

# Web-Scale Semantic Data Integration: Challenges and Perspective

**David Chaves-Fraga**

Ontology Engineering Group  
Universidad Politécnica de Madrid, Spain  
&  
Declarative Languages and Artificial Intelligence,  
KULeuven, Belgium

Postdoctoral research at Ontology Engineering Group (UPM) and DTAI (KULeuven)



PhD in Artificial Intelligence at Universidad Politécnica de Madrid (2021)



Co-chair W3C Community Group on Knowledge Graph Construction (2019-now)

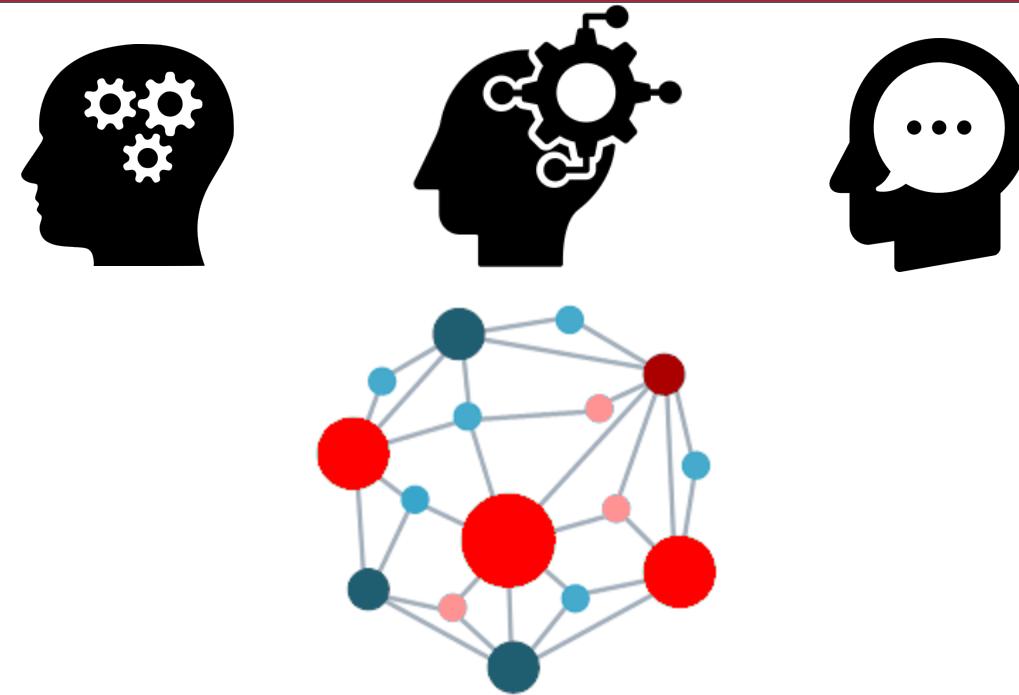


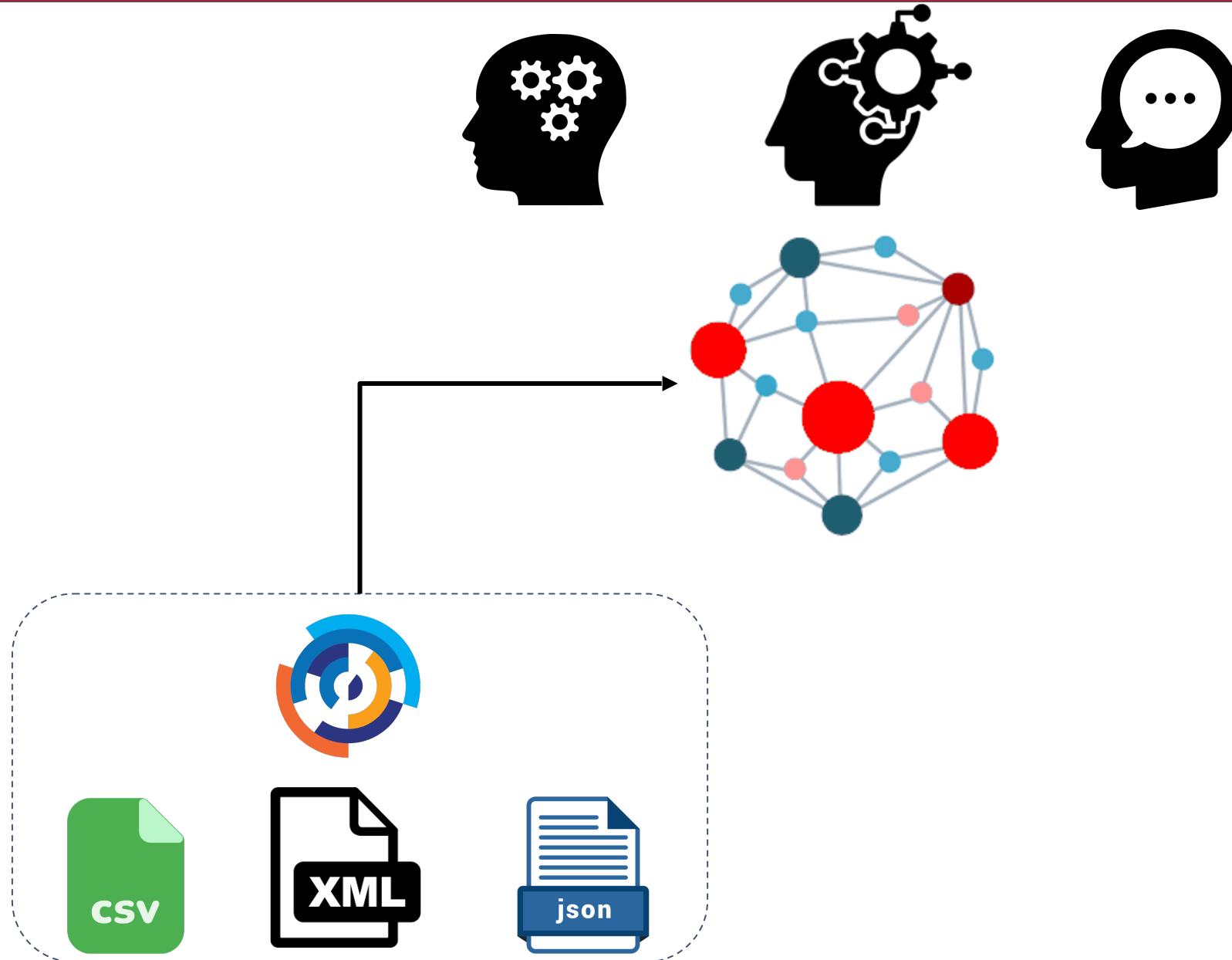
Main author of the Spanish guideline for generating RDF graphs from tabular data (2021)

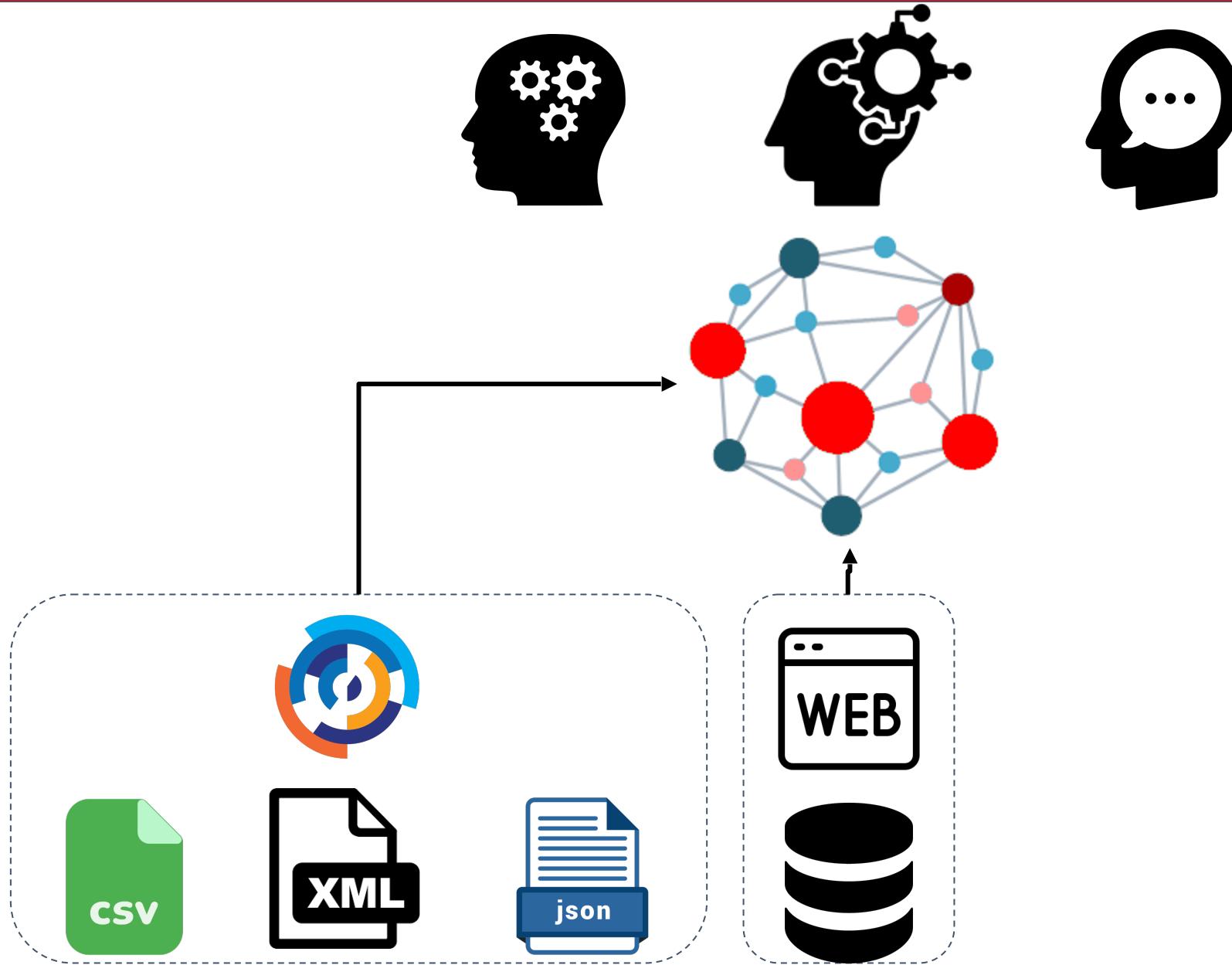


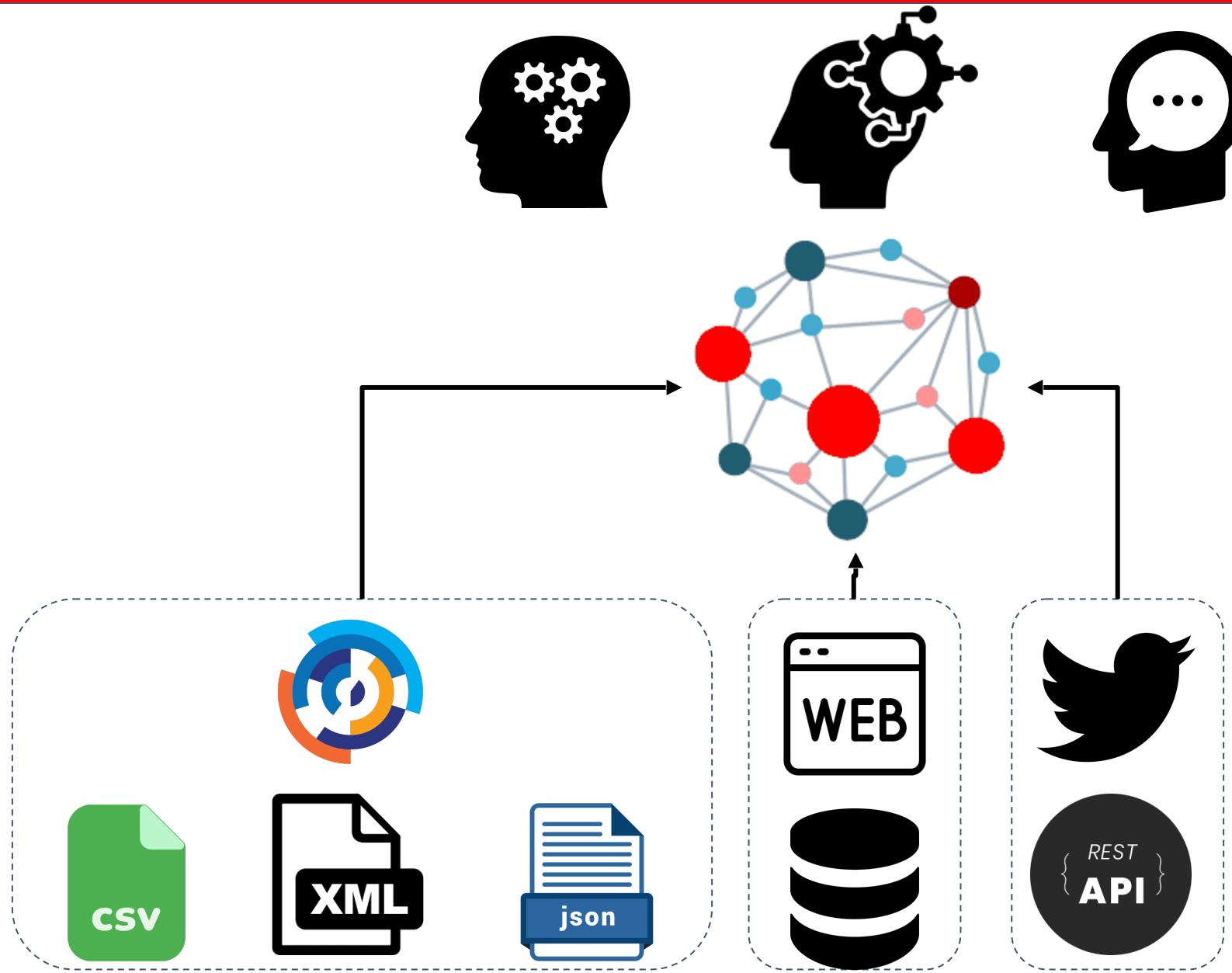
Principal Scientist of the Public Procurement Data Space (2022-2023)

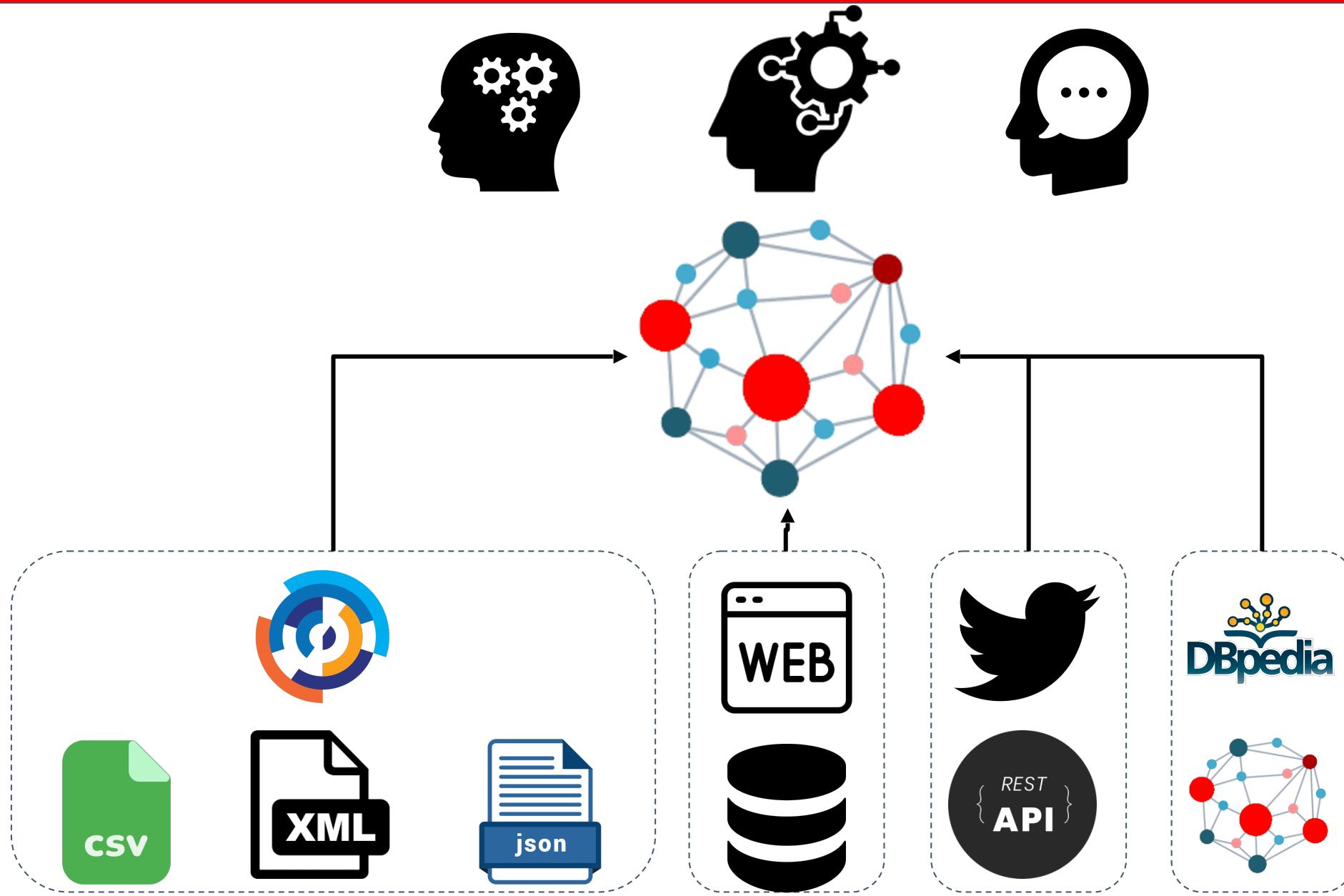


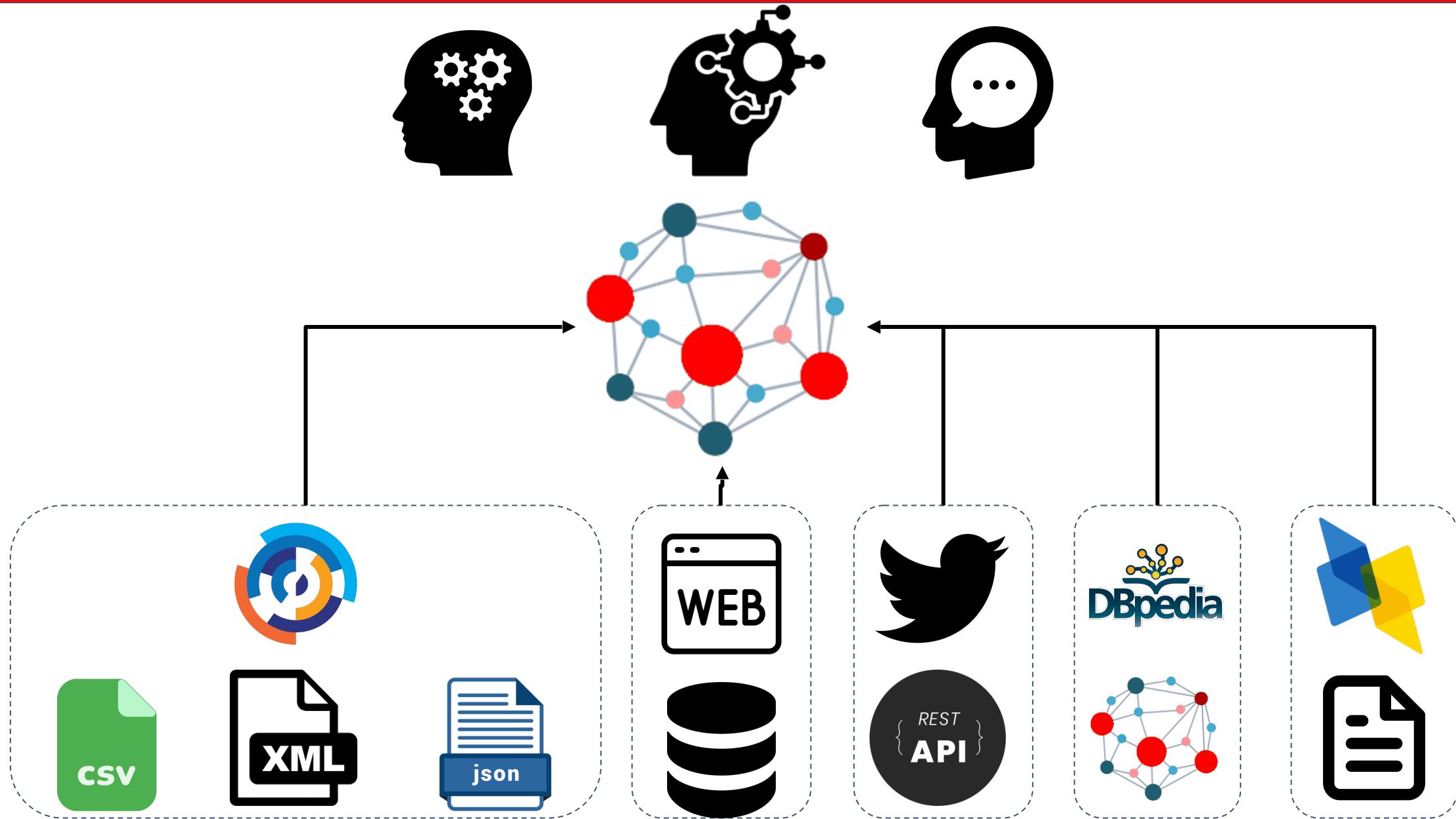


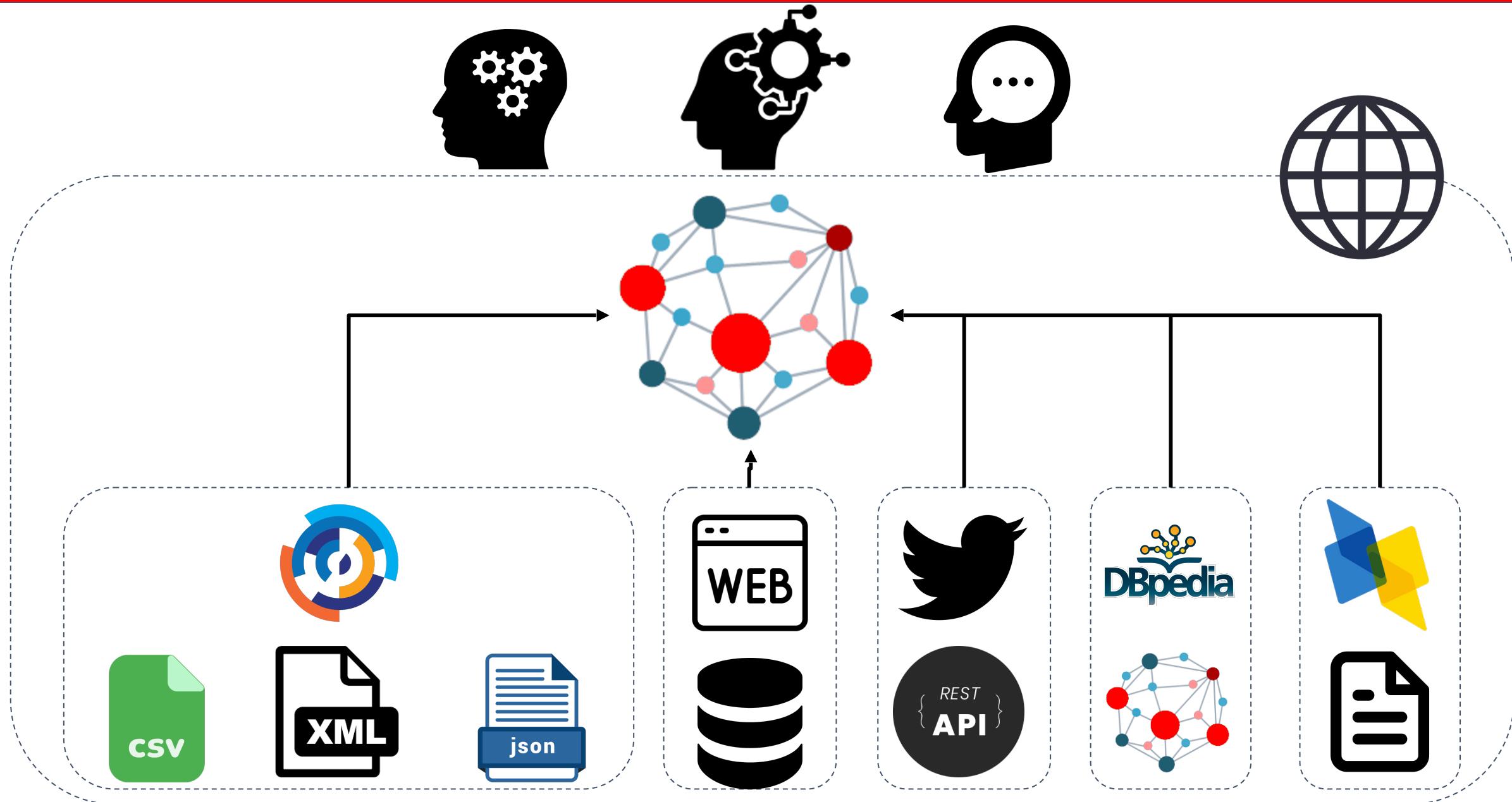




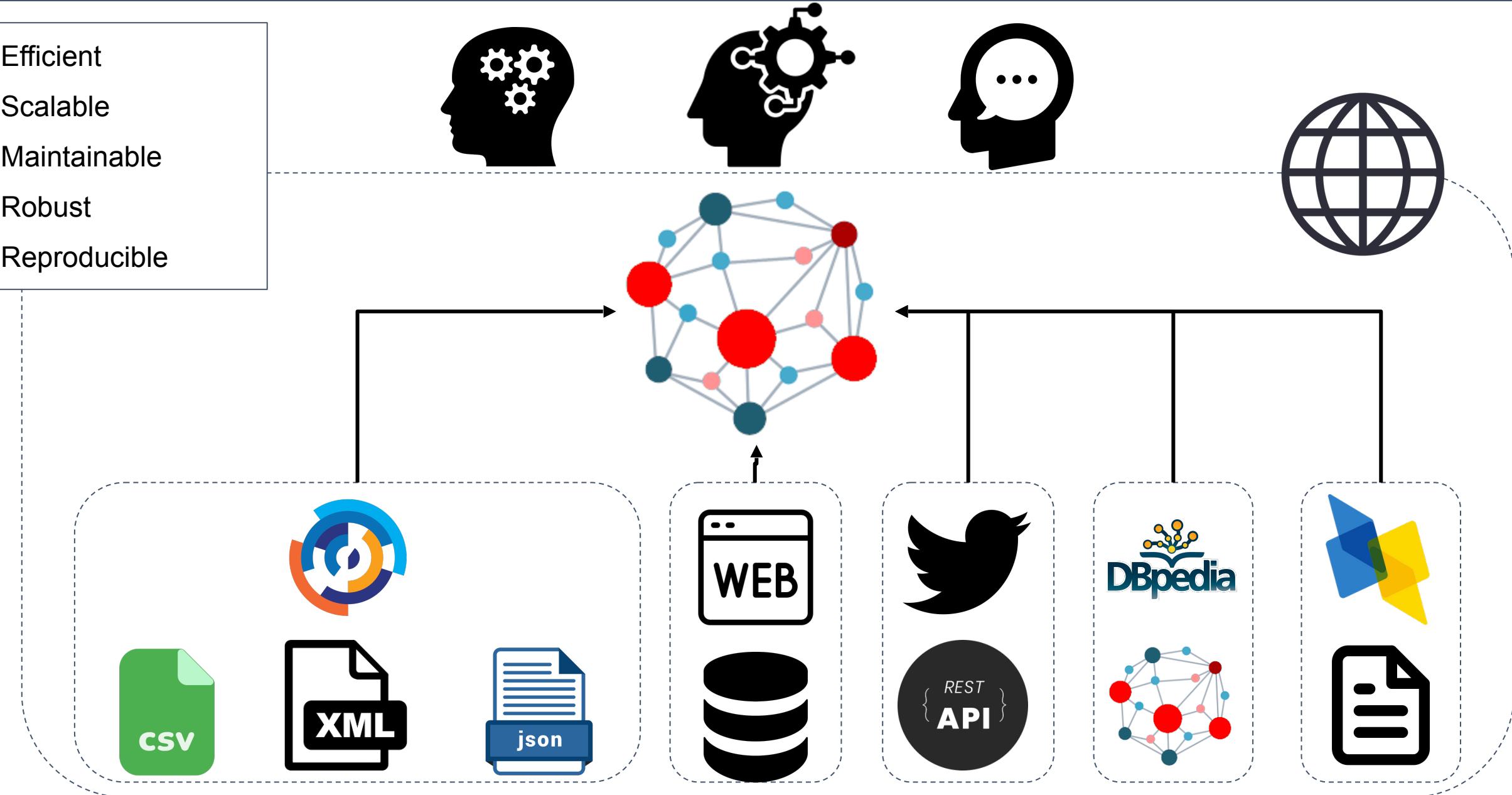




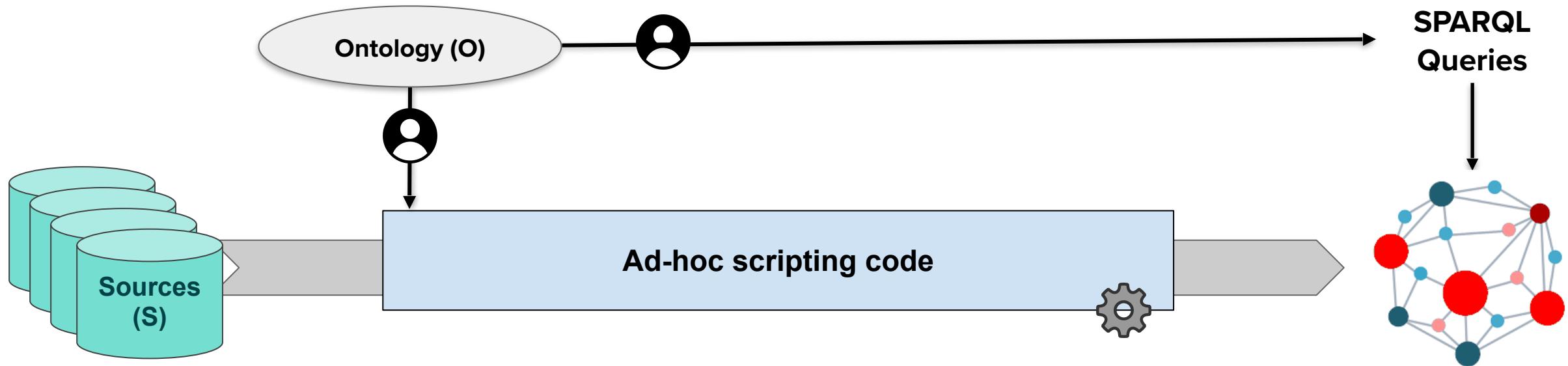




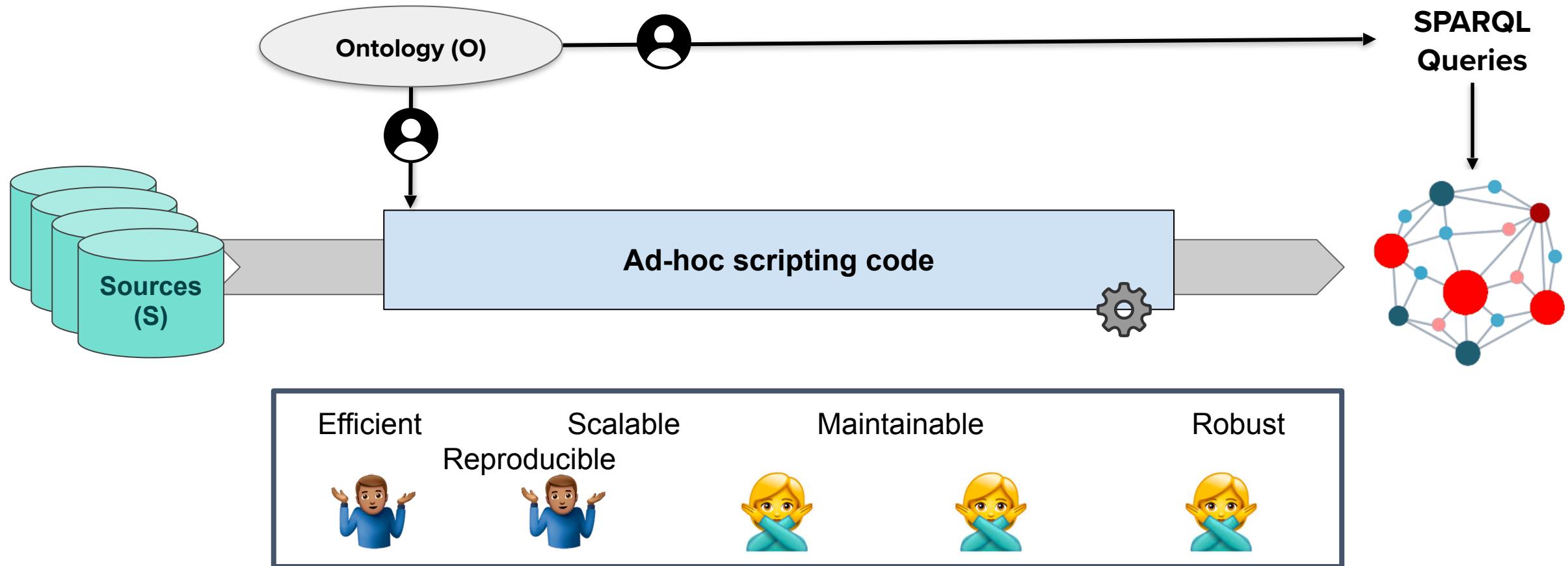
Efficient  
Scalable  
Maintainable  
Robust  
Reproducible

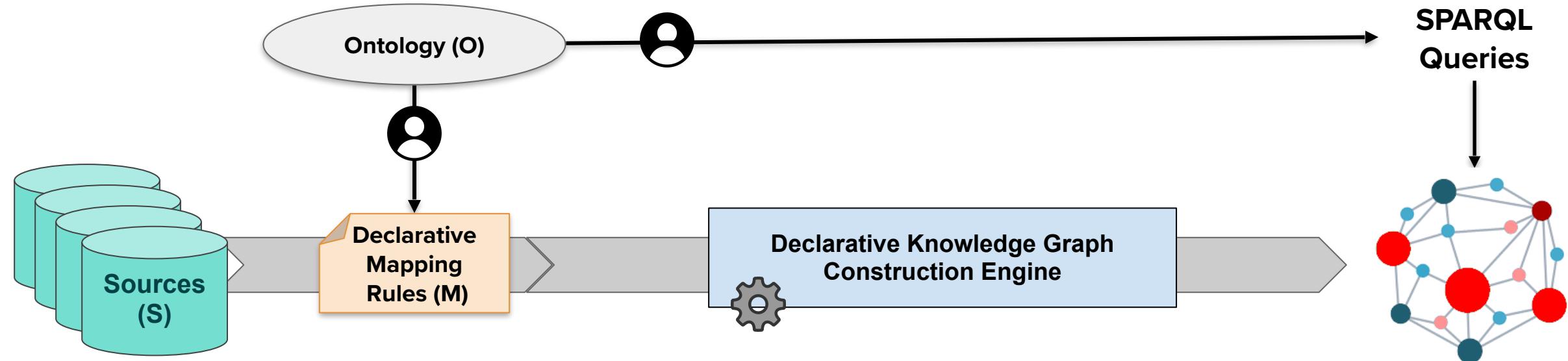


# Knowledge Graph Construction: Scripting-based



# Knowledge Graph Construction: Scripting-based



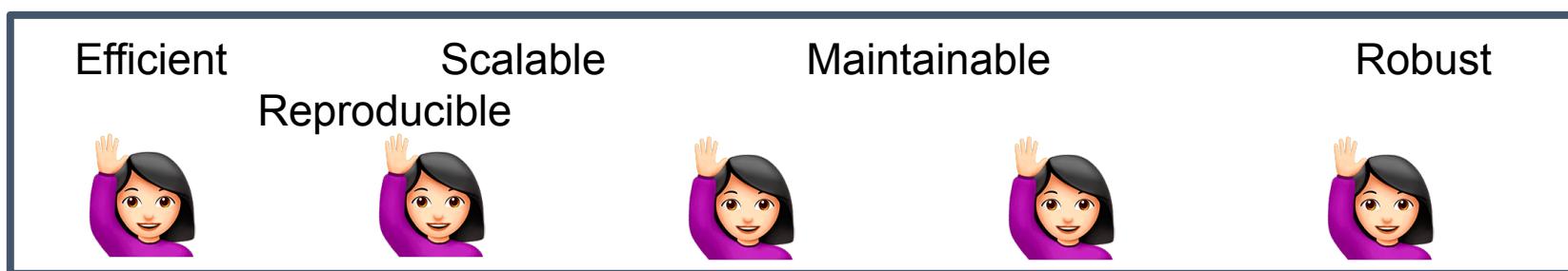
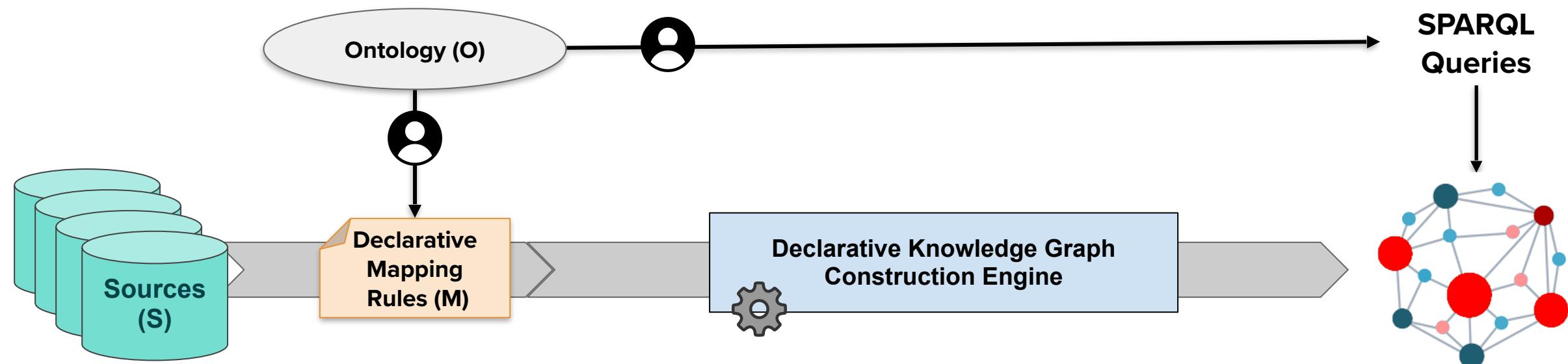


**Knowledge Graph Construction = Data Integration System (DIS) =  $\langle S, M, O \rangle$**



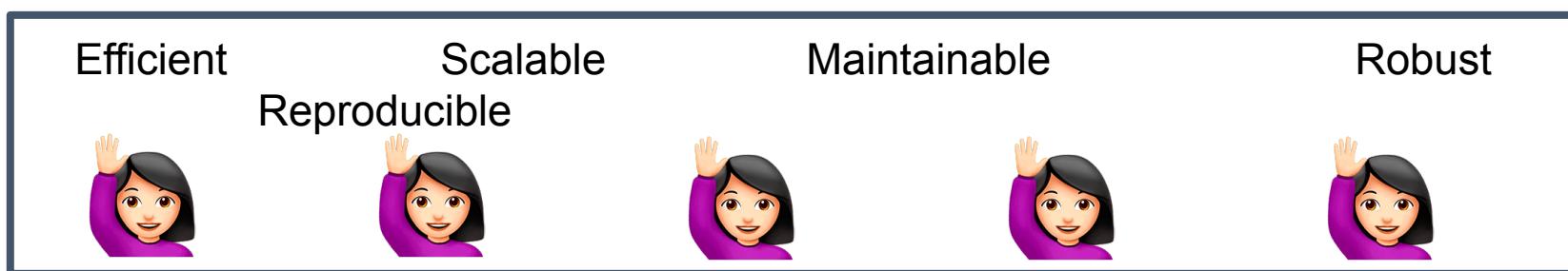
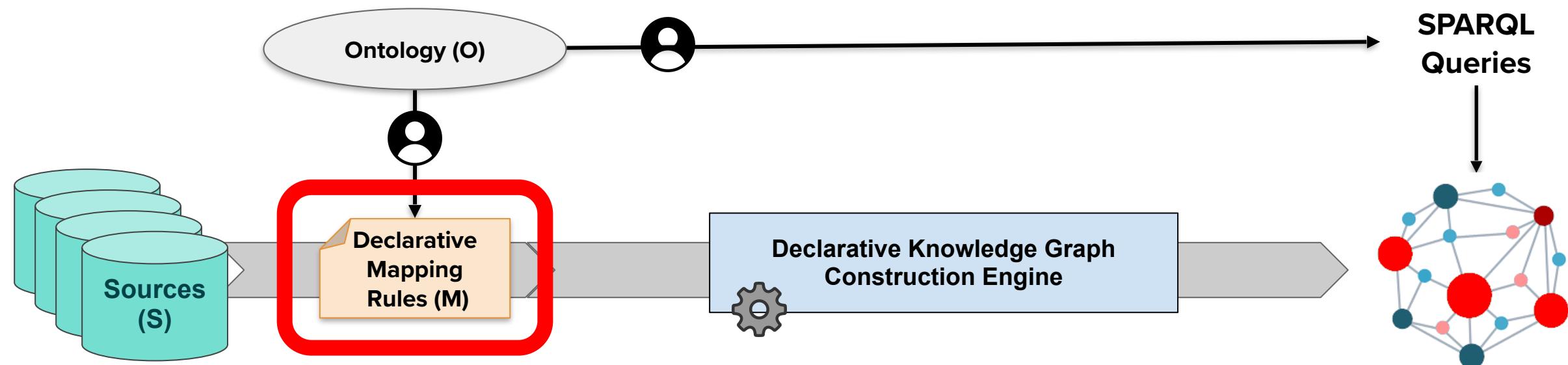
Poggi, A., Lembo, D., Calvanese, D., De Giacomo, G., Lenzerini, M., & Rosati, R. (2008). Linking data to ontologies. In *Journal on data semantics X*  
Lenzerini, M. Data integration: A theoretical perspective. In *Proceedings of the 21st ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*

**Knowledge Graph Construction = Data Integration System (DIS) =  $\langle S, M, O \rangle$**



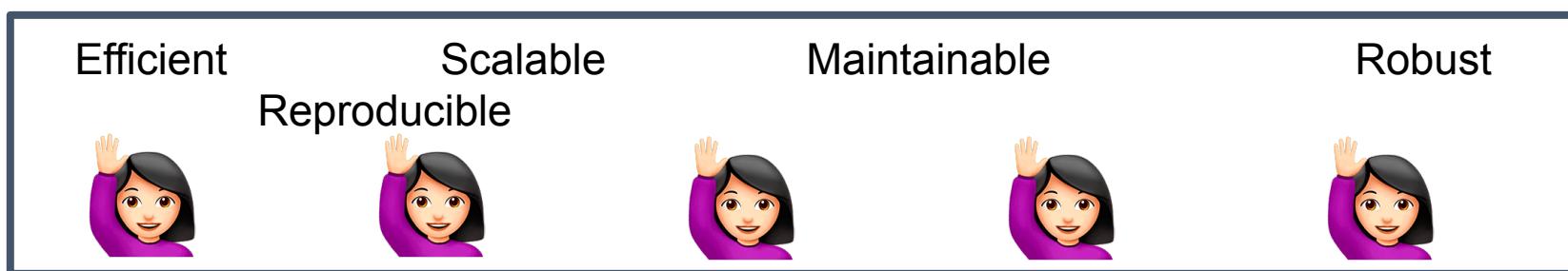
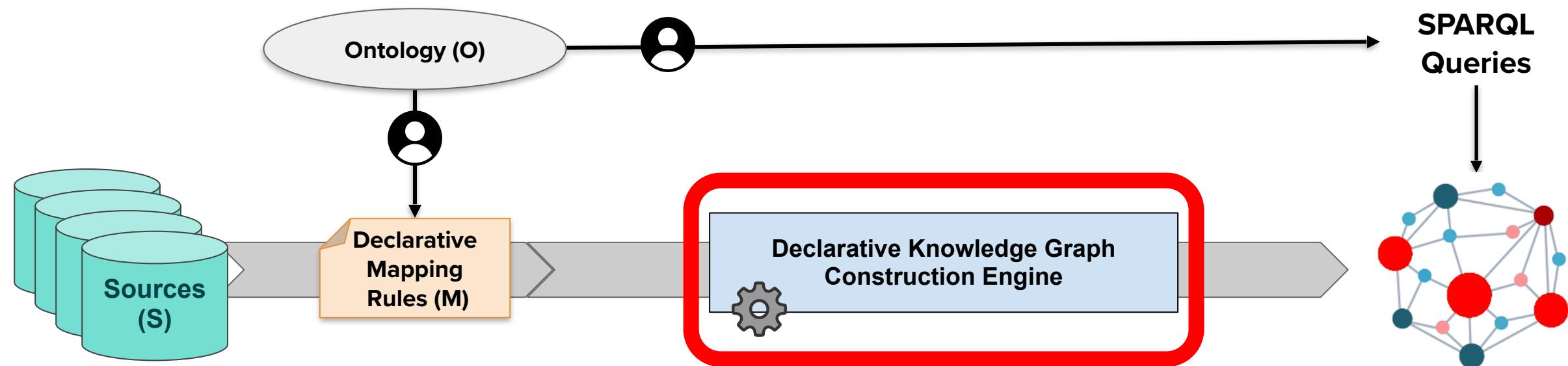
# Knowledge Graph Construction: A declarative approach

**Knowledge Graph Construction = Data Integration System (DIS) =  $\langle S, M, O \rangle$**

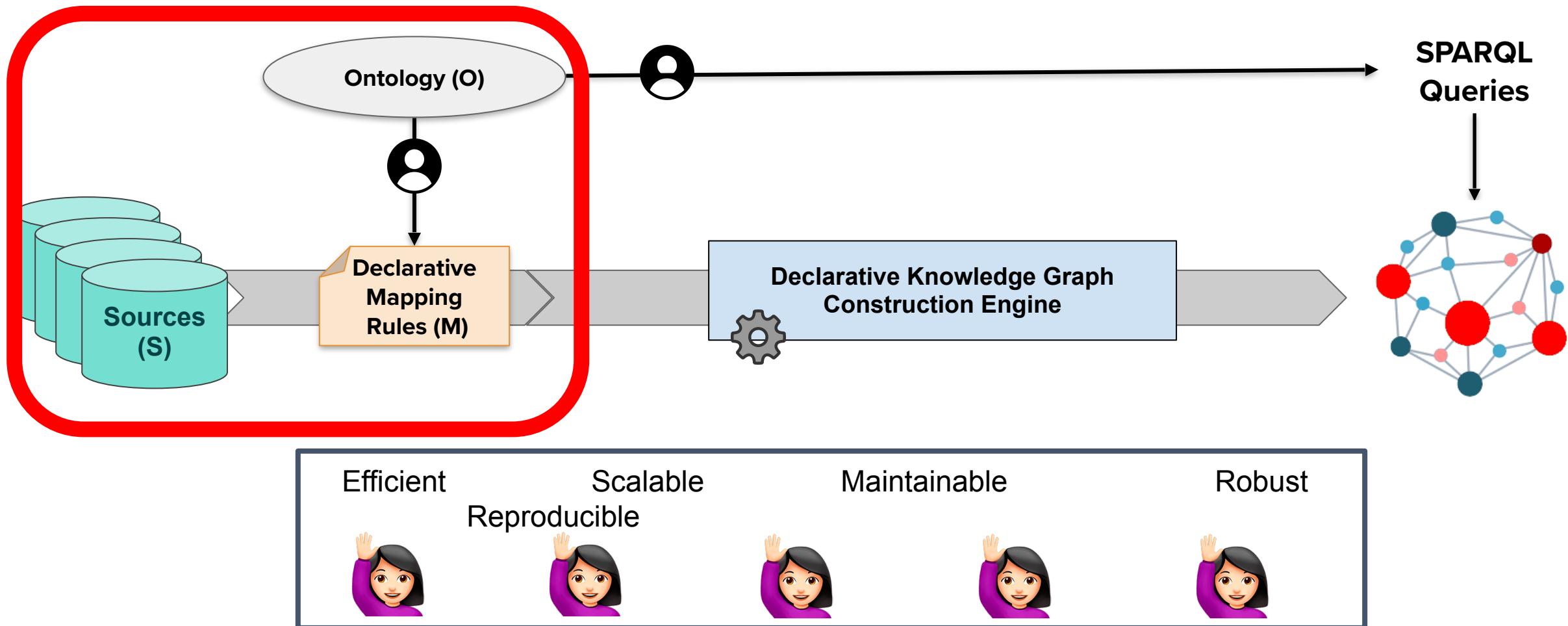


# Knowledge Graph Construction: A declarative approach

**Knowledge Graph Construction = Data Integration System (DIS) =  $\langle S, M, O \rangle$**

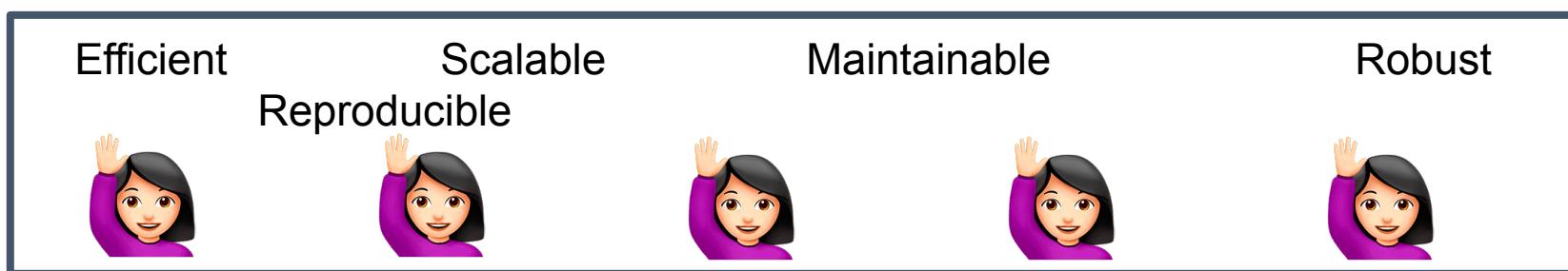
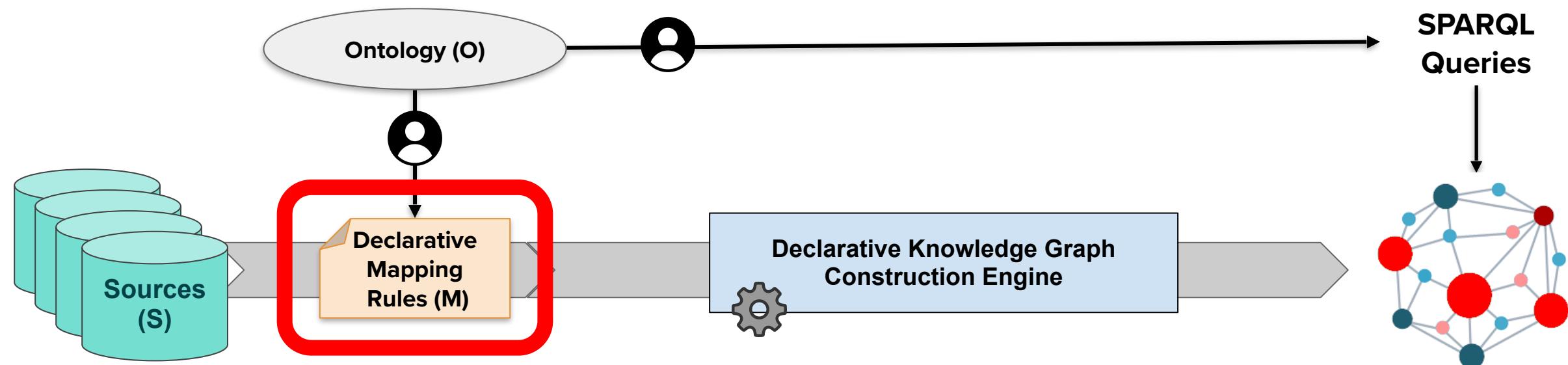


**Knowledge Graph Construction = Data Integration System (DIS) =  $\langle S, M, O \rangle$**



# Knowledge Graph Construction: A declarative approach

**Knowledge Graph Construction = Data Integration System (DIS) =  $\langle S, M, O \rangle$**



- The mapping language more widely used is **RML**
  - Extension of the W3C Recommendation R2RML for **heterogeneous data**
    - RDBs, CSV, JSON, XML
  - RDF syntax based
  - Old spec → <https://rml.io/specs/rml/>
  - Currently ongoing spec → <https://w3id.org/rml/portal>





materialisation

**DB2triples** (<https://github.com/antidot/db2triples>)  
**R2RML Parser** (<https://github.com/nkons/r2rml-parser>)  
**XSPARQL** (<http://xsparql.sourceforge.net/>)  
**R2RML-F** (<https://github.com/chrdebru/r2rml>)

homogeneous  
data sources

**R2RML**  
(*RDBs*)

**Morph-RDB** (<https://github.com/oeg-upm/morph-rdb>)  
**Ontop** (<https://github.com/ontop/ontop>)  
**TripleWave** (<https://github.com/streamreasoning/TripleWave>)  
**SparqlMap-M** (<https://github.com/tomatophantastico/sparqlmap>)  
**Morph-streams++** (<https://github.com/jpcik/morph-streams>)

virtualisation

**RMLMapper**: Java (<https://github.com/RMLio/rmlmapper-java>)  
**CARML**: Java (<https://github.com/carml/carml>)  
**RocketRML**: JavaScript (<https://github.com/semantifyit/RocketRML>)  
**RMLStreamer**: Flink (<https://github.com/RMLio/RMLStreamer>)  
**Chimera**: Camel (<https://github.com/cefriel/chimera>)  
**SDM-RDFizer**: heuristic-based planning  
(<https://github.com/SDM-TIB/SDM-RDFizer>)  
**FunMap**: function-free planning (<https://github.com/SDM-TIB/FunMap>)  
**MapSDI**: deduplication-based optimizations (<https://github.com/SDM-TIB/MapSDI>)  
**Morph-KGC**: mapping planning  
(<https://github.com/oeg-upm/morph-kgc>)

heterogeneous  
data sources

**RML**  
(*RDBs, NoSQL, RDF,  
CSV, XML, JSON, HTML*)

**Morph-xR2RML** (<https://github.com/frmichel/morph-xr2rml>)  
**Squerall** (<https://github.com/EIS-Bonn/Squerall>)  
**Ontario** (<https://github.com/SDM-TIB/Ontario/>)  
**Morph-CSV** (<https://github.com/oeg-upm/morph-csv>)

Slide adapted from Knowledge Graph Construction Tutorial  
@ ESWC 2022 (Anastasia Dimou)



materialisation

**DB2triples** (<https://github.com/antidot/db2triples>)  
**R2RML Parser** (<https://github.com/nkons/r2rml-parser>)  
**XSPARQL** (<http://xsparql.sourceforge.net/>)  
**R2RML-F** (<https://github.com/chrdebru/r2rml>)

homogeneous  
data sources

**R2RML**  
(RDBs)

**Morph-RDB** (<https://github.com/oeg-upm/morph-rdb>)  
**Ontop** (<https://github.com/ontop/ontop>)  
**TripleWave** (<https://github.com/streamreasoning/TripleWave>)  
**SparqlMap-M** (<https://github.com/tomatophantastico/sparqlmap>)  
**Morph-streams++** (<https://github.com/jpcik/morph-streams>)

virtualisation

materialisation

**RMLMapper**: Java (<https://github.com/RMLio/rmlmapper-java>)  
**CARML**: Java (<https://github.com/carml/carml>)  
**RocketRML**: JavaScript (<https://github.com/semantifyit/RocketRML>)  
**RMLStreamer**: Flink (<https://github.com/RMLio/RMLStreamer>)  
**Chimera**: Camel (<https://github.com/cefriel/chimera>)

**SDM-RDFizer**: heuristic-based planning  
(<https://github.com/SDM-TIB/SDM-RDFizer>)  
**FunMap**: function-free planning (<https://github.com/SDM-TIB/FunMap>)  
**MapSDI**: deduplication-based optimizations (<https://github.com/SDM-TIB/MapSDI>)  
**Morph-KGC**: mapping planning  
(<https://github.com/oeg-upm/morph-kgc>)

heterogeneous  
data sources

**RML**

(RDBs, NoSQL, RDF,  
CSV, XML, JSON, HTML)

**Morph-xR2RML** (<https://github.com/frmichel/morph-xr2rml>)  
**Squerall** (<https://github.com/EIS-Bonn/Squerall>)  
**Ontario** (<https://github.com/SDM-TIB/Ontario/>)  
**Morph-CSV** (<https://github.com/oeg-upm/morph-csv>)

Slide adapted from Knowledge Graph Construction Tutorial  
@ ESWC 2022 (Anastasia Dimou)



materialisation

**DB2triples** (<https://github.com/antidot/db2triples>)  
**R2RML Parser** (<https://github.com/nkons/r2rml-parser>)  
**XSPARQL** (<http://xsparql.sourceforge.net/>)  
**R2RML-F** (<https://github.com/chrdebru/r2rml>)

homogeneous  
data sources

**R2RML**  
(RDBs)

**Morph-RDB** (<https://github.com/oeg-upm/morph-rdb>)  
**Ontop** (<https://github.com/ontop/ontop>)  
**TripleWave** (<https://github.com/streamreasoning/TripleWave>)  
**SparqlMap-M** (<https://github.com/tomatophantastico/sparqlmap>)  
**Morph-streams++** (<https://github.com/jpcik/morph-streams>)

virtualisation

materialisation

**RMLMapper**: Java (<https://github.com/RMLio/rmlmapper-java>)  
**CARML**: Java (<https://github.com/carml/carml>)  
**RocketRML**: JavaScript (<https://github.com/semantifyit/RocketRML>)  
**RMLStreamer**: Flink (<https://github.com/RMLio/RMLStreamer>)  
**Chimera**: Camel (<https://github.com/cefriel/chimera>)

**SDM-RDFizer**: heuristic-based planning  
(<https://github.com/SDM-TIB/SDM-RDFizer>)  
**FunMap**: function-free planning (<https://github.com/SDM-TIB/FunMap>)  
**MapSDI**: deduplication-based optimizations (<https://github.com/SDM-TIB/MapSDI>)  
**Morph-KGC**: mapping planning  
(<https://github.com/oeg-upm/morph-kgc>)

heterogeneous  
data sources

**RML**

(RDBs, NoSQL, RDF,  
CSV, XML, JSON, HTML)

**Morph-xR2RML** (<https://github.com/frmichel/morph-xr2rml>)  
**Squerall** (<https://github.com/EIS-Bonn/Squerall>)  
**Ontario** (<https://github.com/SDM-TIB/Ontario/>)  
**Morph-CSV** (<https://github.com/oeg-upm/morph-csv>)

Slide adapted from Knowledge Graph Construction Tutorial  
@ ESWC 2022 (Anastasia Dimou)

```
@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>
.

<PERSON>
rml:logicalSource [
    rml:source "data/people.csv" ;
    rml:referenceFormulation ql:CSV ;
];

```



people.csv



ID	Name	Birthdate	SportID
1	Emily	08/02/90	2
2	Jonah	18/11/75	2





```

@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>

.

<PERSON>
  rml:logicalSource [
    rml:source "data/people.csv" ;
    rml:referenceFormulation ql:CSV ;
  ];
  rr:subjectMap [
    rr:class ex:Person;
    rr:template "http://ex.com/Person/{ID}" ;
  ];

```



<Person/1>  
ex:Person

<Person/2>  
ex:Person

people.csv



ID	Name	Birthdate	SportID
1	Emily	08/02/90	2
2	Jonah	18/11/75	2



```

@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>
.

<PERSON>
rml:logicalSource [
  rml:source "data/people.csv" ;
  rml:referenceFormulation ql:CSV ;
];
rr:subjectMap [
rr:class ex:Person;
  rr:template "http://ex.com/Person/{ID}" ;
];

```



<Person/1>  
ex:Person

<Person/2>  
ex:Person

people.csv



ID	Name	Birthdate	SportID
1	Emily	08/02/90	2
2	Jonah	18/11/75	2

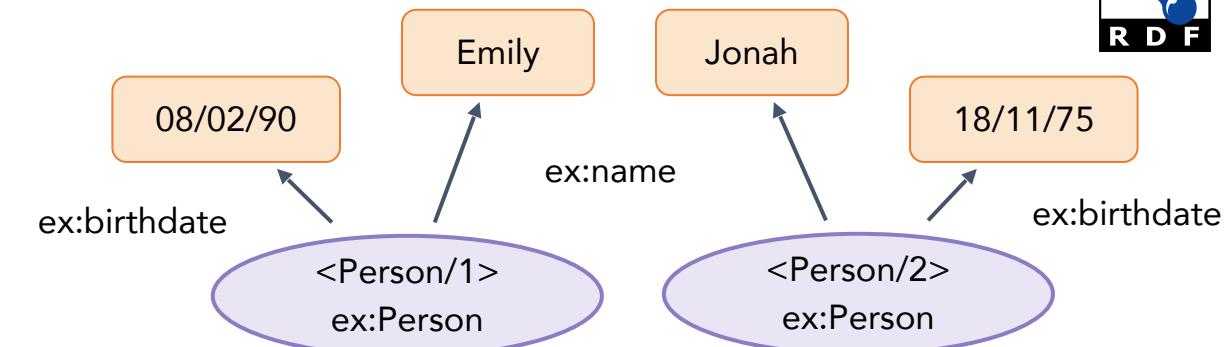


```

@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>
.

<PERSON>
  rml:logicalSource [
    rml:source "data/people.csv" ;
    rml:referenceFormulation ql:CSV ;
  ];
  rr:subjectMap [
    rr:class ex:Person;
    rr:template "http://ex.com/Person/{ID}" ;
  ];
  rr:predicateObjectMap [
    rr:predicateMap [ rr:constant ex:name ];
    rr:objectMap [ rml:reference "Name" ];
  ];
  rr:predicateObjectMap [
    rr:predicateMap [ rr:constant ex:birthdate ];
    rr:objectMap [ rml:reference "Birthdate" ];
  ];

```



people.csv			
ID	Name	Birthdate	SportID
1	Emily	08/02/90	2
2	Jonah	18/11/75	2

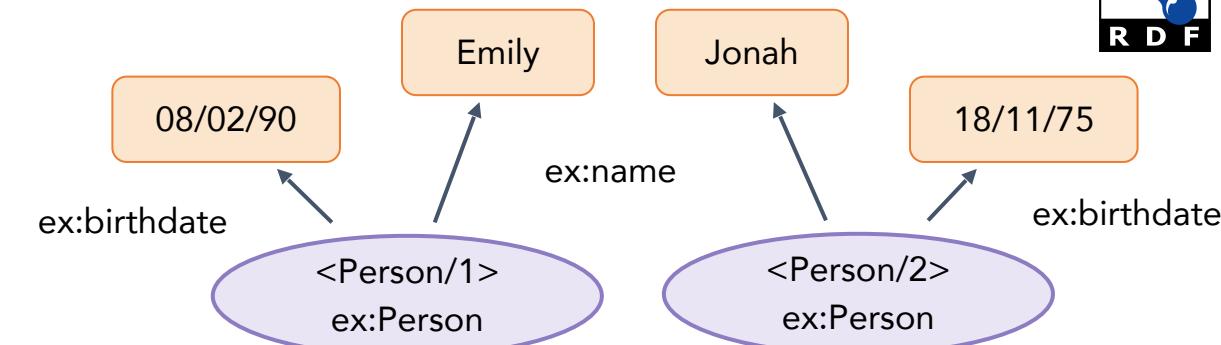


```

@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>
.

<PERSON>
  rml:logicalSource [
    rml:source "data/people.csv" ;
    rml:referenceFormulation ql:CSV ;
  ];
  rr:subjectMap [
    rr:class ex:Person;
    rr:template "http://ex.com/Person/{ID}" ;
  ];
  rr:predicateObjectMap [ rr:predicateMap [ rr:constant ex:name ];
    rr:objectMap [ rml:reference "Name" ] ;
  ];
  rr:predicateObjectMap [ rr:predicateMap [ rr:constant ex:birthdate ];
    rr:objectMap [ rml:reference "Birthdate" ] ;
  ];

```



people.csv 

ID	Name	Birthdate	SportID
1	Emily	08/02/90	2
2	Jonah	18/11/75	2

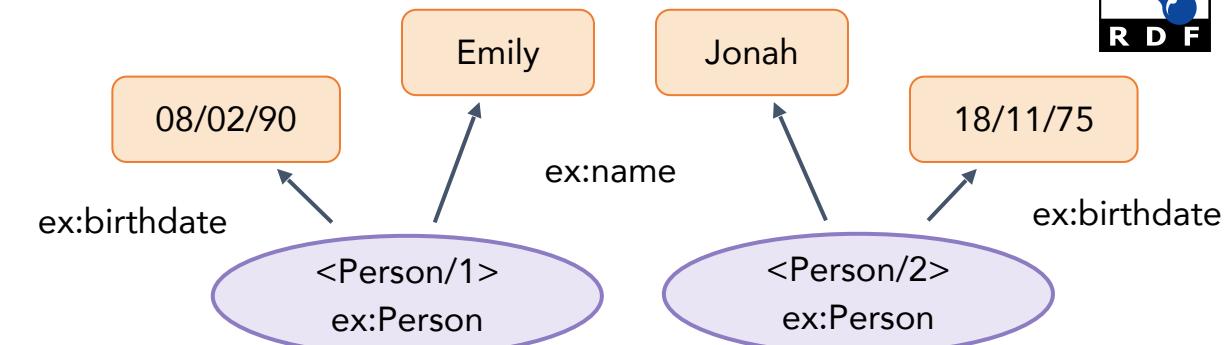


```

@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>
.

<PERSON>
  rml:logicalSource [
    rml:source "data/people.csv" ;
    rml:referenceFormulation ql:CSV ;
  ];
  rr:subjectMap [
    rr:class ex:Person;
    rr:template "http://ex.com/Person/{ID}" ;
  ];
  rr:predicateObjectMap [
    rr:predicateMap [ rr:constant ex:name ];
    rr:objectMap [ rml:reference "Name" ];
  ];
  rr:predicateObjectMap [
    rr:predicateMap [ rr:constant ex:birthdate ];
    rr:objectMap [ rml:reference "Birthdate" ];
  ];

```



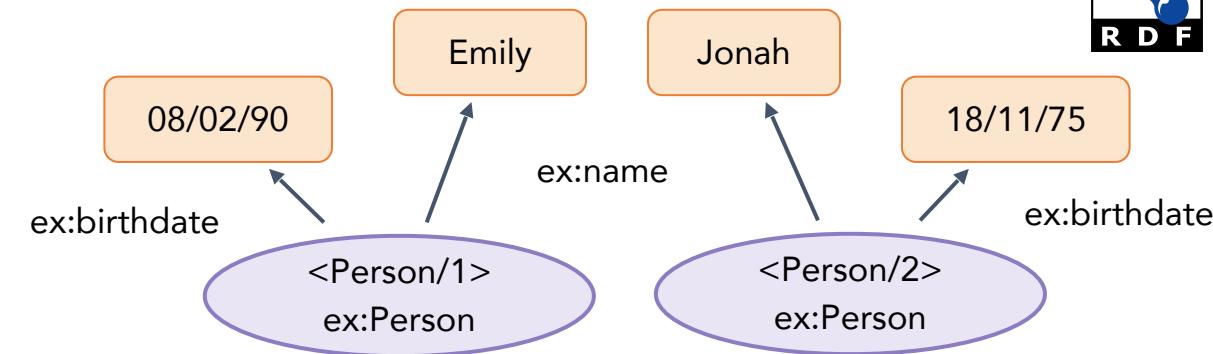
people.csv			
ID	Name	Birthdate	SportID
1	Emily	08/02/90	2
2	Jonah	18/11/75	2

```

@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>
.

<PERSON>
rml:logicalSource [
  rml:source "data/sports.csv" ;
  rml:referenceFormulation ql:CSV ;
  rml:referenceFrame "Name" ;
];
rr:subjectMap [
  rr:class ex:Person ;
  rr:template "http://ex.com/Person/{ID}" ;
];
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant ex:name ];
  rr:objectMap [ rml:reference "Name" ];
];
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant ex:birthdate ];
  rr:objectMap [ rml:reference "Birthdate" ];
];

```



people.csv				csv
ID	Name	Birthdate	SportID	
1	Emily	08/02/90	2	
2	Jonah	18/11/75	2	

sports.csv		csv
ID	Sport	
1	Ice Skating	
2	Rugby	

```

@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>
.

<PERSON>
rml:logicalSource [
  rml:source "data/sports.csv" ;
  rml:referenceFormulation ql:CSV ;
];
rr:subjectMap [
  rr:class ex:Sport;
  rr:template "http://ex.com/Sport/{ID}" ;
  rr:template "http://ex.com/Person/{ID}" ;
];
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant ex:name ];
  rr:objectMap [ rml:reference "Name" ];
];
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant ex:birthdate ];
  rr:objectMap [ rml:reference "Birthdate" ];
];

```

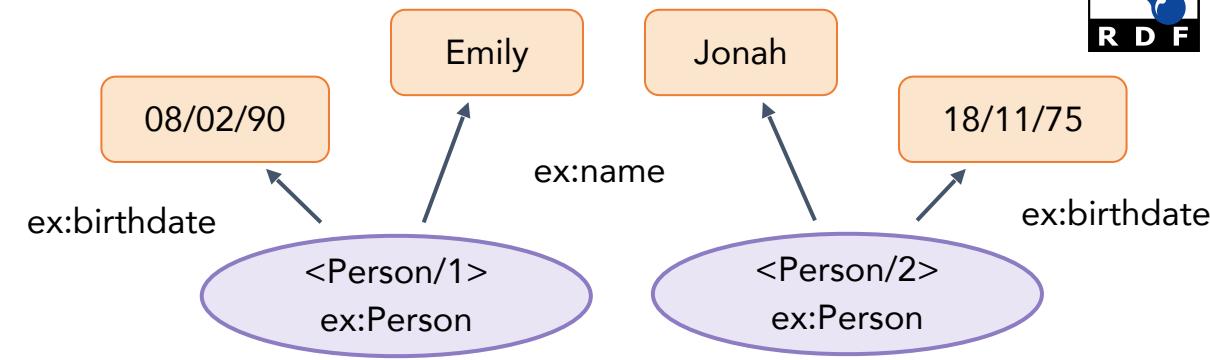


```

@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>
.

<SPORT>
rml:logicalSource [
  rml:source "data/sports.csv" ;
  rml:referenceFormulation ql:CSV ;
];
rr:subjectMap [
  rr:class ex:Sport;
  rr:template "http://ex.com/Sport/{ID}" ;
];
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant ex:name ];
  rr:objectMap [ rml:reference "Name" ];
];
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant ex:birthdate ];
  rr:objectMap [ rml:reference "Birthdate" ];
];

```



people.csv				csv
ID	Name	Birthdate	SportID	
1	Emily	08/02/90	2	
2	Jonah	18/11/75	2	

sports.csv		csv
ID	Sport	
1	Ice Skating	
2	Rugby	

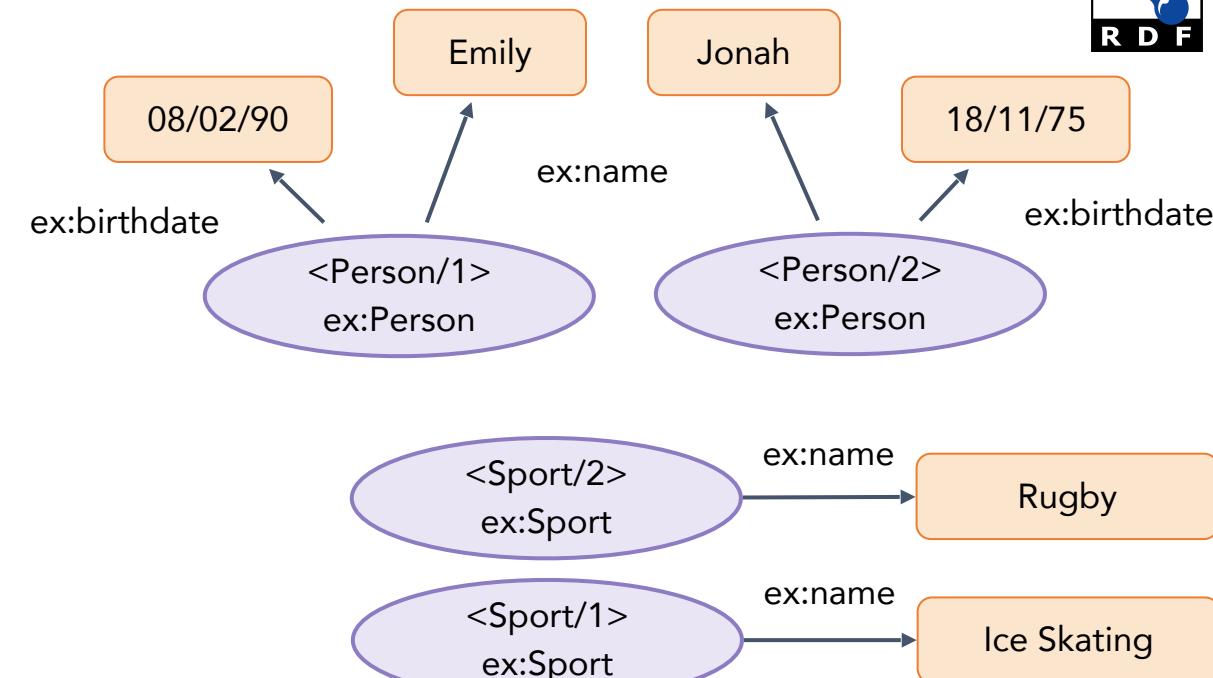


```

@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>
.

<SPORT>
rml:logicalSource [
    rml:source "data/sports.csv" ;
    rml:referenceFormulation ql:CSV ;
];
rr:subjectMap [
    rr:class ex:Sport;
    rr:template "http://ex.com/Sport/{ID}" ;
];
rr:predicateObjectMap [
    rr:predicateMap [ rr:constant ex:name ];
    rr:predicateMap [ rr:objectMap ex:name; rml:reference "Sport" ];
    rr:objectMap [ rml:reference "Name" ];
];
rr:predicateObjectMap [
    rr:predicateMap [ rr:constant ex:birthdate ];
    rr:objectMap [ rml:reference "Birthdate" ];
];

```



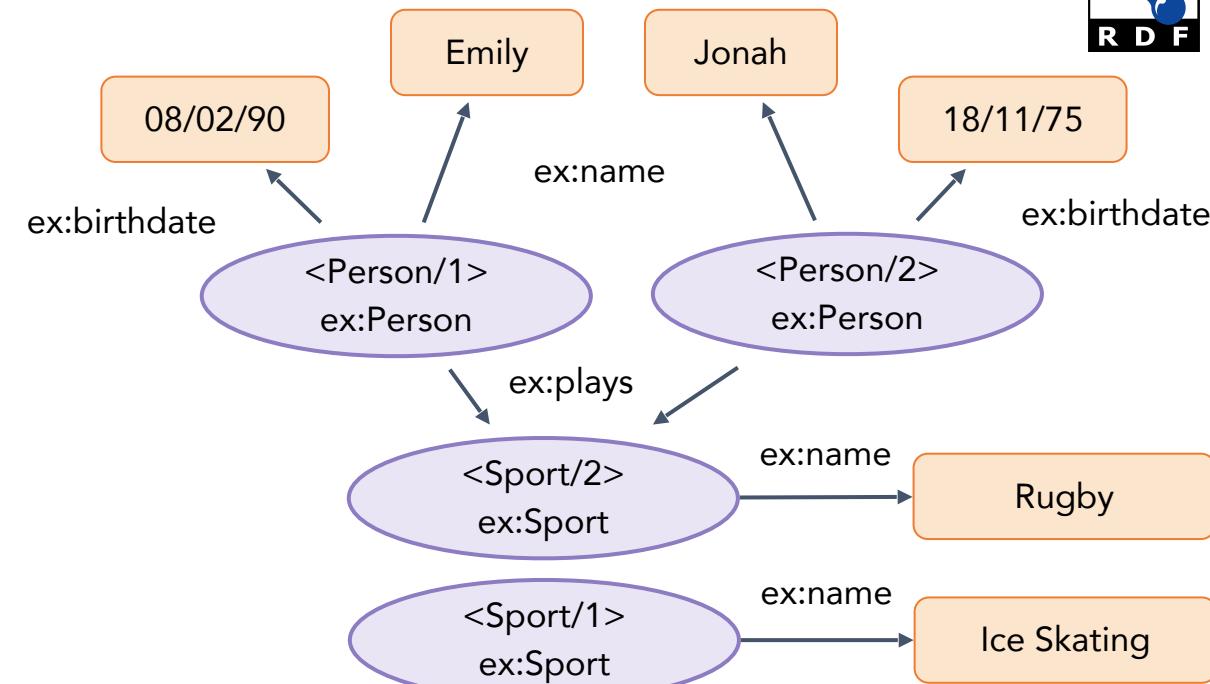
people.csv				sports.csv	
ID	Name	Birthdate	SportID	ID	Sport
1	Emily	08/02/90	2	1	Ice Skating
2	Jonah	18/11/75	2	2	Rugby

```

@prefix ex: <http://ex.com/>
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#>
.

<PERSON>
rml:logicalSource [
  rml:source "data/people.csv" ;
  rml:referenceFormulation ql:CSV ;
];
rr:subjectMap [
  rr:class ex:Person;
  rr:template "http://ex.com/Person/{ID}" ;
];
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant ex:name ];
  rr:objectMap [ rml:reference "Name" ; reference "Sport" ];
];
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant ex:birthdate ];
  rr:objectMap [ rml:reference "Birthdate" ];
];
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant ex:sport ];
  rr:objectMap [ rr:parentTriplesMap <SPORT> ;
    rml:joinCondition[ rr:child "SportID" ; rr:parent "ID" ];
    rml:joinCondition[ rr:child "SportID" ; rr:parent "ID" ];
    rml:joinCondition[ rr:child "SportID" ; rr:parent "ID" ];
  ];
];

```



people.csv				sports.csv	
ID	Name	Birthdate	SportID	ID	Sport
1	Emily	08/02/90	2	1	Ice Skating
2	Jonah	18/11/75	2	2	Rugby

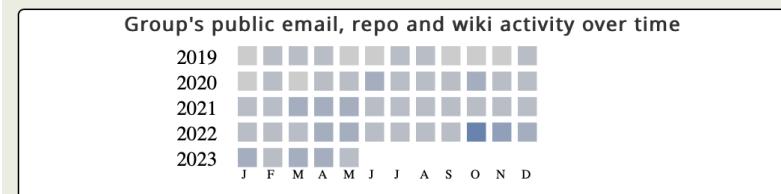


# W3C Community Group - Knowledge Graph Construction

## KNOWLEDGE GRAPH CONSTRUCTION COMMUNITY GROUP

The overall goal of this community group is to support its participants into developing better methods for Knowledge Graphs construction. The Community Group will (i) study current Knowledge Graph construction methods and implementations, (ii) identify the corresponding requirements and issues that hinder broader Knowledge Graph construction, (iii) discuss use cases, (iv) formulate guidelines, best practices and test cases for Knowledge Graph construction, (v) develop methods, resources and tools for evaluating Knowledge Graphs construction, and in general (vi) continue the development of the W3C-recommended R2RML language beyond relational databases. The proposed Community Group could be instrumental to advance research, increase the level of education and awareness and enable learning and participation with respect to Knowledge Graph construction.

### [kg-construct](#)



Note: Community Groups are proposed and run by the community. Although W3C hosts these conversations, the groups do not necessarily represent the views of the W3C Membership or staff.

### No Reports Yet Published

Chairs, when logged in, may publish draft and final reports. Please see [report requirements](#).

[PUBLISH REPORTS](#)

### biweekly meetings

#### Tools for this group

- Mailing List
- IRC
- Github repositories
- RSS
- Contact This Group

#### Get involved

Anyone may join this Community Group. All participants in the group have signed the [W3C Community Contributor License Agreement](#).

[JOIN OR LEAVE THIS GROUP](#)

Anastasia Dimou

David Chaves-Fraga

Alessandro Negro

Chairs  
→

#### Participants (162)



162 participants (~25 active)

Bi-weekly meetings



<http://w3id.org/kg-construct>



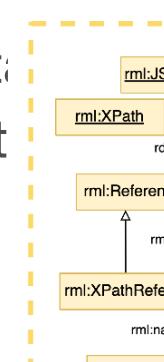
<http://github.com/kg-construct>



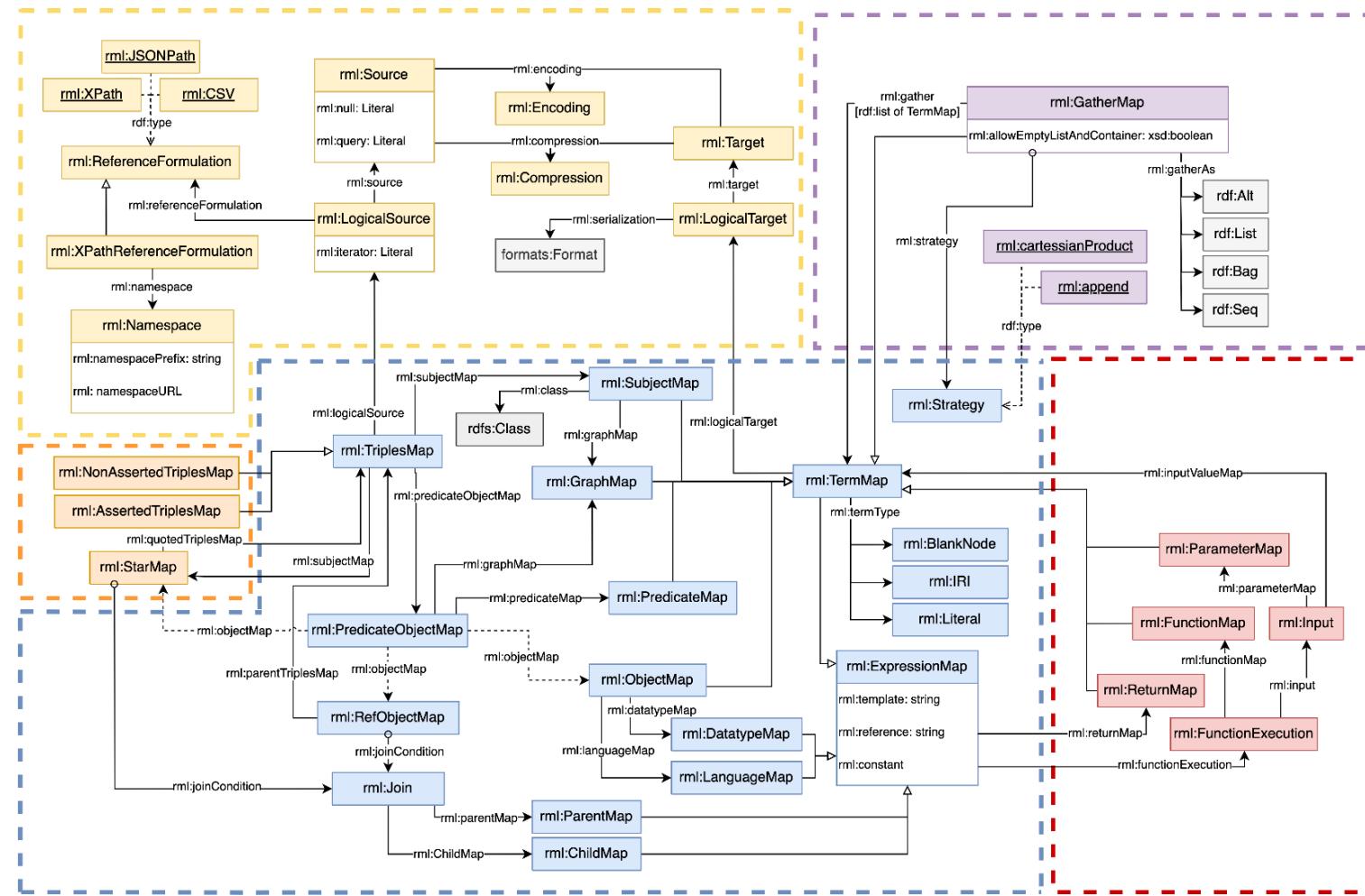
- Five on-going specs:
  - RML-Core: Schema transformations
  - RML-IO: Source and target
  - RML-CC: Collection and containers
  - RML-FNML: Data transformation functions
  - RML-star: RDF-star
- Modular approach
- Unification of prefixes





- Five on-going specs:
    - RML-Core: Schema transformations
    - RML-IO: Source and target
    - RML-CC: Collection and context
    - RML-FNML: Data transformation
    - RML-star: RDF-star
  - Modular approach
  - Unification of prefixes

```
graph TD; JS[rml:JS] --- XPAth[rml:XPath]; XPAth --- Reference[rml:Reference]; XPAth --- XPAthRef[rml:XPathRef]; Reference --- XPAthRef; Reference --- Namespaces[rml:Namespaces]; XPAthRef --- Namespaces;
```

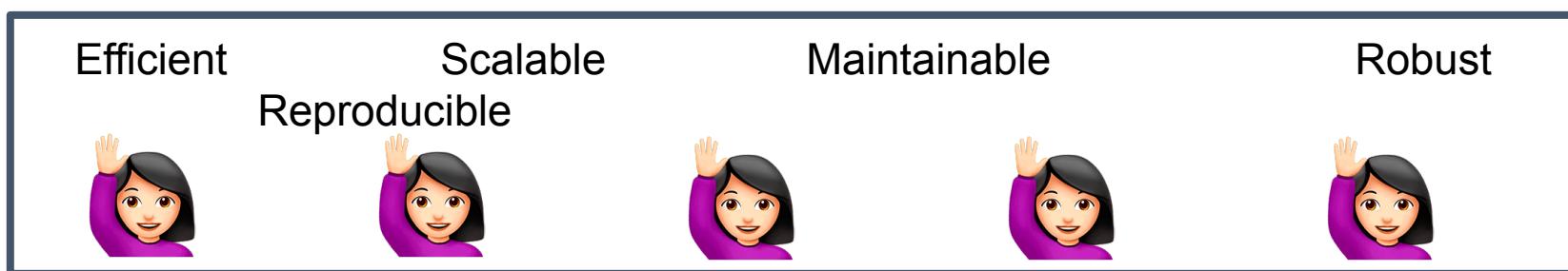
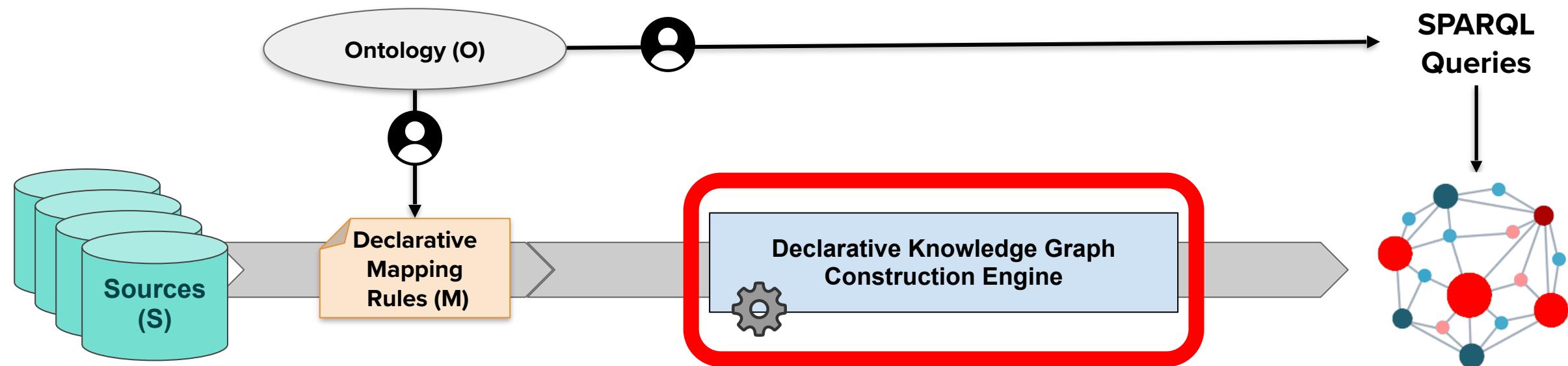


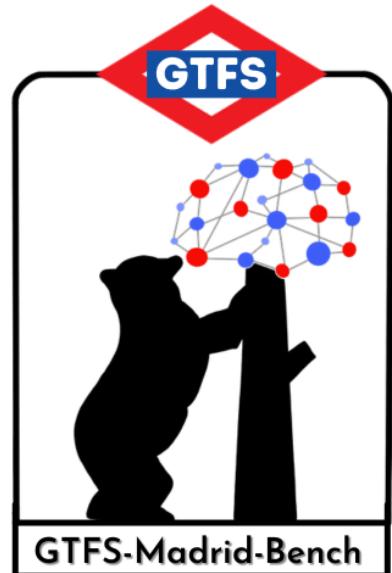
 Iglesias et al. The RML Ontology: A Community-Driven Modular Redesign After a Decade of Experience in Mapping Heterogeneous Data to RDF (*Under Review*)

 Iglesias-Molina, A., Chaves-Fraga, D., Dasoulas, I., & Dimou, A. Human-Friendly Knowledge Graph Construction: Which one do you chose?. *ICWE 2023*

# Knowledge Graph Construction: A declarative approach

**Knowledge Graph Construction = Data Integration System (DIS) =  $\langle S, M, O \rangle$**



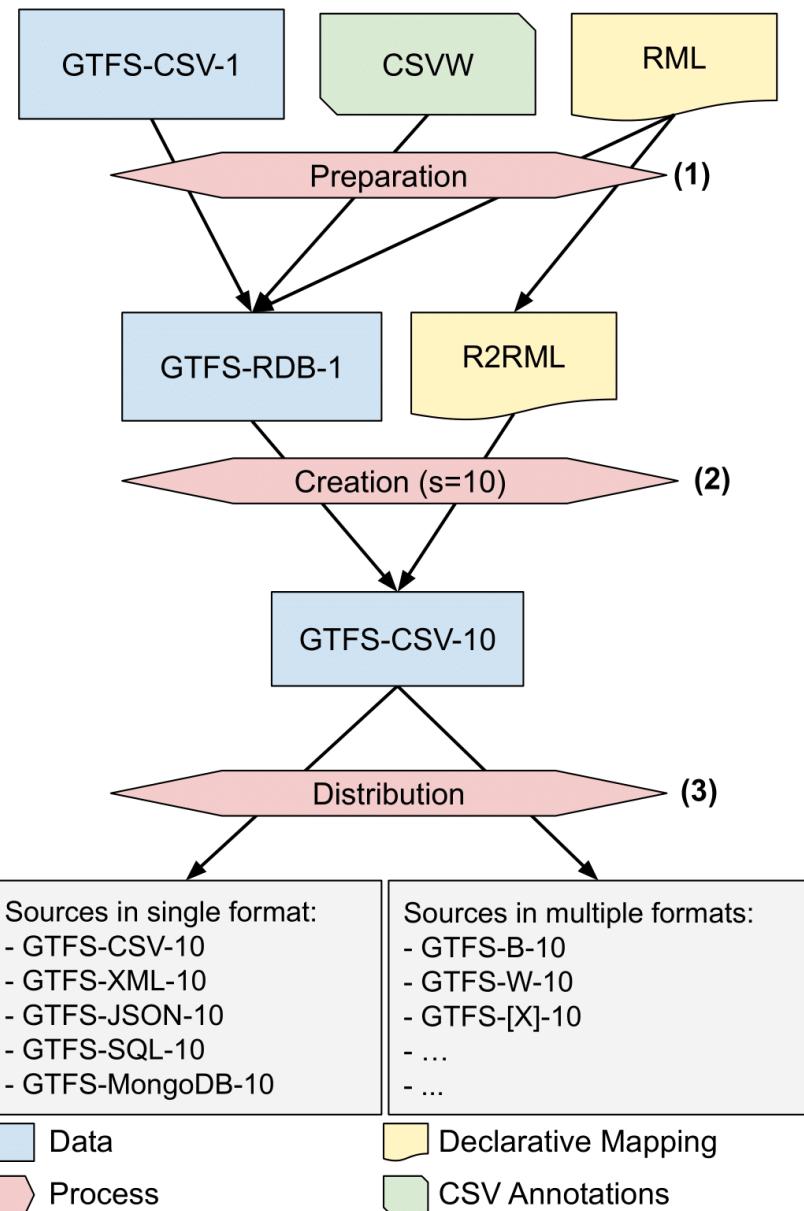


## A comprehensive benchmark for (virtual) knowledge graph access

- Query translation and data materialization over heterogeneous data
- Transport Domain (GTFS)
- Unified evaluation framework for heterogeneous KGC engines
- Used by most of the tools from the state of the art
- Highly influenced by BSBM (queries) and NPD (data generation)



**Chaves-Fraga, D., Priyatna, F., Cimmino, A., Toledo, J., Ruckhaus, E., & Corcho, O. (2020). GTFS-Madrid-Bench: A benchmark for virtual knowledge graph access in the transport domain. *Journal of Web Semantics* (Q2).**



## Dataset:

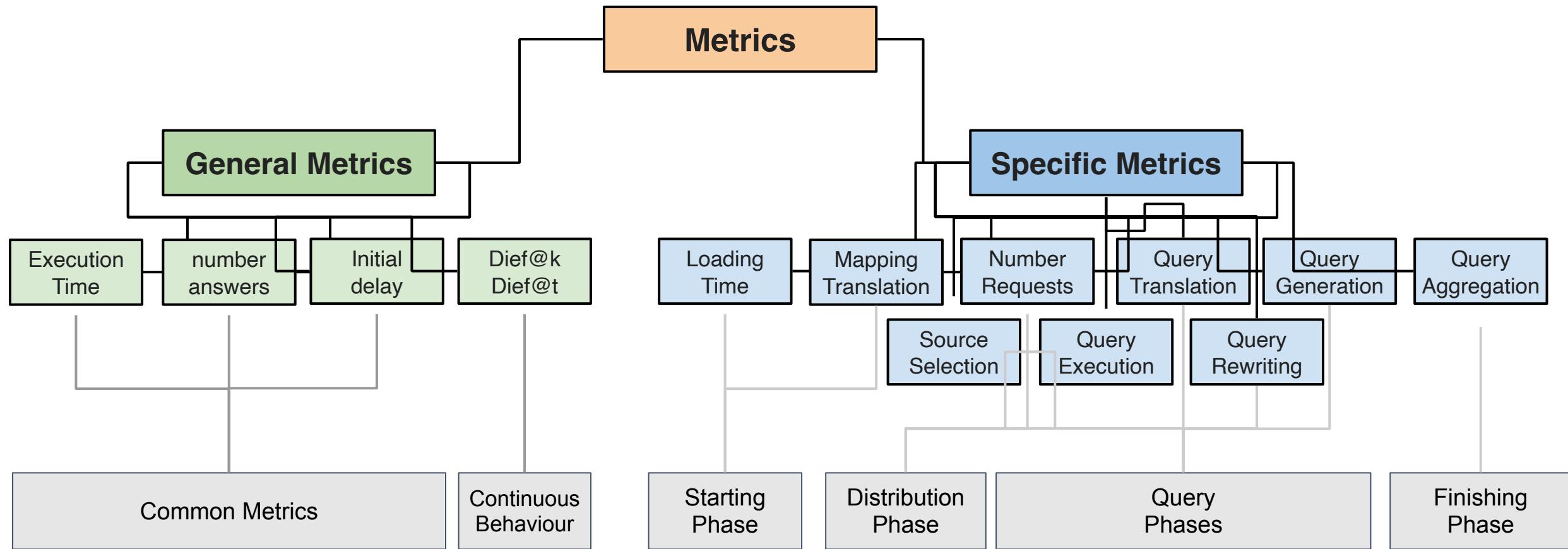
- Morph-CSV to generate GTFS-RDB
  - VIG to scale-up
  - Distribution based on user preferences

## Queries:

- 18 queries covering different configurations and SPARQL operators
  - Aligned with user stories in Madrid's transport domain
  - Triple patterns: from 3 to 15; Sources: 1 to 5
  - Single and chain star-shaped groups

## Mappings:

- 10 Sources, 12 TriplesMap (12 Classes), 71 POM (70 P), 60 SOM, 11 ROM
  - 1 R2RML, 5 RML (YARRRML serialization), 1 xR2RML, 1 CSVW annotations + RML-Mapping generator





*TO (TimeOut), W (wrong n<sup>o</sup> results), E (error executing the query)*

Dataset	Processor		Query																	
	Cache	Name	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	q11	q12	q13	q14	q15	q16	q17	q18
GTFS-SQL-1	Warm	Morph-RDB	5.85	02.07	E	1.82	W	1.86	1.97	E	26.02	1.80	E	1.81	2.06	W	1.89	E	2.11	E
		Ontario	18.02	E	TO	E	E	E	E	W	E	E	E	E	E	W	E	E	E	
	Cold	Morph-RDB	7.14	2.65	E	2.42	W	2.36	2.43	E	28.65	2.38	E	2.41	2.69	W	2.58	E	2.68	E
		Ontop	8.37	05.04	5.18	E	W	E	W	E	16.56	E	E	E	05.06	W	5.10	W	5.00	W
GTFS-MongoDB-1	Warm	Morph-xR2RML	W	W	W	W	W	W	W	W	W	W	W	W	W	28.67	W	W	6.52	W
	Cold	Morph-xR2RML	W	W	W	W	W	W	W	W	W	W	W	W	W	28.17	W	W	6.96	W
GTFS-CSV-1	Cold	Morph-RDB	6.94	03.04	E	2.78	E	2.78	TO	E	TO	2.97	E	6.23	3.97	E	E	E	3.14	W
		Morph-CSV	15.11	10.88	E	10.72	E	9.95	10.84	E	40.90	10.70	E	11.60	11.82	E	E	E	11.48	W
		Ontario	W	E	17.34	E	E	E	E	W	E	E	E	E	E	W	E	E	E	
GTFS-XML-1	Cold	Ontario	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	
GTFS-JSON-1	Cold	Ontario	18.04	E	17.14	E	E	E	E	W	E	E	E	E	E	W	E	E	E	
GTFS-MINEXT-1	Cold	Ontario	W	E	E	E	E	E	E	W	E	E	E	E	E	W	E	E	E	
GTFS-MAXEXT-1	Cold	Ontario	W	E	17.14	E	E	E	E	W	E	E	E	E	E	W	E	E	E	

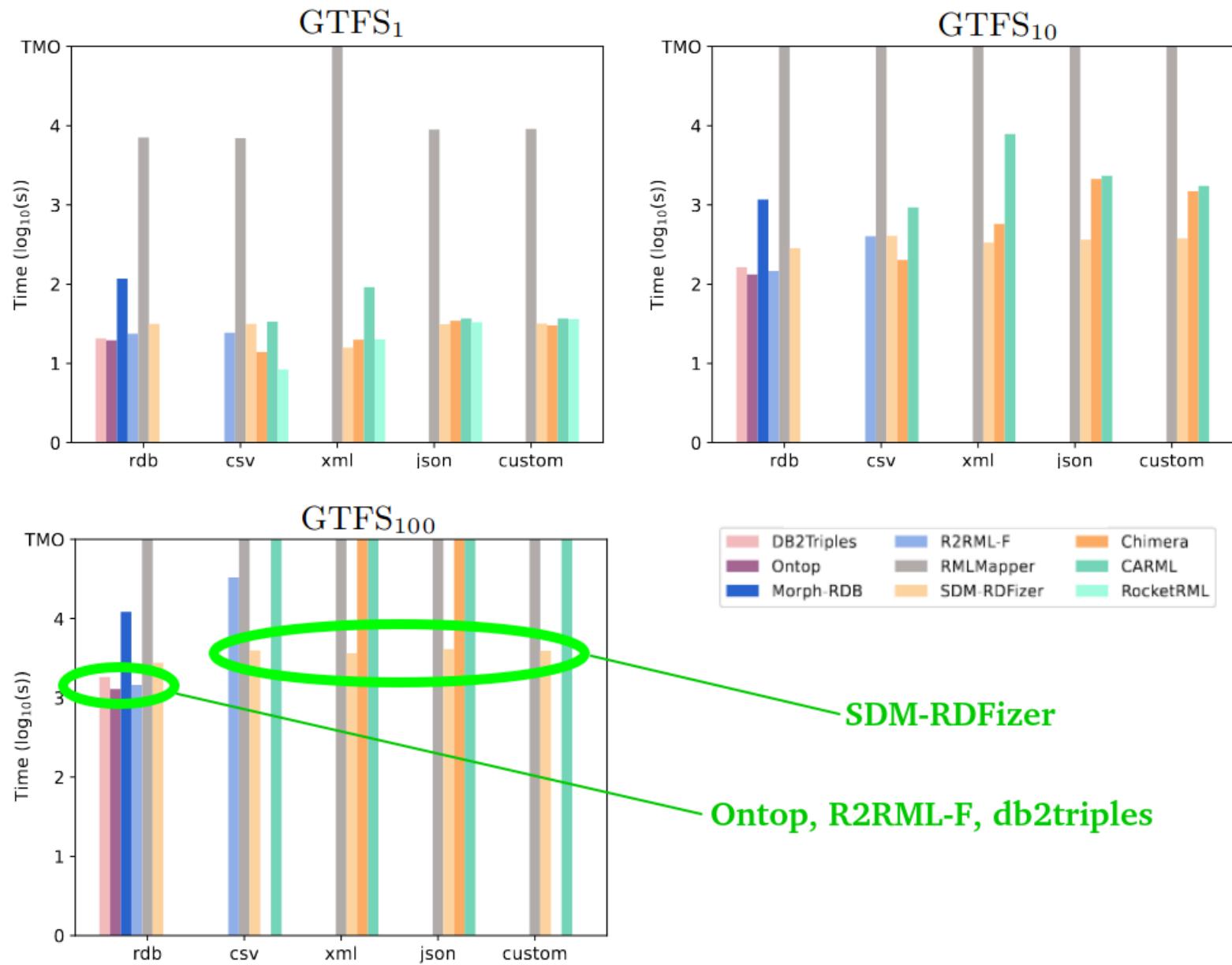
- Only the SPARQL-to-SQL engines provide an acceptable support for SPARQL operators
- Virtual KGC proposals beyond relational databases are not mature enough and more research is needed
- The problem of translating SPARQL queries for querying raw data (CSV, JSON, XML) should not be understood as a technical case

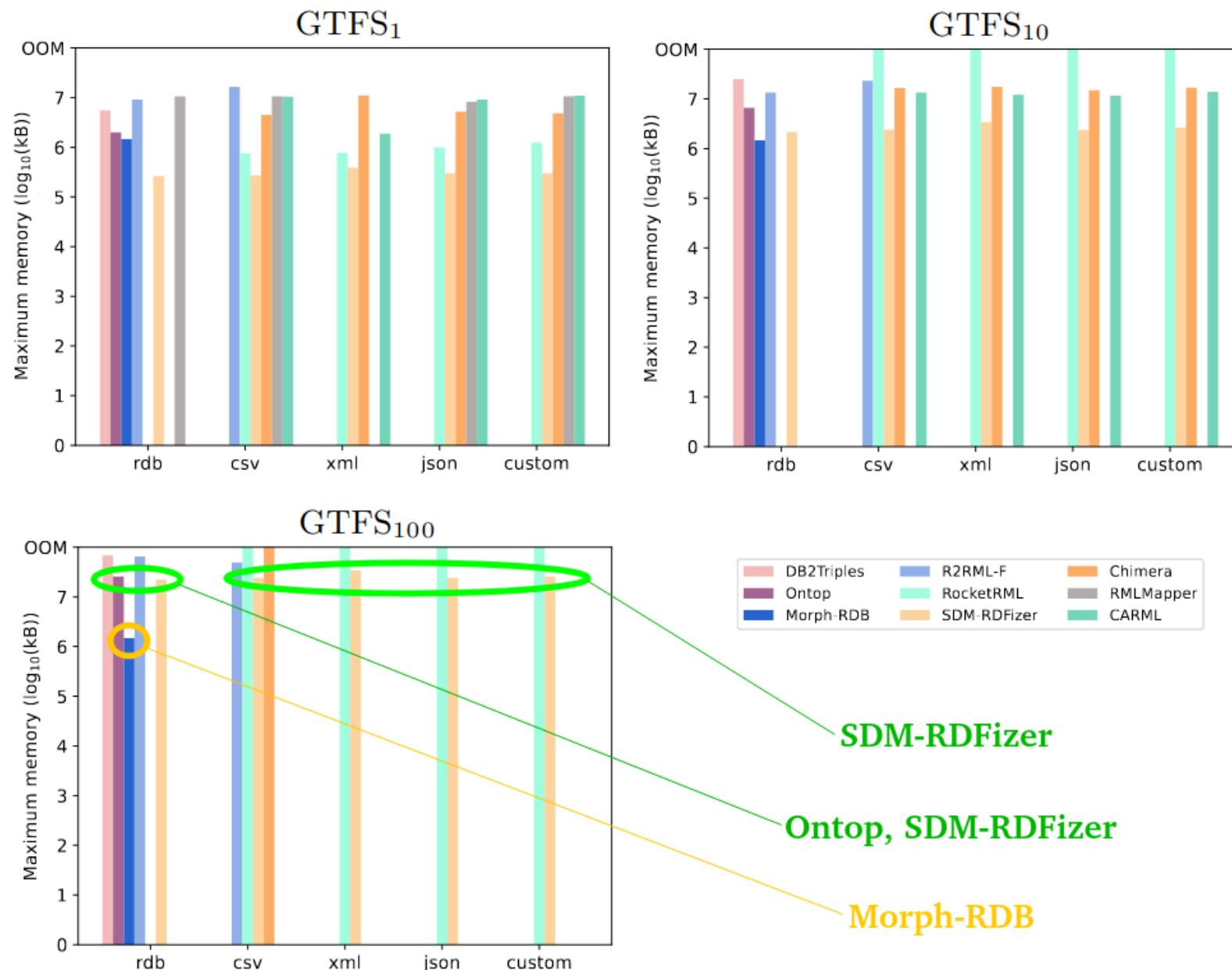


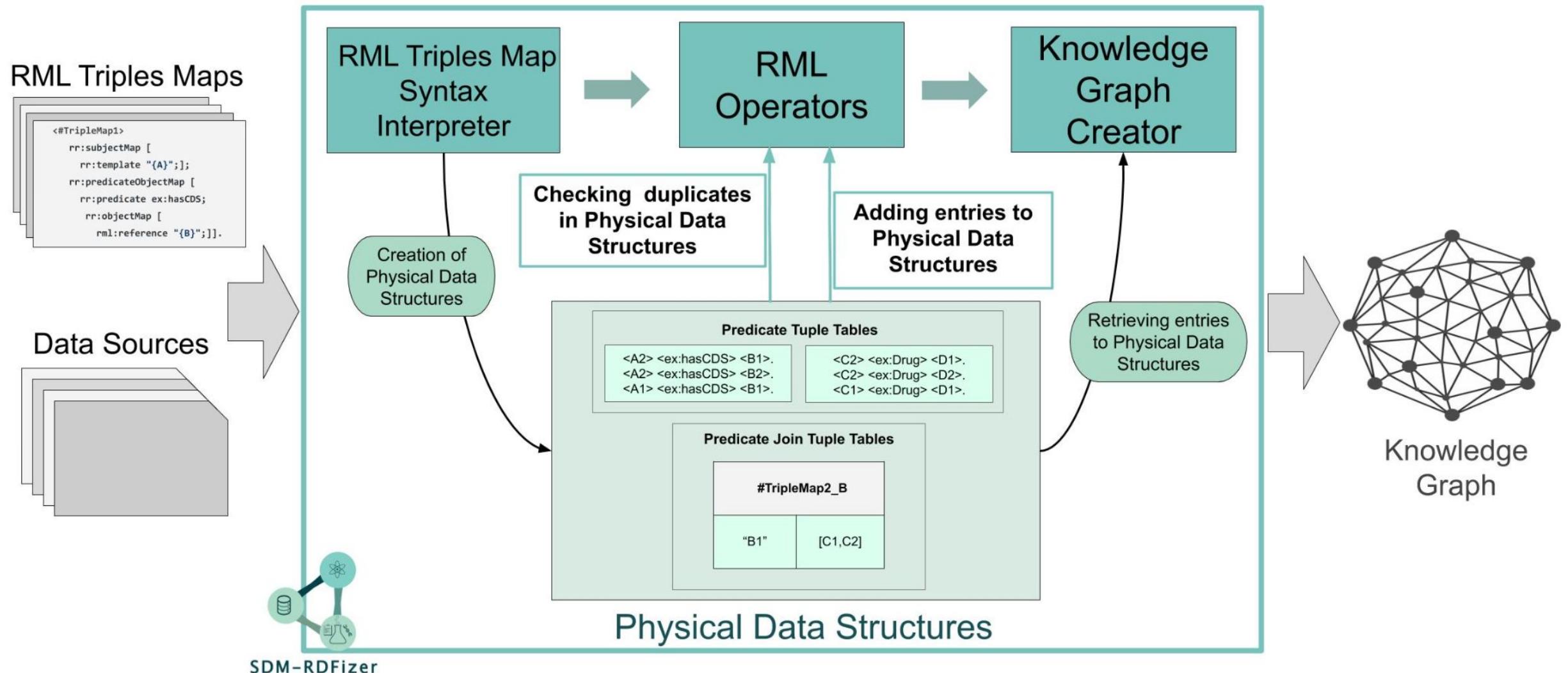
*TO (TimeOut), W (wrong n° results), E (error executing the query)*

Dataset	Processor		Query																	
	Cache	Name	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	q11	q12	q13	q14	q15	q16	q17	q18
GTFS-SQL-1	Warm	Morph-RDB	5.85	02.07	E	1.82	W	1.86	1.97	E	26.02	1.80	E	1.81	2.06	W	1.89	E	2.11	E
		Ontario	18.02	E	TO	E	E	E	E	W	E	E	E	E	E	W	E	E	E	
	Cold	Morph-RDB	7.14	2.65	E	2.42	W	2.36	2.43	E	28.65	2.38	E	2.41	2.69	W	2.58	E	2.68	E
		Ontop	8.37	05.04	5.18	E	W	E	W	E	16.56	E	E	E	05.06	W	5.10	W	5.00	W
GTFS-MongoDB-1	Warm	Morph-xR2RML	W	W	W	W	W	W	W	W	W	W	W	W	W	28.67	W	W	6.52	W
	Cold	Morph-xR2RML	W	W	W	W	W	W	W	W	W	W	W	W	W	28.17	W	W	6.96	W
GTFS-CSV-1	Cold	Morph-RDB	6.94	03.04	E	2.78	E	2.78	TO	E	TO	2.97	E	6.23	3.97	E	E	E	3.14	W
		Morph-CSV	15.11	10.88	E	10.72	E	9.95	10.84	E	40.90	10.70	E	11.60	11.82	E	E	E	11.48	W
		Ontario	W	E	17.04	E	E	E	E	W	E	E	E	E	E	W	E	E	E	
GTFS-XML-1	Cold	Ontario		E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E
GTFS-JSON-1	Cold	Ontario		18.04	E	17.14	E	E	E	W	E	E	E	E	E	W	E	E	E	
GTFS-MINEXT-1	Cold	Ontario		W	E	E	E	E	E	W	E	E	E	E	E	W	E	E	E	
GTFS-MAXEXT-1	Cold	Ontario		W	E	17.14	E	E	E	W	E	E	E	E	E	W	E	E	E	

- Only the SPARQL-to-SQL engines provide an acceptable support for SPARQL operators
- Virtual KGC proposals beyond relational databases are not mature enough and more research is needed
- The problem of translating SPARQL queries for querying raw data (CSV, JSON, XML) should not be understood as a technical case

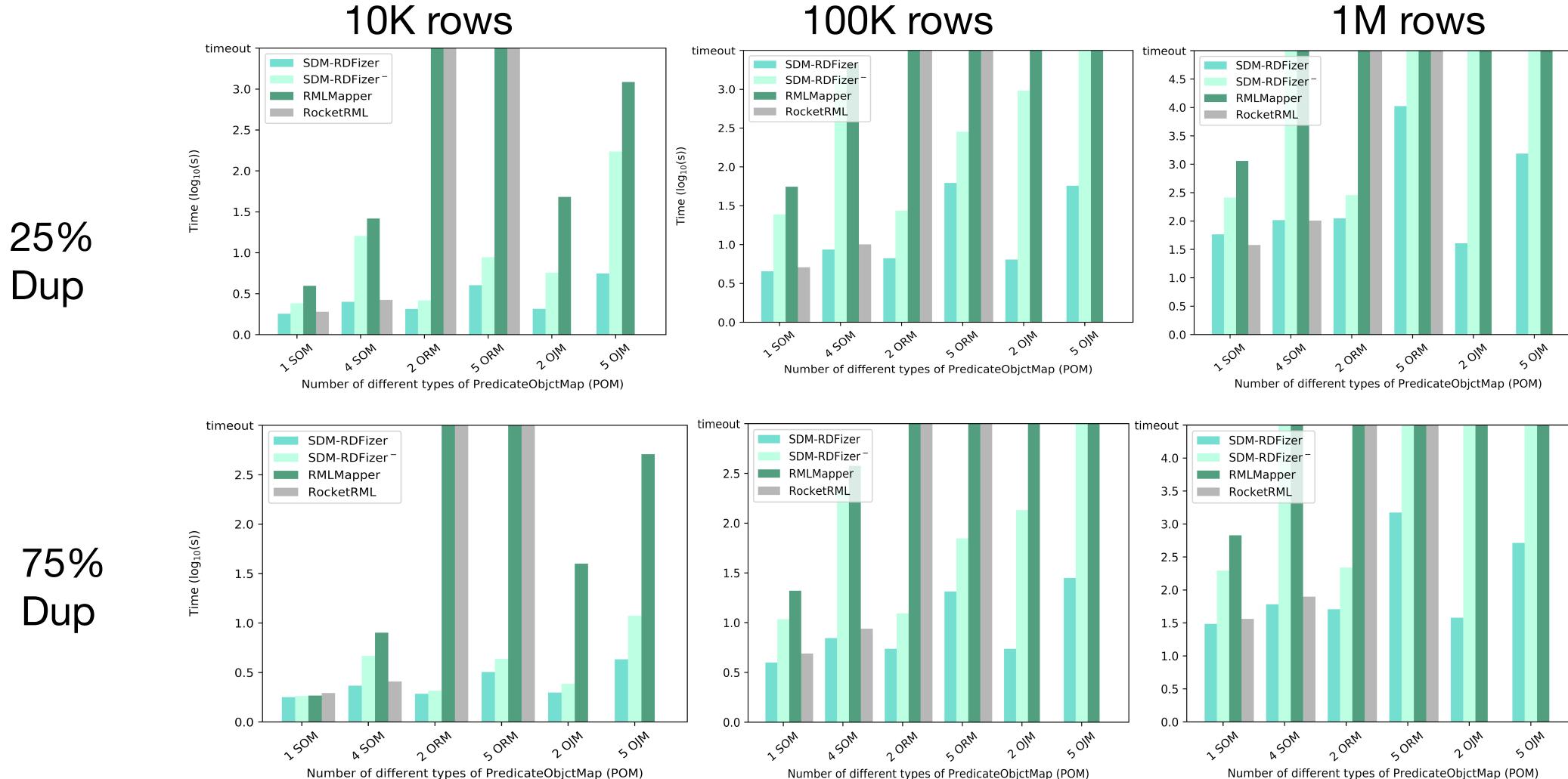




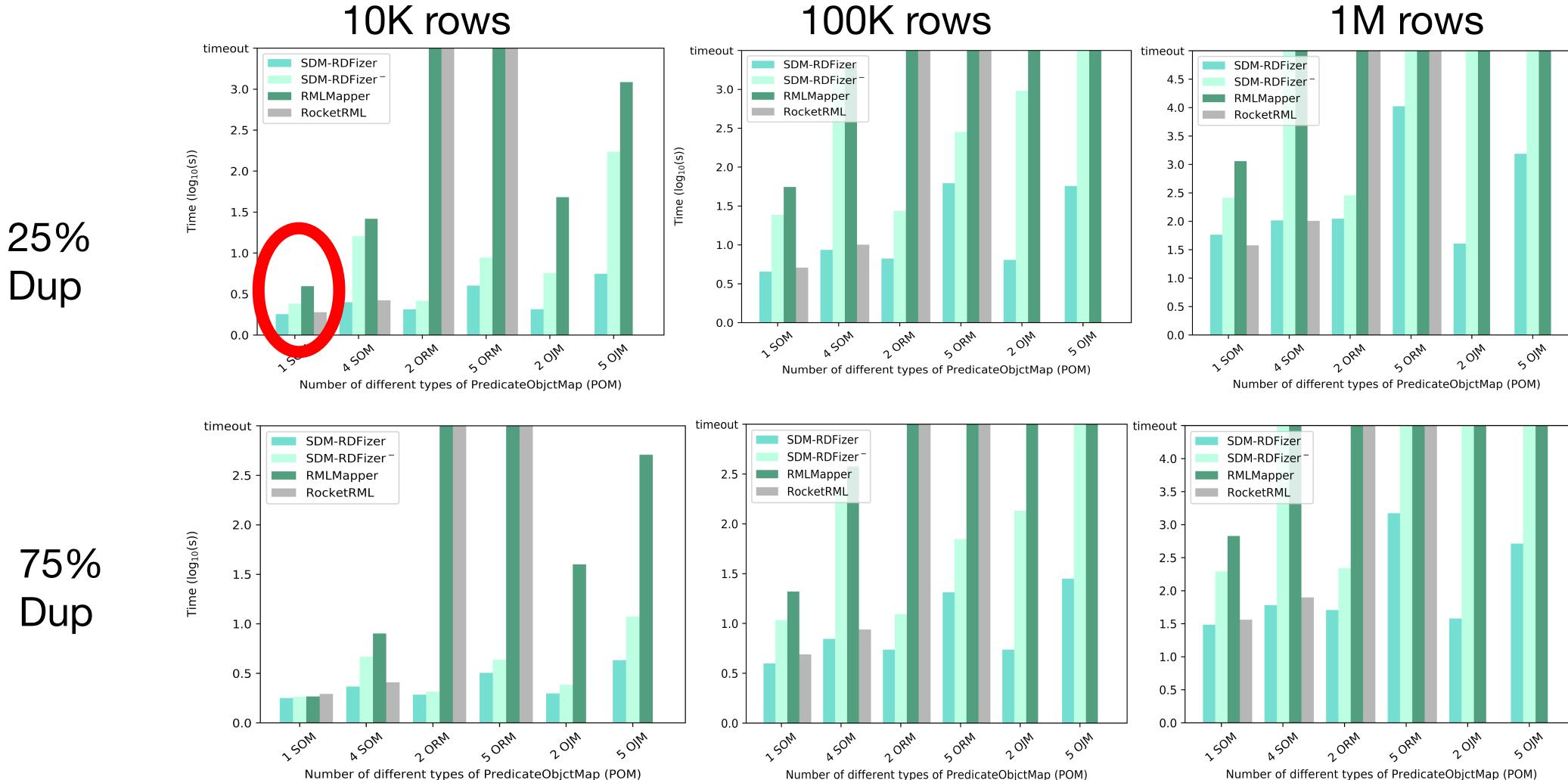


Iglesias, E., Jozashoori, S., **Chaves-Fraga, D.**, Collarana, D., & Vidal, M. E. (2020). SDM-RDFizer: An RML interpreter for the efficient creation of rdf knowledge graphs. In *Proceedings of the 29th ACM CIKM*.

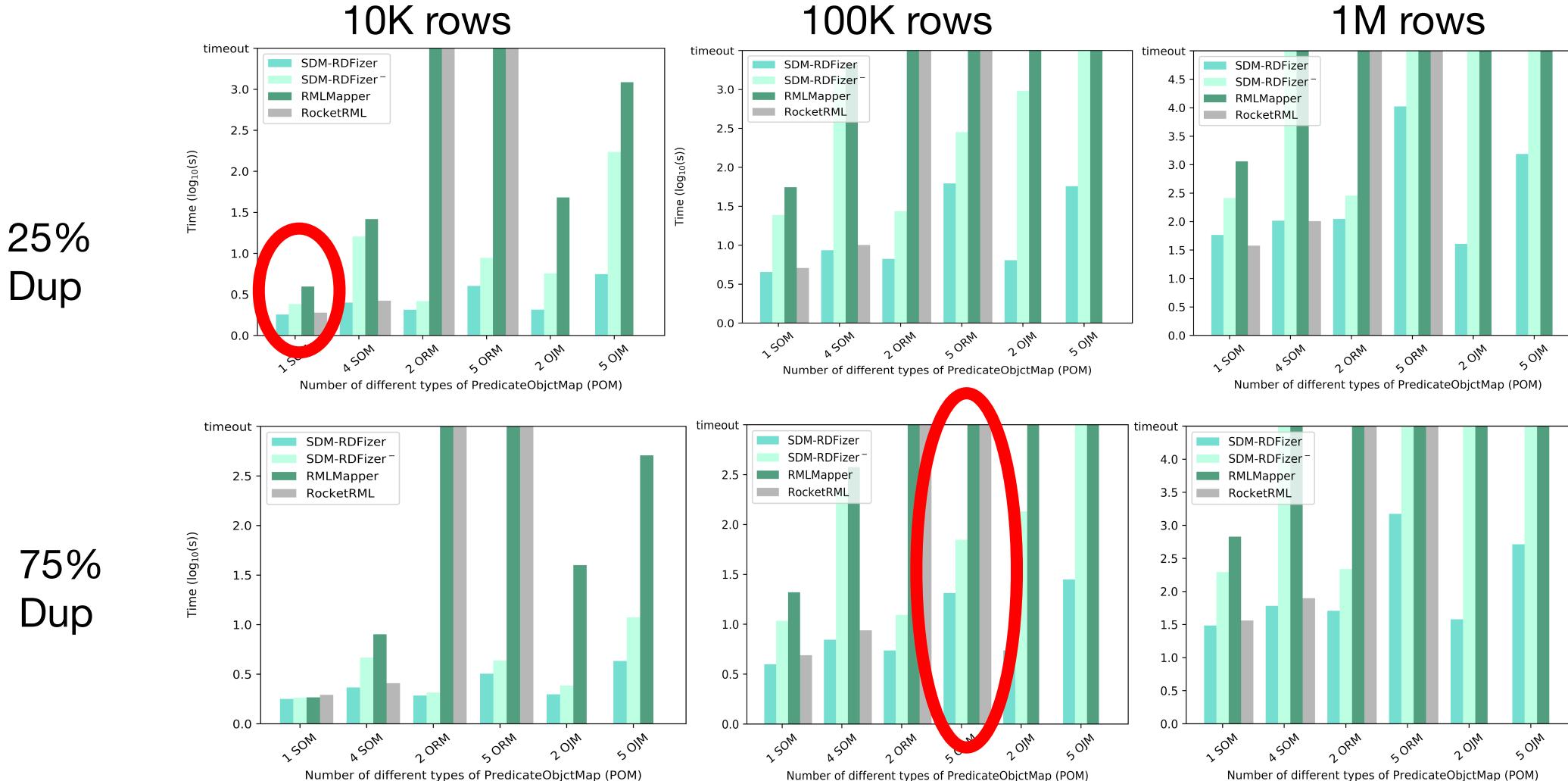
What is the impact of data duplication rate, data size and triples map types in the execution time of a knowledge graph creation approach?



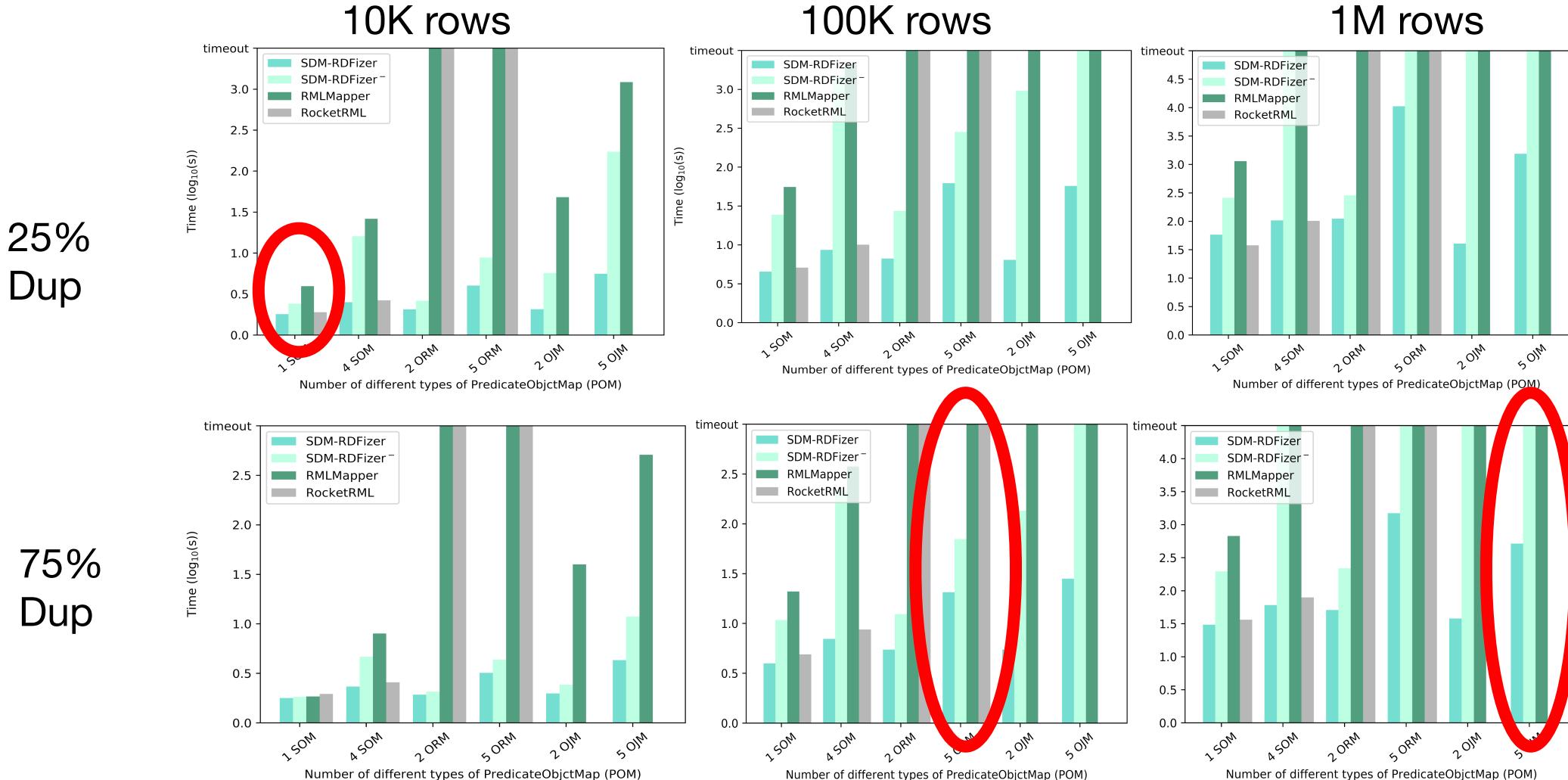
What is the impact of data duplication rate, data size and triples map types in the execution time of a knowledge graph creation approach?



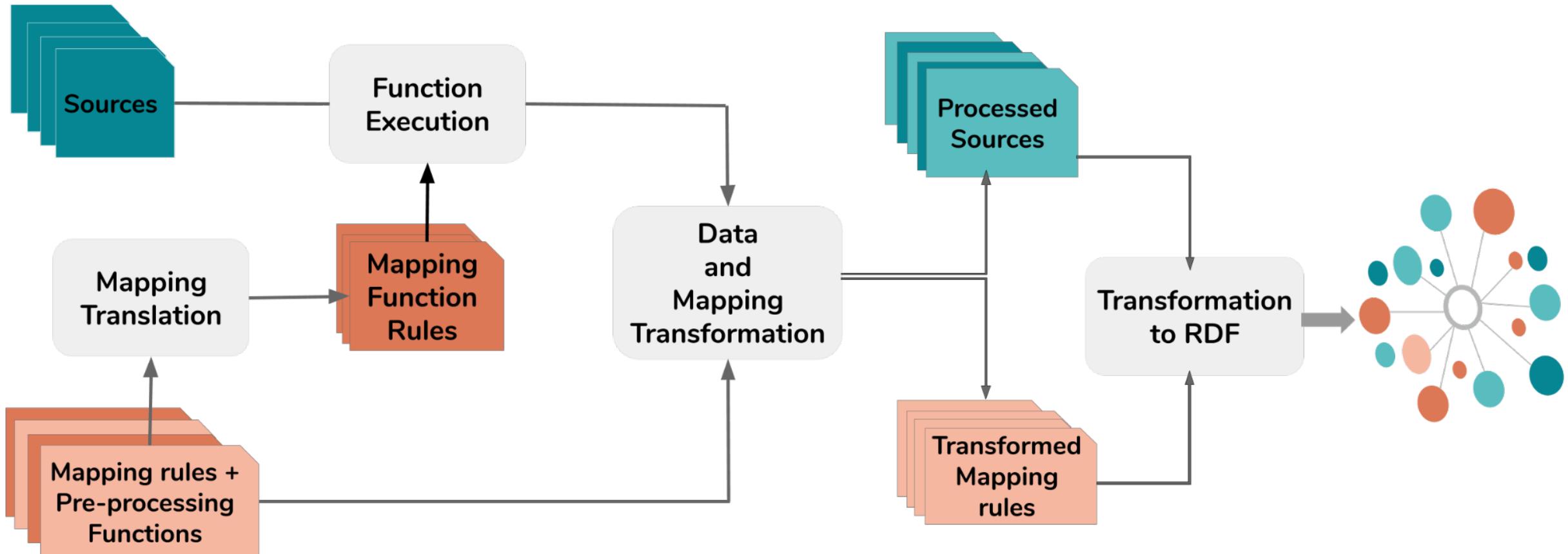
What is the impact of data duplication rate, data size and triples map types in the execution time of a knowledge graph creation approach?



What is the impact of data duplication rate, data size and triples map types in the execution time of a knowledge graph creation approach?



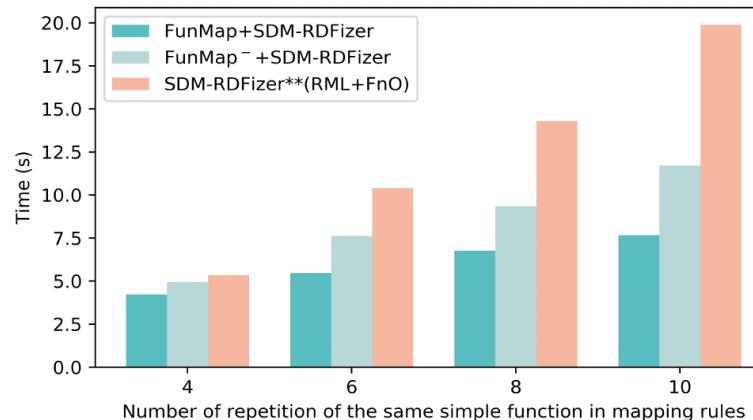
## How can efficiently run mappings that also contain ad-hoc data transformations?



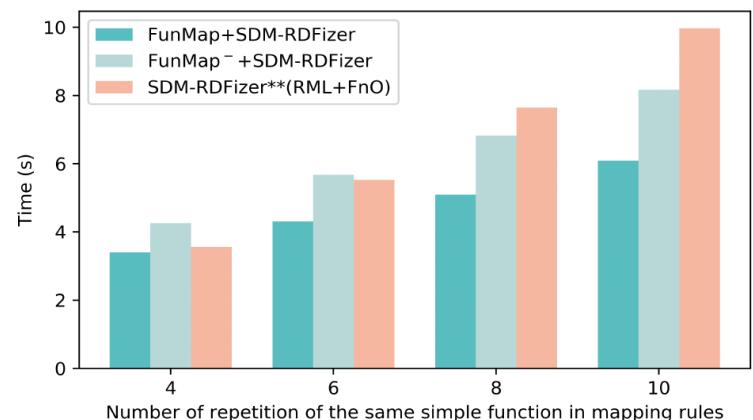
Jozashoori, S., Chaves-Fraga, D., Iglesias, E., Vidal, M. E., & Corcho, O. (2020, November). FunMap: Efficient Execution of Functional Mappings for Knowledge Graph Creation. In *International Semantic Web Conference*. [Fully reproduced paper](#)

What is the impact of data duplication rate and different types of complexity over transformation functions in the execution time of a knowledge graph construction approach?

Simple  
functions  
(lower, upper)

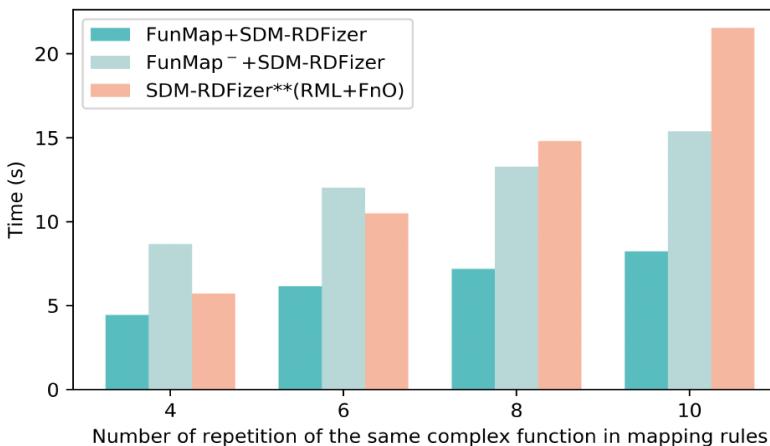


(a) SDM-RDFizer - 25% of duplicates

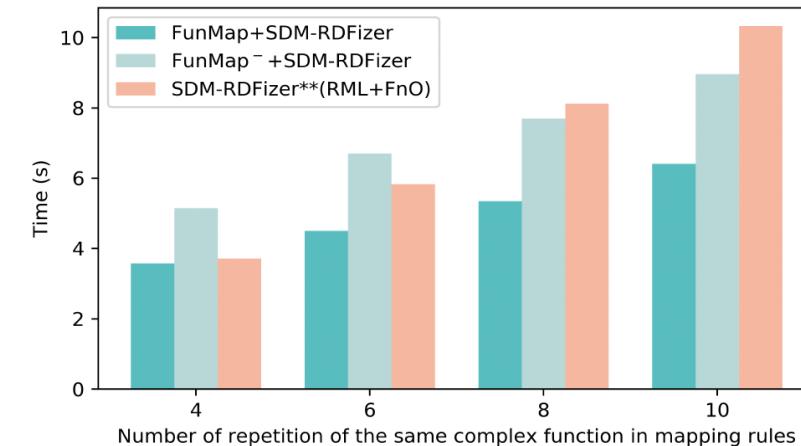


(b) SDM-RDFizer - 75% of duplicates

Complex  
functions  
(if, replace,  
multiple  
columns)



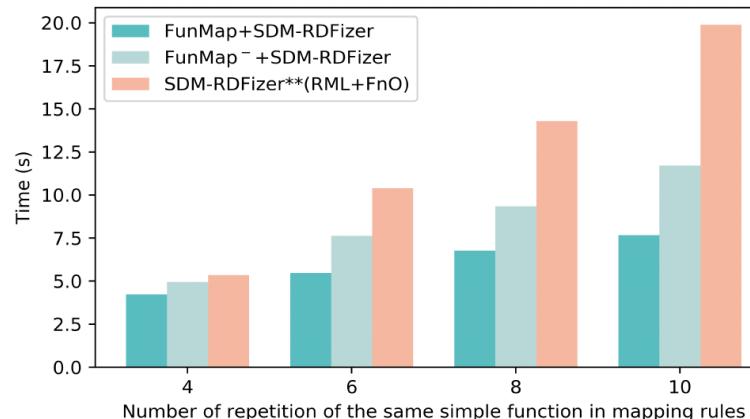
(a) SDM-RDFizer - 25% of duplicates



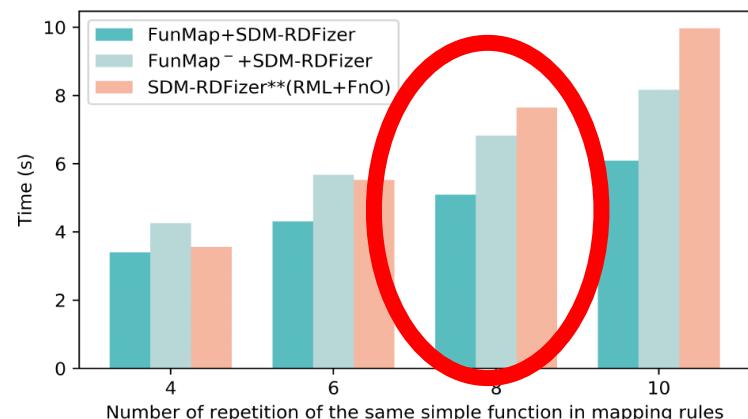
(b) SDM-RDFizer - 75% of duplicates

What is the impact of data duplication rate and different types of complexity over transformation functions in the execution time of a knowledge graph construction approach?

Simple  
functions  
(lower, upper)

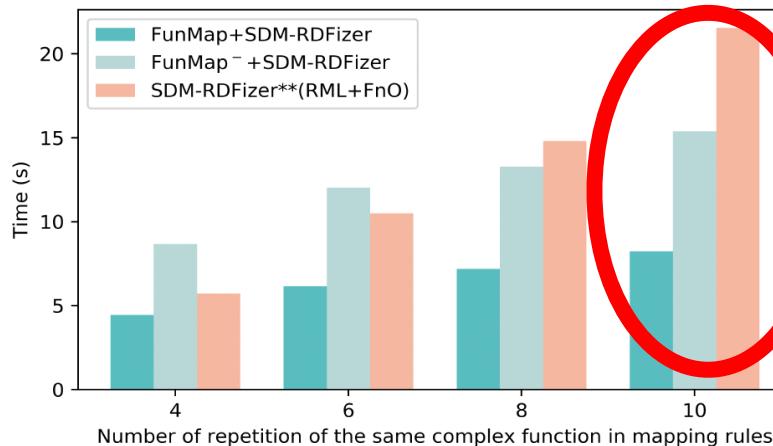


(a) SDM-RDFizer - 25% of duplicates

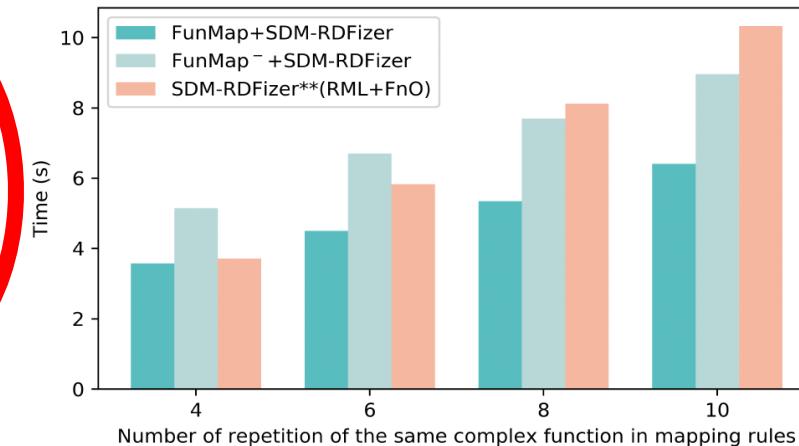


(b) SDM-RDFizer - 75% of duplicates

Complex  
functions  
(if, replace,  
multiple  
columns)



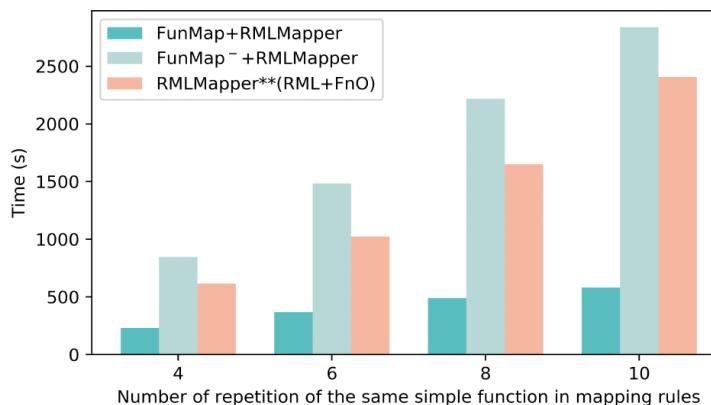
(a) SDM-RDFizer - 25% of duplicates



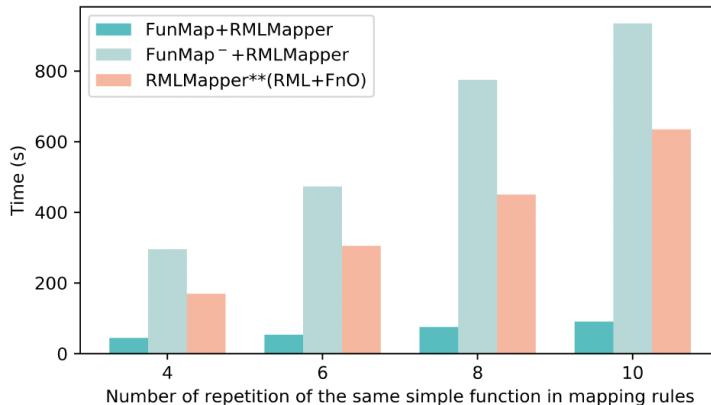
(b) SDM-RDFizer - 75% of duplicates

What is the impact of data duplication rate and different types of complexity over transformation functions in the execution time of a knowledge graph construction approach?

Simple  
functions  
(lower, upper)

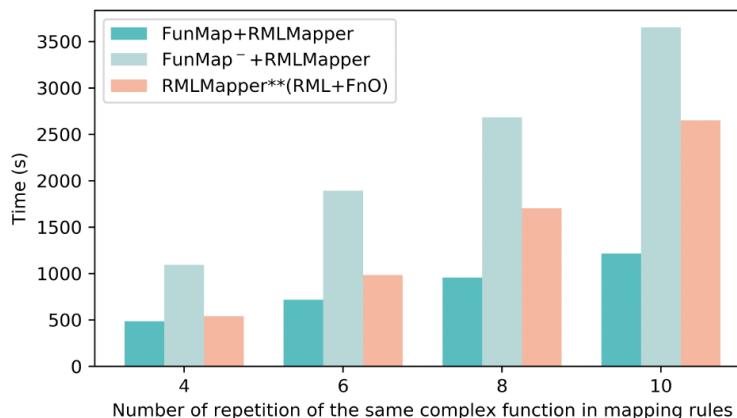


(c) RMLMapper - 25% of duplicates

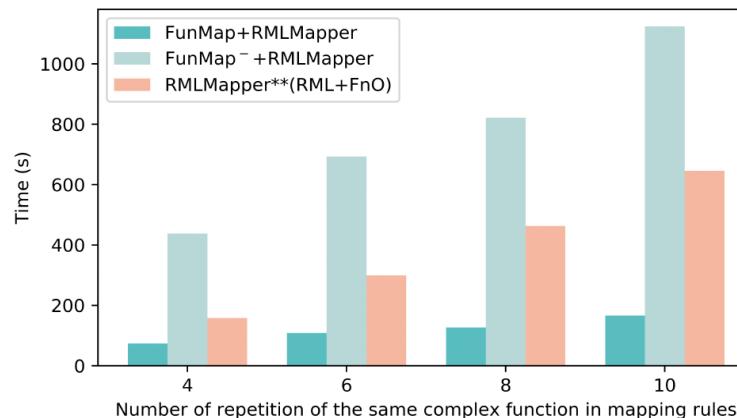


(d) RMLMapper - 75% of duplicates

Complex  
functions  
(if, replace,  
multiple  
columns)

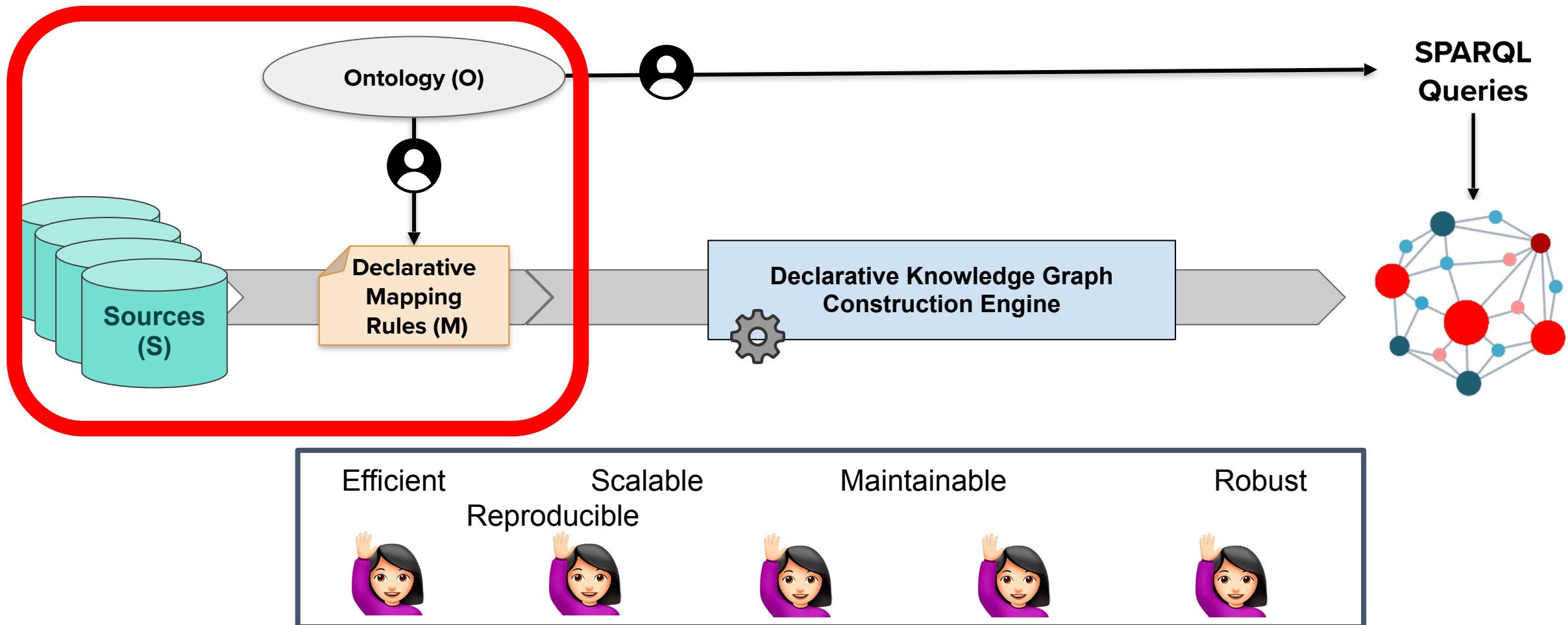


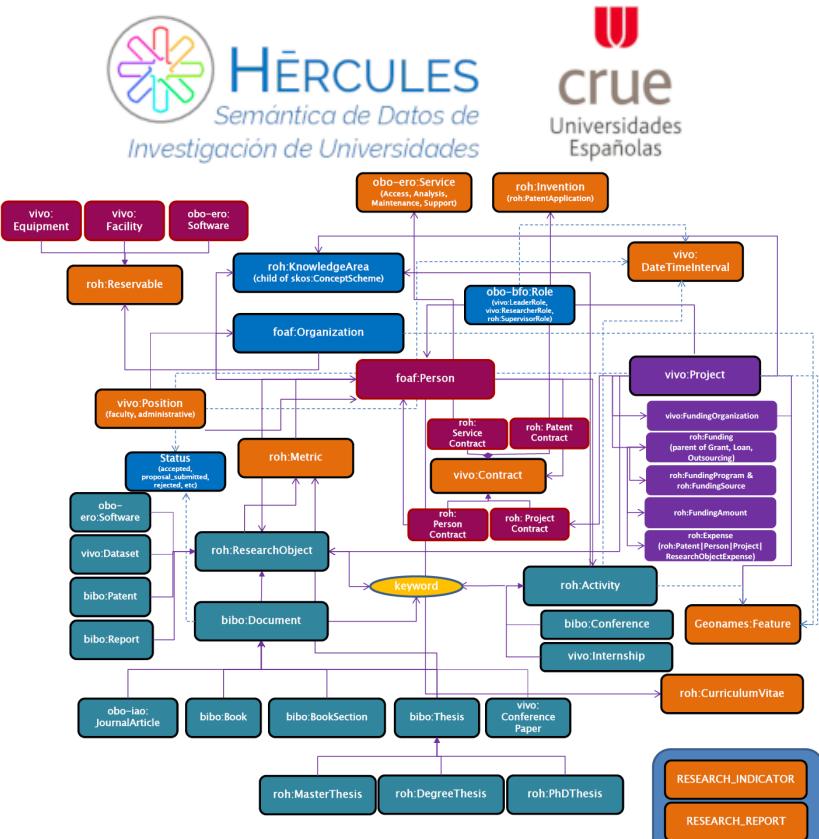
(c) RMLMapper - 25% of duplicates



(d) RMLMapper - 75% of duplicates

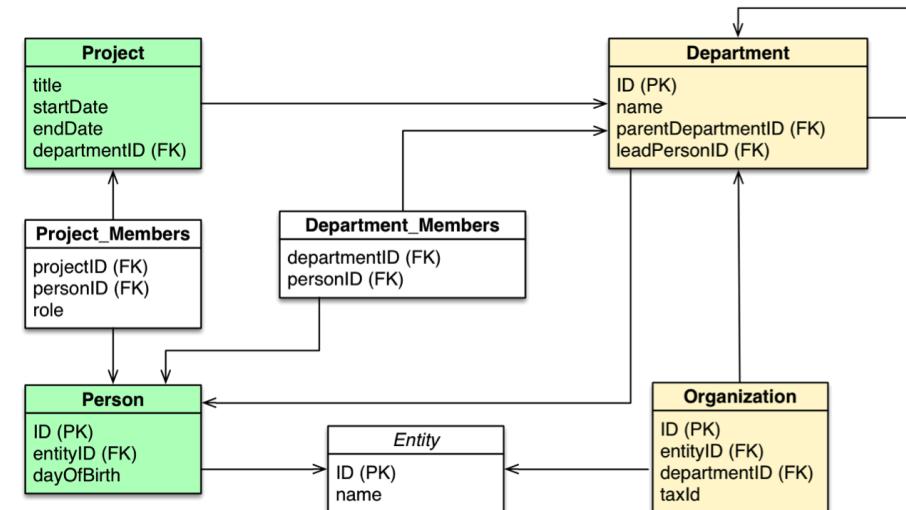
**Knowledge Graph Construction = Data Integration System (DIS) =  $\langle S, M, O \rangle$**





- Ontology v0.3 (will change)
- Not standard documentation (PDF file)

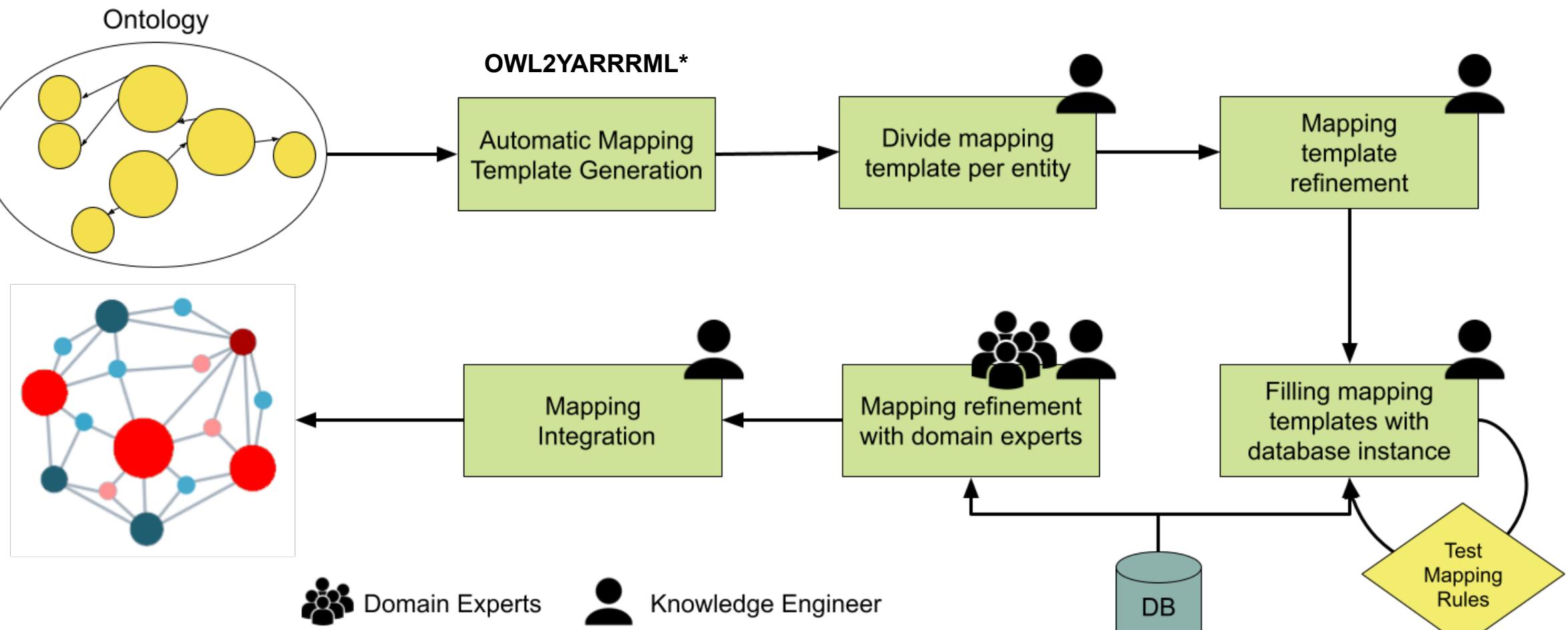
# ORACLE



- More than 1800 tables
- Database very well documented
- Oracle supports R2RML mappings



Chaves-Fraga, D., Corcho, O., Yedro, F., Moreno, R., Olías, J., & De La Azuela, A. (2022). Systematic Construction of Knowledge Graphs for Research-Performing Organizations. *Information*, 13(12), 562.



\* <https://github.com/oeg-upm/owl2yarrml>



### Outcomes:

- Total time: 7 months for mapping creation
- More than 5K rules in R2RML (N-Triples syntax)
- Virtual KG over each entity
- Materialized KG to feed a central repository

### Lessons Learned:

- Simple but useful support tools (OWL2YARRML)
- Domain experts with technical knowledge in the loop
- Divide and Conquer in complex scenarios
- Delegate complex tasks to the DBMS



# Constructing the EU decentralized Data Space for Public Procurement

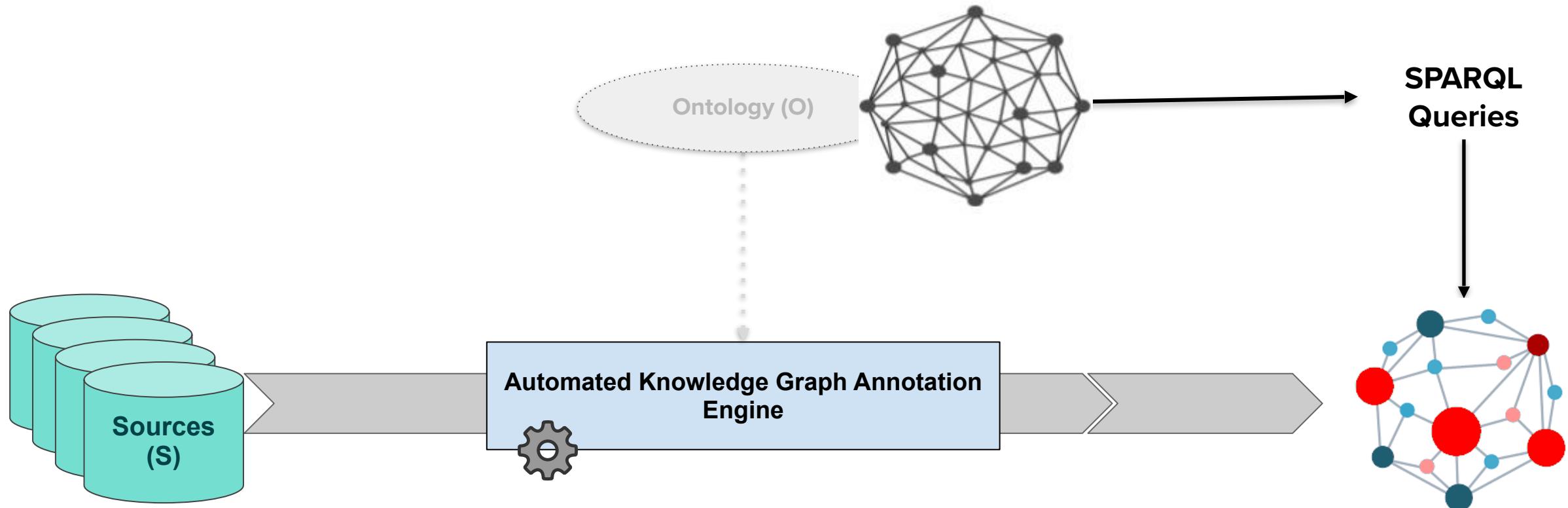
- Homogenize the access to all public procurement data across Europe
- First semantic data architecture of this magnitude in the world
- Calculate standard transparency indicators for each member state
- Researching on: vocabularies, resources maintainability, federated query processing, semantic data ingestion, query performance/scalability...

- Partners:





# **Challenges for ensuring data integration scalability during the decentralization of the Web**



Chaves-Fraga, D. & Dimou, A. (2022). Declarative Description of Knowledge Graphs Construction Automation: Status & Challenges. In *Third International Workshop on Knowledge Graph Construction co-located with ESWC2022*.

# KGCW2023 Challenge

Knowledge graph construction of heterogeneous data has seen a lot of uptake in the last decade from compliance to performance optimizations with respect to execution time. Besides execution time as a metric for comparing knowledge graph construction, other metrics e.g. CPU or memory usage are not considered. This challenge aims at benchmarking systems to find which RDF graph construction system optimizes for metrics e.g. execution time, CPU, memory usage, or a combination of these metrics.

## Task Description

The task is to reduce and report the execution time and computing resources (CPU and memory usage) for the parameters listed in this challenge, compared to the state-of-the-art of the existing tools and the baseline results provided by this challenge. This challenge is not limited to execution times to create the fastest pipeline, but also computing resources to achieve the most efficient pipeline.

- Mapping rules (in any form) are the central resource of KG generation
- Background of domain experts / users have to be considered
- Adaptability means successful (one fits doesn't fit all)
- Trade-offs: Automation VS Data quality
- Query Federation
- Governance of Data Integration Systems (Sources, Mappings, Ontology)

# Are Knowledge Graphs Ready for the Real World? Challenges and Perspective

Accepted Long-Large Dagstuhl Seminar (February 2024)



Anastasia  
Dimou



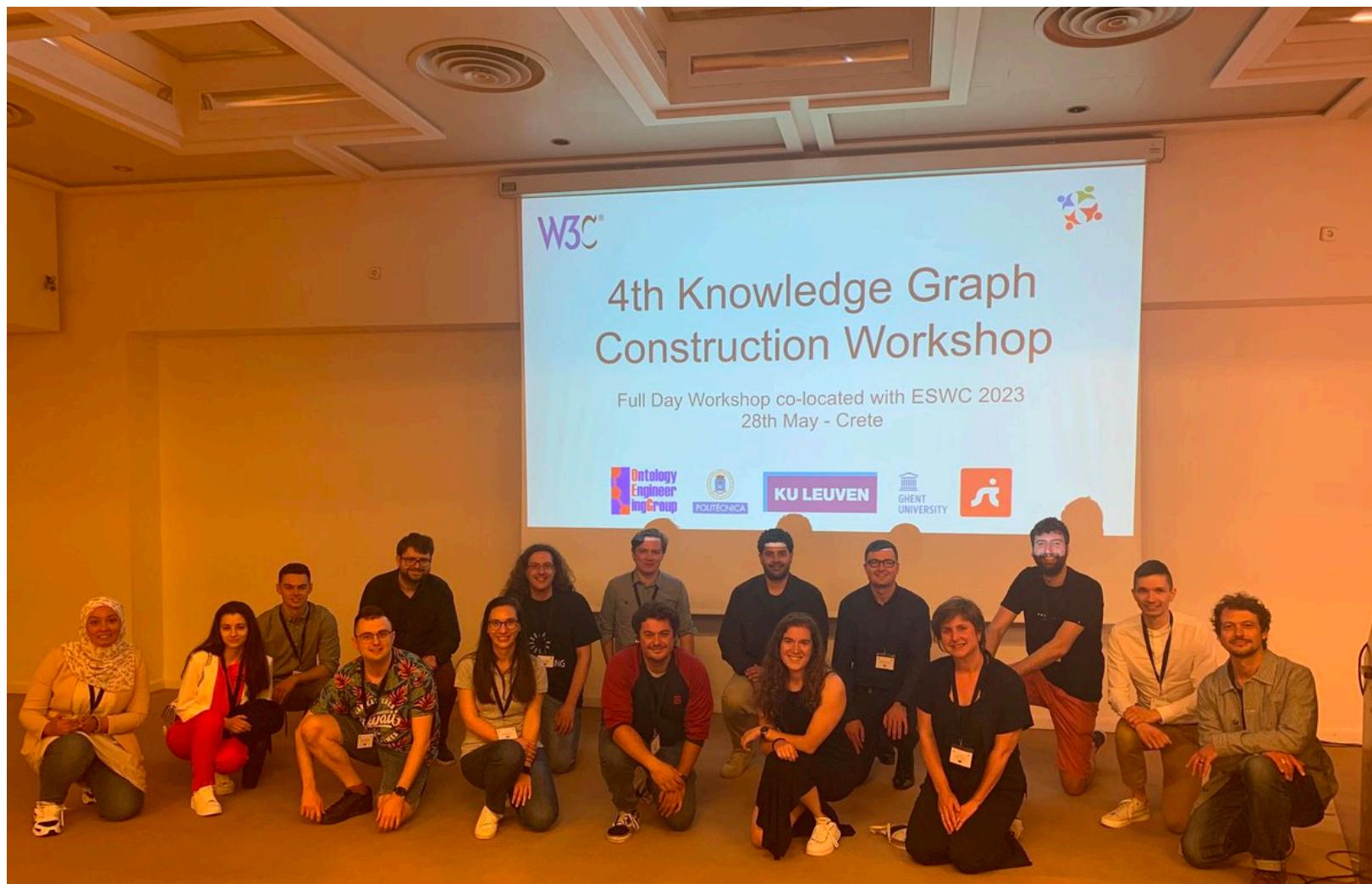
David  
Chaves-Fraga

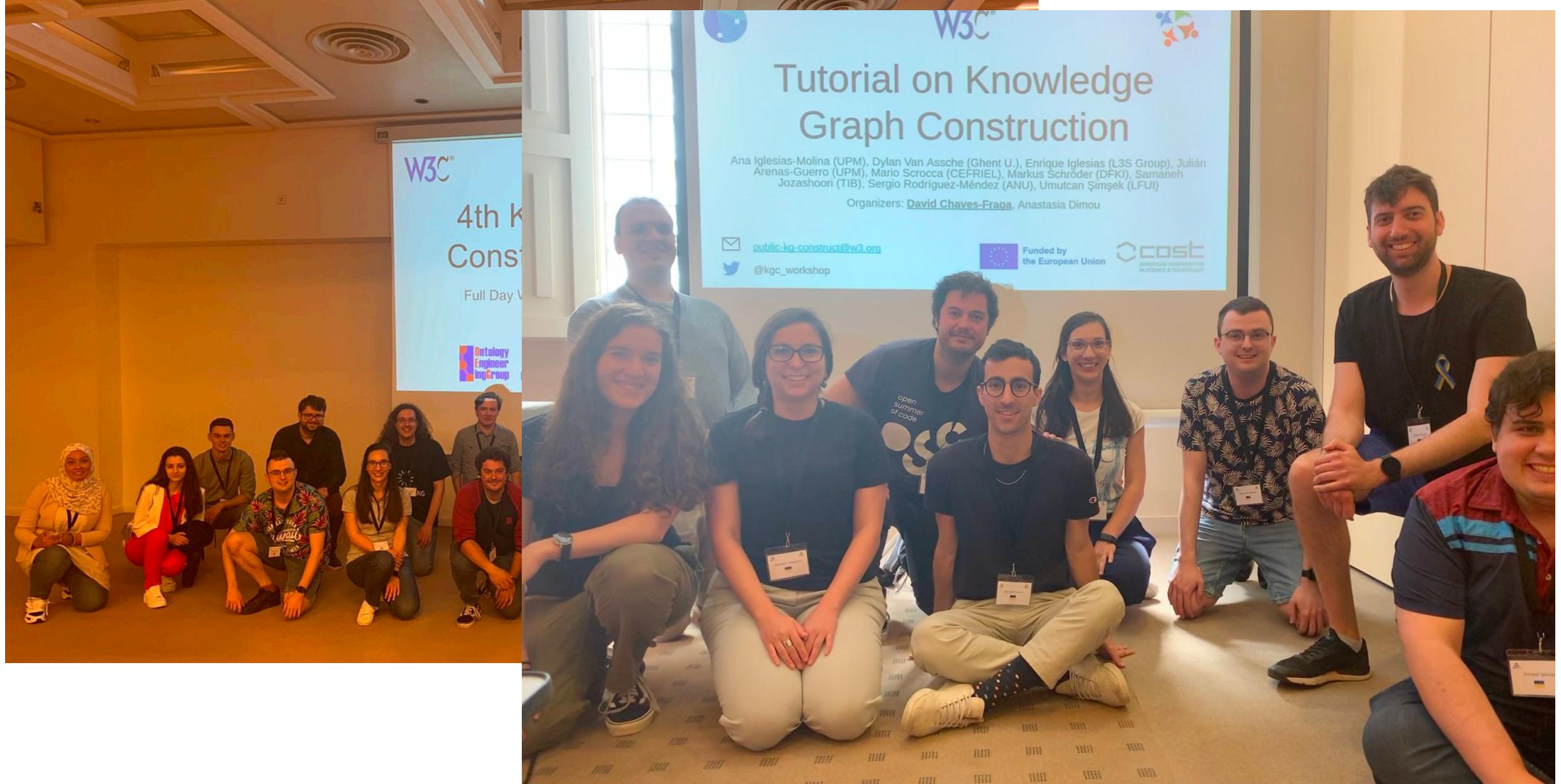


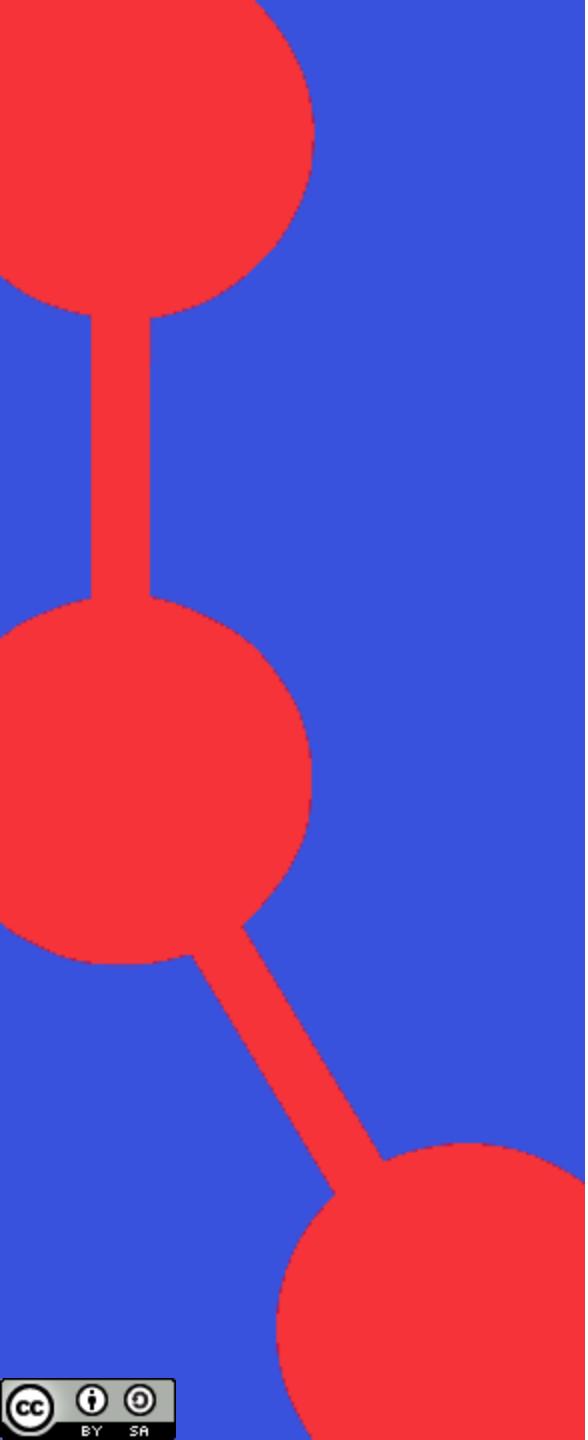
Maria-Ester  
Vidal



Oscar  
Corcho







# Web-Scale Semantic Data Integration: Challenges and Perspective

David Chaves-Fraga

Ontology Engineering Group  
Universidad Politécnica de Madrid, Spain

&

Declarative Languages and Artificial Intelligence,  
KULeuven, Belgium