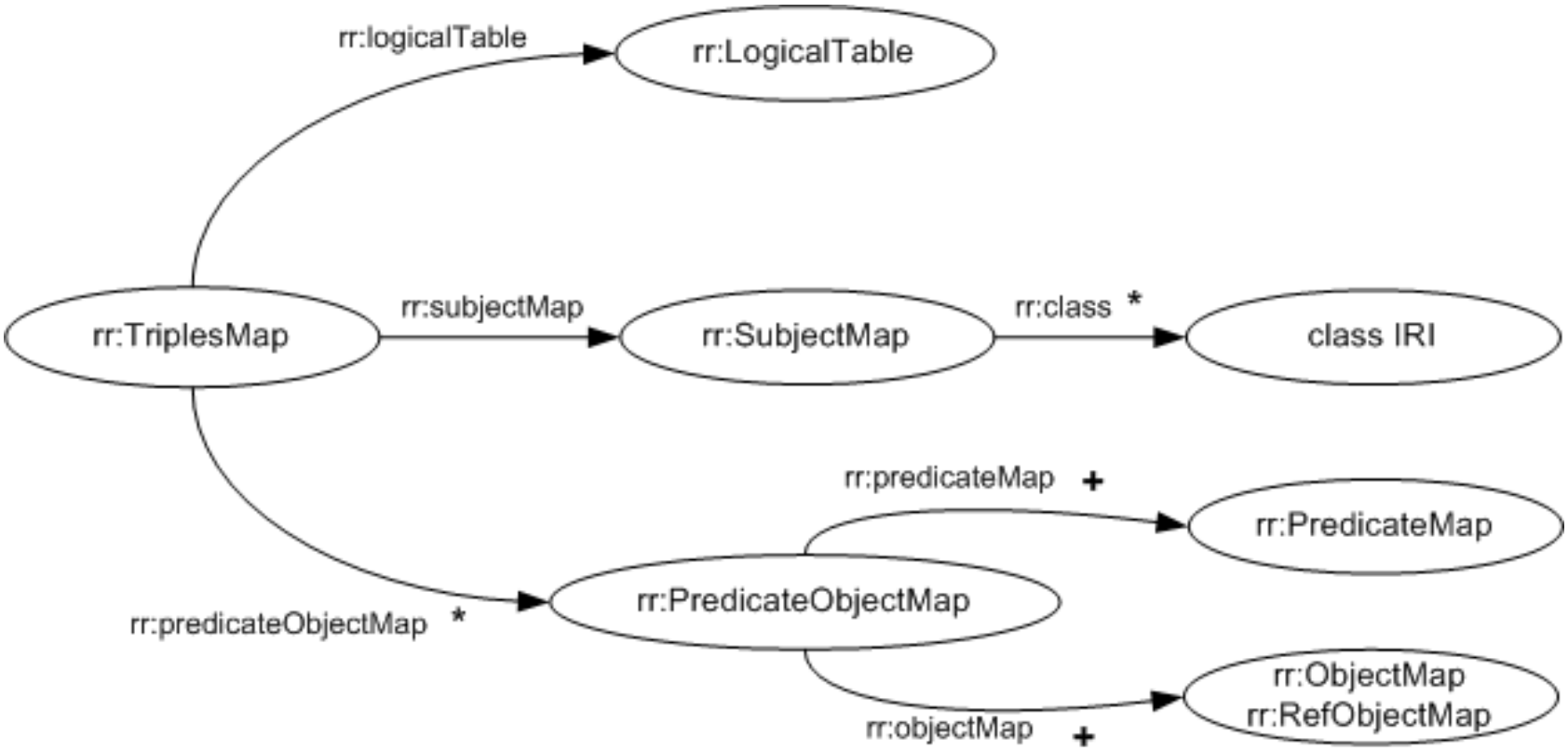
name	dist	radius	grav	day	year	temp	ring
Mercury	0.39	0.38	2.8	58.646	0.241	440	false
Venus	0.72	0.95	8.9	-243.019	0.615	730	false
Earth	1.00	1.00	9.8	0.997	1.000	288	false
Mars	1.52	0.53	3.7	1.026	1.880	186	false
Jupiter	5.20	10.97	22.9	0.414	11.862	152	true
Saturn	9.54	9.14	9.1	0.444	29.447	134	true
Uranus	19.19	3.98	7.8	-0.719	84.017	76	true
Neptune	30.07	3.86	11.0	0.671	164.791	53	true

Moon

name	pname	discoverer	year
Luna	Earth	$\perp$	$\perp$
Ganimedes	Jupiter	Galileo Galilei	1610
Calisto	Jupiter	Galileo Galilei	1610
Europa	Jupiter	Galileo Galilei	1610
Io	Jupiter	Galileo Galilei	1610
Titan	Saturn	Christiaan Huygens	1655
Triton	Neptune	William Lassell	1846



Summary of Triple Map Structure



* zero or more

\+ one or more

(otherwise, precisely one)

R2RML: Graph Maps

```
@prefix ...
<MoonMap> a rr:TriplesMap ;
  rr:logicalTable [ rr:tableName "Moon" ] ;
  rr:subjectMap [
     rr:template "http://ex.org/Moon/name={name}" ;
     rr:class <http://ex.org/Moon> ;
     rr:graph <http://ex.org/graph/SolarSystem>
  ] ;
  rr:predicateObjectMap [
     rr:predicate <http://ex.org/Moon/ref-pname> ;
     rr:objectMap [
         rr:parentTriplesMap <PlanetMap> ;
         rr:joinCondition [
            rr:child "pname" ; rr:parent "name"
         ]
     ] ;
     rr:graphMap [
        rr:template "http://ex.org/graph/{pname}"
     ] ;
     rr:graph rr:defaultGraph
  ] ;
  rr:predicateObjectMap [
     rr:predicate <http://ex.org/discoveryYear> ;
     rr:objectMap [ rr:column "year" ] ;
     rr:graphMap [
        rr:template "http://ex.org/graph/{year}"
     ]
  ]
```

Triples added to named graph(s)

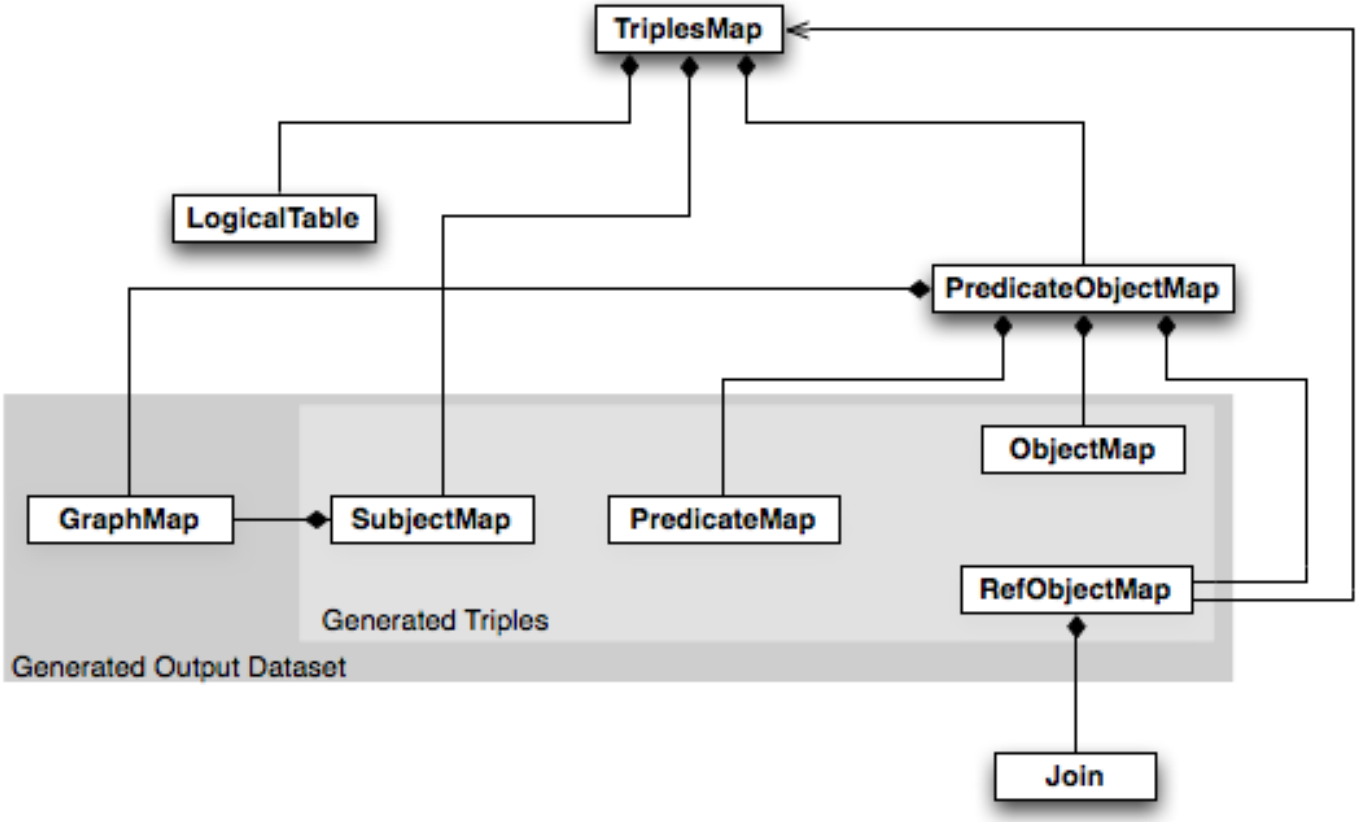
Graphs specified by term maps

`rr:defaultGraph` for default graph

Zero or many graph maps
on subject or predicate-object map

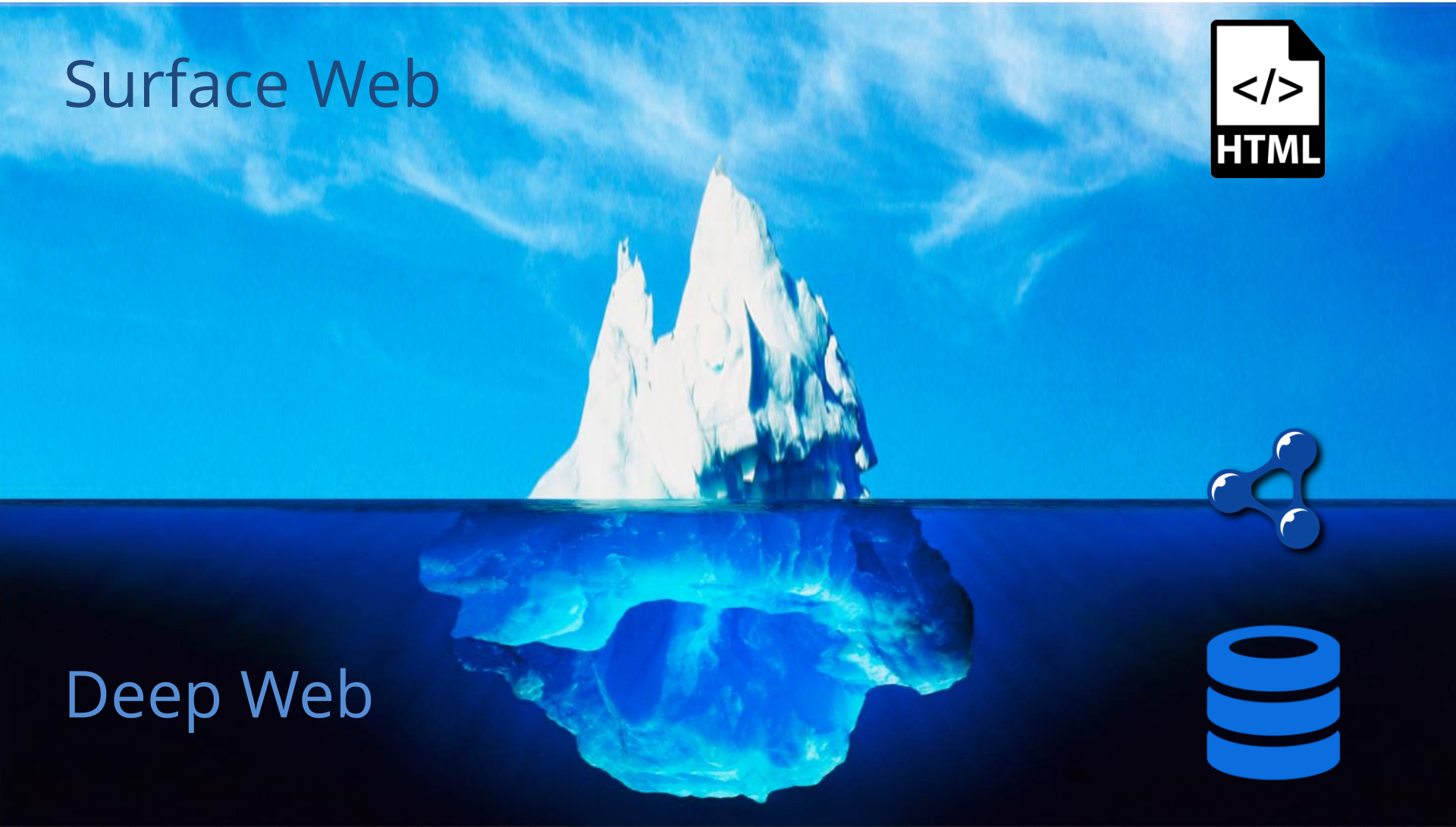
Predicate-object maps "inherit"
from subject map

R2RML ...


TriplesMap
LogicalTable
PredicateObjectMap
ObjectMap
GraphMap
SubjectMap
PredicateMap
RefObjectMap
Generated Triples
Generated Output Dataset
Join

RDB2RDF

RDB2RDF: Surfacing (some of) the Deep Web



Questions?