Declarative Knowledge Graph Construction:
A Practical Introduction

David Chaves-Fraga

✉ david.chaves@kuleuven.be

🐦 @dchavesf

A bit about myself…

PhD in Artificial Intelligence (2021)

"Knowledge Graph Construction from Heterogeneous
Data Sources Exploiting Declarative Mapping Rules"

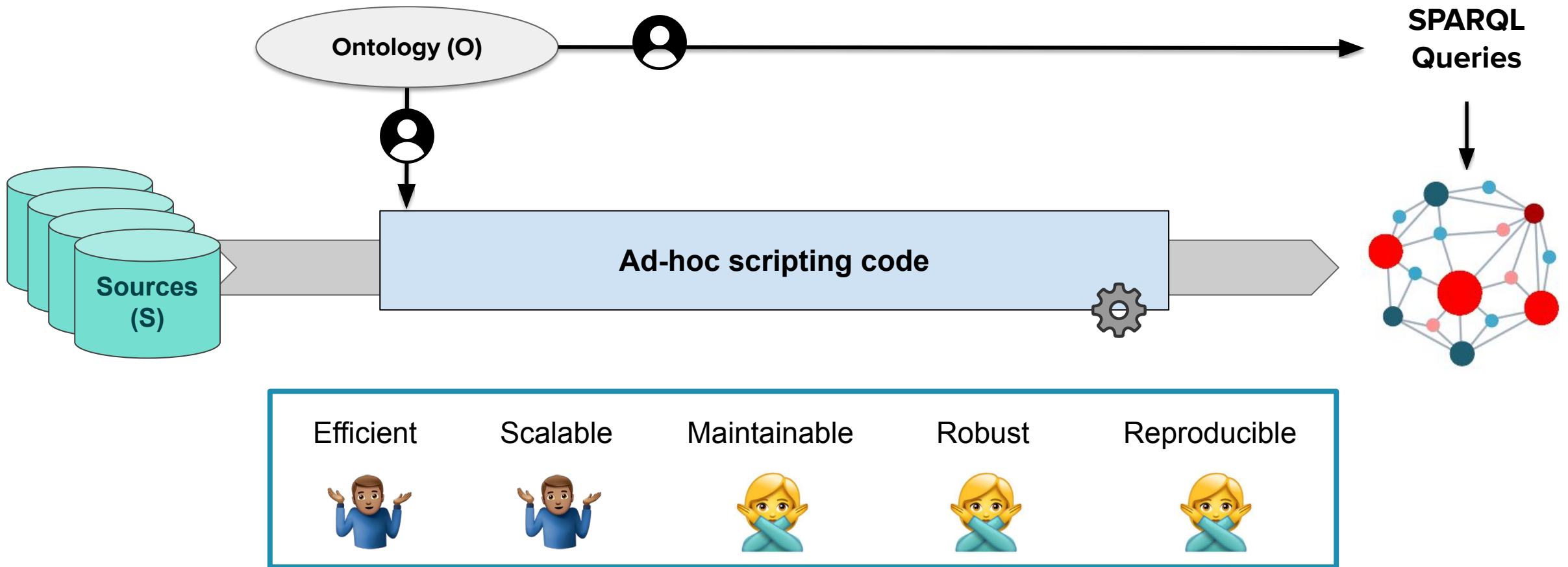Co-chair W3C CG on Knowledge Graph Construction (2019-now)

Joint postdoctoral researcher (2022- now)

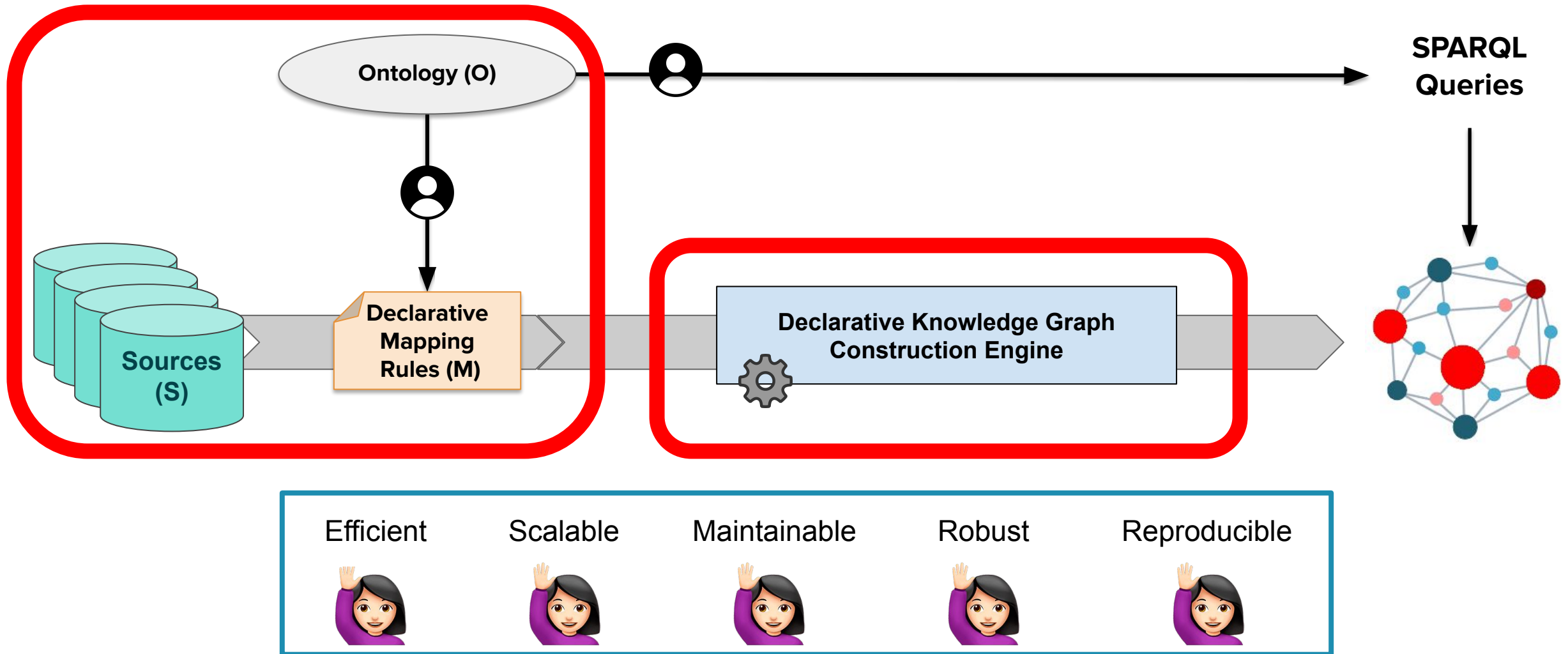Knowledge Graph Construction Workshop Organizer (2019-now)

KU LEUVEN

Knowledge Graph Construction: Scripting-based

Ontology (O)

SPARQL Queries

Sources (S)

Ad-hoc scripting code

| Efficient | Scalable | Maintainable | Robust | Reproducible |

KG Construction with Mapping Rules

Ontology (O)

SPARQL Queries

Sources (S)

Declarative Mapping Rules (M)

Declarative Knowledge Graph Construction Engine

| Efficient | Scalable | Maintainable | Robust | Reproducible |

KU LEUVEN

Mapping Rules Creation: (YARR)RML

SPARQL Queries

Ontology (O)

Sources (S)

Declarative Mapping Rules (M)

Declarative Knowledge Graph Construction Engine

Efficient   Scalable   Maintainable   Robust   Reproducible

KU LEUVEN
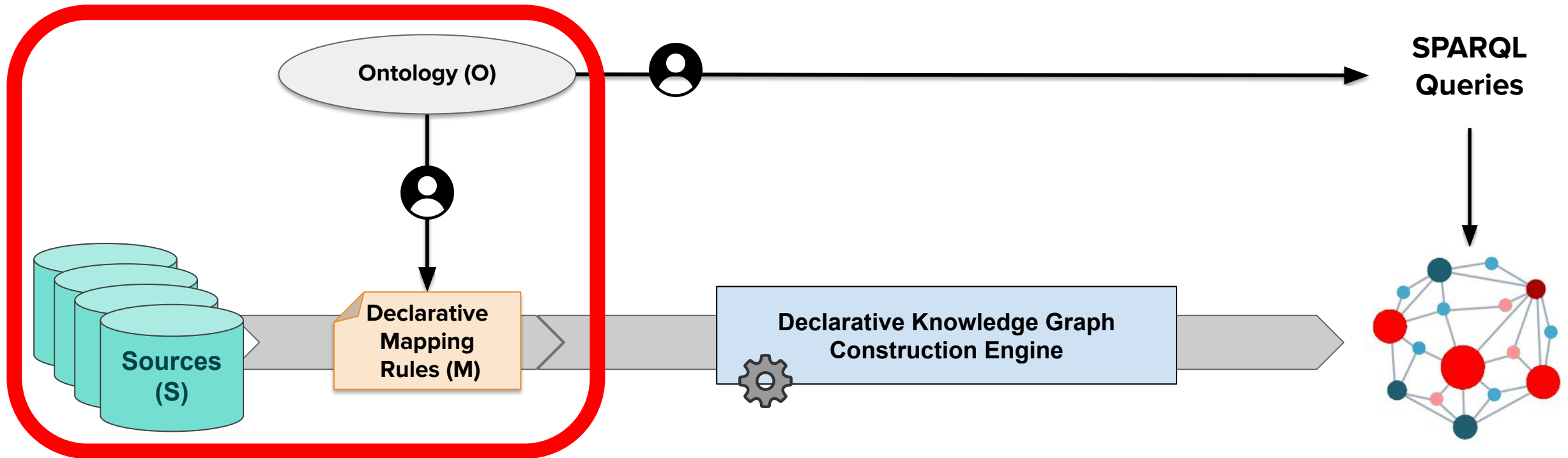
Declarative mapping languages - *schema* transformations

query languages
SPARQL syntax

(Facade-X)
SPARQL-Anything
(2021)

XSPARQL
(2008)

SPARQL-Generate
(2017)

RML
(2013)

xR2RML
(2015)

D2RML
(2018)

YARRRML
(2018)

D-REPR
(2019)

ShExML
(2020)

dedicated languages
R2RML syntax

data-serialisation languages
YAML syntax

constraint languages
ShEx syntax

* Adapted from Knowledge Graph Construction Tutorial @ ESWC 2022 (Anastasia Dimou)

KU LEUVEN

rank	name	nationality	mark	notes
1	Anzhelika Sidorova	Russia	4.95	WL,PB
2	Sandi Morris	USA	4.90	SB
3	Katerina Stefanidi	Greece	4.85	SB
4	Holly Bradshaw	UK	4.80	-
5	Alysha Newman	Canada	4.80	-
6	Angelica Bengtsson	Sweden	4.80	NR

* Adapted from Knowledge Graph Construction Tutorial @ ESWC 2022 (Anastasia Dimou)

KU LEUVEN
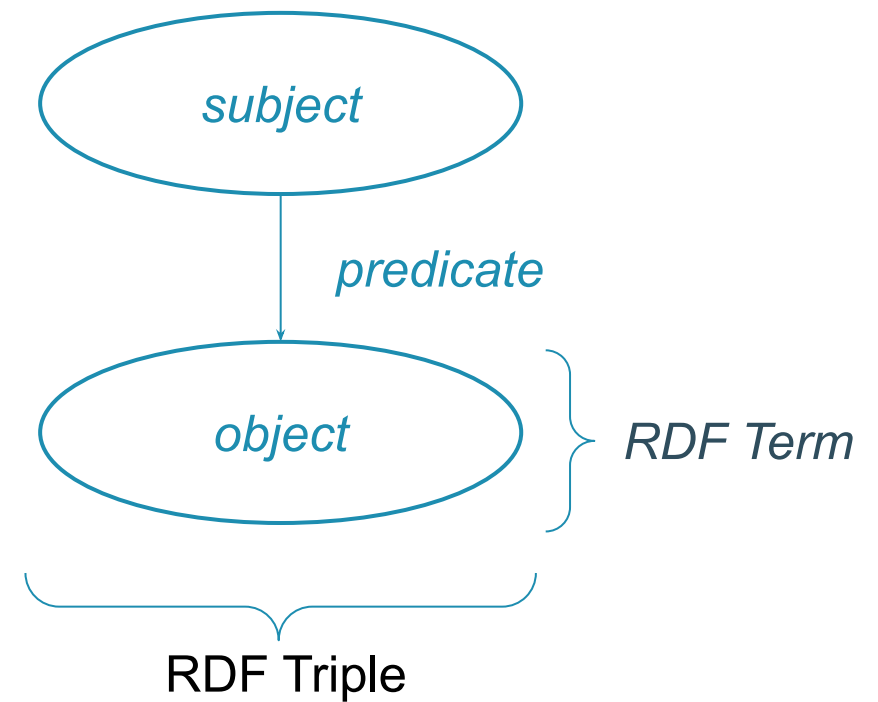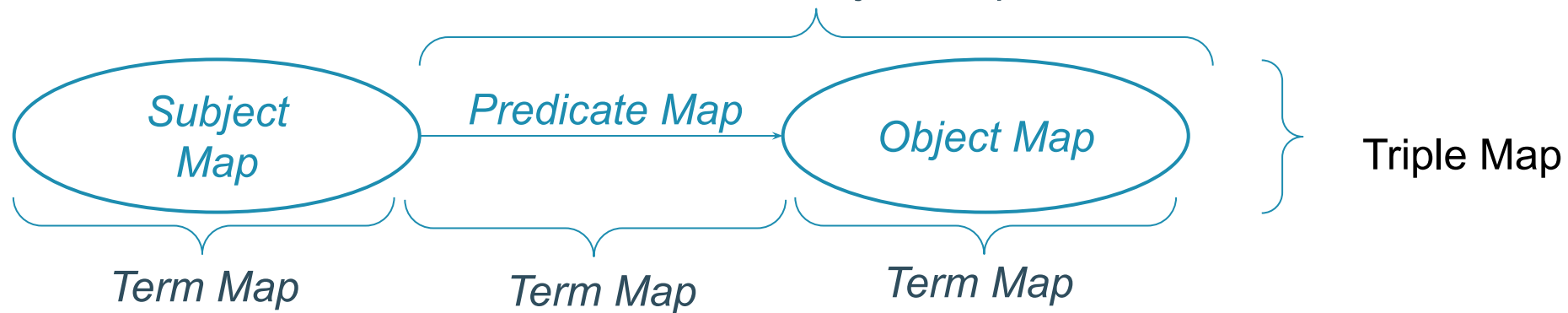
rank	name	surname	nationality	mark	notes
1	Anzhelika	Sidorova	Russia	4.95	WL,PB
2	Sandi	Morris	USA	4.90	SB
3	Katerina	Stefanidi	Greece	4.85	SB
4	Holly	Bradshaw	UK	4.80	-
5	Alysha	Newman	Canada	4.80	-
6	Angelica	Bengtsson	Sweden	4.80	NR

```
<#TriplesMap_1> [
    rr:subjectMap [
        rr:template "http://ex.com/{name}";
        rr:class foaf:Person; ];
    rr:predicateObjectMap [
        rr:predicateMap [rr:constant foaf:name];
        rr:objectMap [ rr:template "{name} {surname}";
                        rr:termType rr:Literal;
                        rr:language "en"] ] ].
```
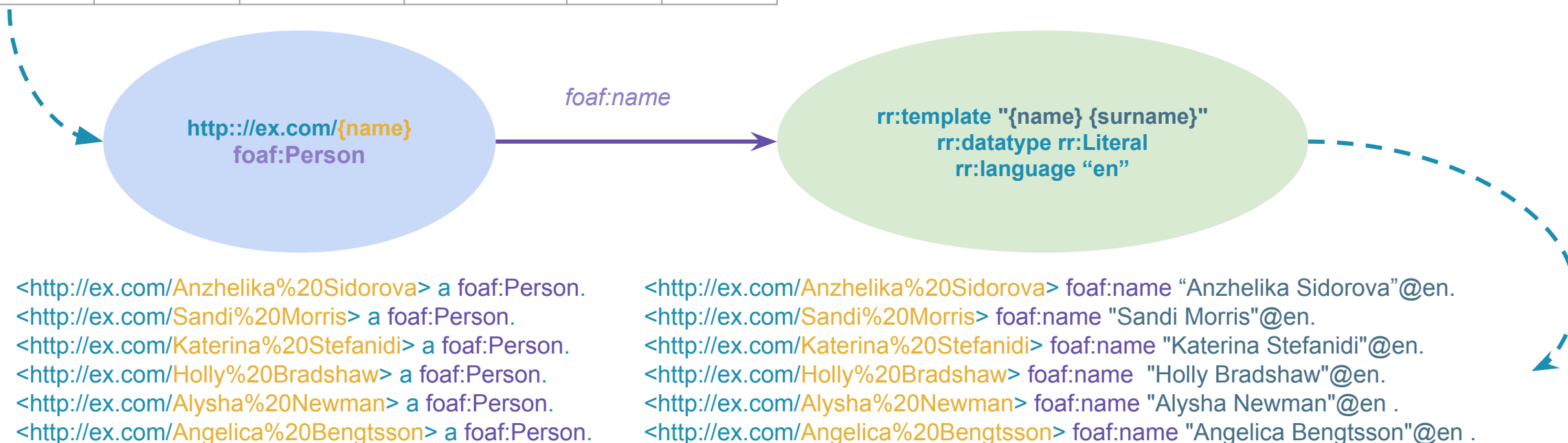

http://ex.com/{name}
foaf:Person

foaf:name

rr:template "{name} {surname}"
rr:datatype rr:Literal
rr:language "en"

<http://ex.com/Anzhelika%20Sidorova> a foaf:Person.
<http://ex.com/Sandi%20Morris> a foaf:Person.
<http://ex.com/Katerina%20Stefanidi> a foaf:Person.
<http://ex.com/Holly%20Bradshaw> a foaf:Person.
<http://ex.com/Alysha%20Newman> a foaf:Person.
<http://ex.com/Angelica%20Bengtsson> a foaf:Person.

<http://ex.com/Anzhelika%20Sidorova> foaf:name "Anzhelika Sidorova"@en.
<http://ex.com/Sandi%20Morris> foaf:name "Sandi Morris"@en.
<http://ex.com/Katerina%20Stefanidi> foaf:name "Katerina Stefanidi"@en.
<http://ex.com/Holly%20Bradshaw> foaf:name "Holly Bradshaw"@en.
<http://ex.com/Alysha%20Newman> foaf:name "Alysha Newman"@en .
<http://ex.com/Angelica%20Bengtsson> foaf:name "Angelica Bengtsson"@en .

* Adapted from Knowledge Graph Construction Tutorial @ ESWC 2022 (Anastasia Dimou)

KU LEUVEN

YARRRML

https://rml.io/yarrrml/

```
<#TriplesMap_1> [
  rml:logicalSource [
    rml:source "poleVaulters.csv";
    rml:referenceFormulation ql:CSV; ]; ]
  rr:subjectMap [
    rr:template "http://ex.com/{name}"; ];
  rr:predicateObjectMap [
    rr:predicateMap [rr:constant ex:score];
    rr:objectMap [ rml:reference "mark";
                   rr:datatype xsd:decimal]; ];
  rr:predicateObjectMap [
    rr:predicateMap [rr:constant foaf:name];
    rr:objectMap [ rml:reference "name"; rr:language "en"]; ];
  rr:predicateObjectMap [
    rr:predicateMap [rr:constant ex:country];
    rr:objectMap [ rr:parentTriplesMap <#TriplesMap_2>;
                   rr:joinCondition [
                          rr:parent "country_name";
                          rr:child "nationality"] ]; ] ].

<#TriplesMap_2> [
rml:logicalSource [
  rml:source "countries.xml";
  rml:referenceFormulation ql:XPath;
  rml:iterator "countries/country" ];
  rr:subjectMap [
    rr:template "http://ex.com/{country_abb}";
    rr:graphMap [ rr:constant ex:CountryGraph ]; ].
```

RML

```
mapping:
  person:
    sources:
      - [poleVaulters.csv~csv]
    subject:
      - "http://ex.com/{name}"
    predicateobjects:
      - [ex:score, $(mark), xsd:decimal]
      - [foaf:name, $(name), en~lang]
      - [foaf:name, $(name) $(surname), en~lang]
      - predicates: ex:country
        objects:
          - mapping: country
            condition:
              function: equal
              parameters:
                - [str1, $(nationality), s]
                - [str2, $(country_name), o]
  country:
    sources:
      - [countries.xml~xPath, countries/country]
    subjects: http://ex.com/{country_abb}
    graph: ex:CountryGraph
```

YARRRML

* Adapted from Knowledge Graph Construction Tutorial @ ESWC 2022 (Dylan Van Assche)

KU LEUVEN

Mapping Helpers

User interfaces:
- Matey: https://github.com/rmlio/matey (for YARRRML)
- Mapeathor (based on Excel): https://morph.oeg.fi.upm.es/tool/mapeathor (for [R2]RML)
- RMLEditor: https://app.rml.io/rmleditor/ (for RML)

Mapping creators:

- OWL2YARRRML: https://github.com/oeg-upm/owl2yarrrml (from OWL Ontology to YARRRML template)
- Spread2RML: https://www.dfki.uni-kl.de/~mschroeder/demo/spread2rml/ (from SpreadSheet to RML)

Mapping translators:

- YARRRML-parser: https://github.com/rmlio/yarrrml-parser (from YARRRML to [R2]RML)
- Pretty-YARRRML2RML: https://github.com/oeg-upm/pretty-yarrrml2rml (from YARRRML to Pretty-RML)

Engine for KG Construction: Morph-KGC

KU LEUVEN

materialisation
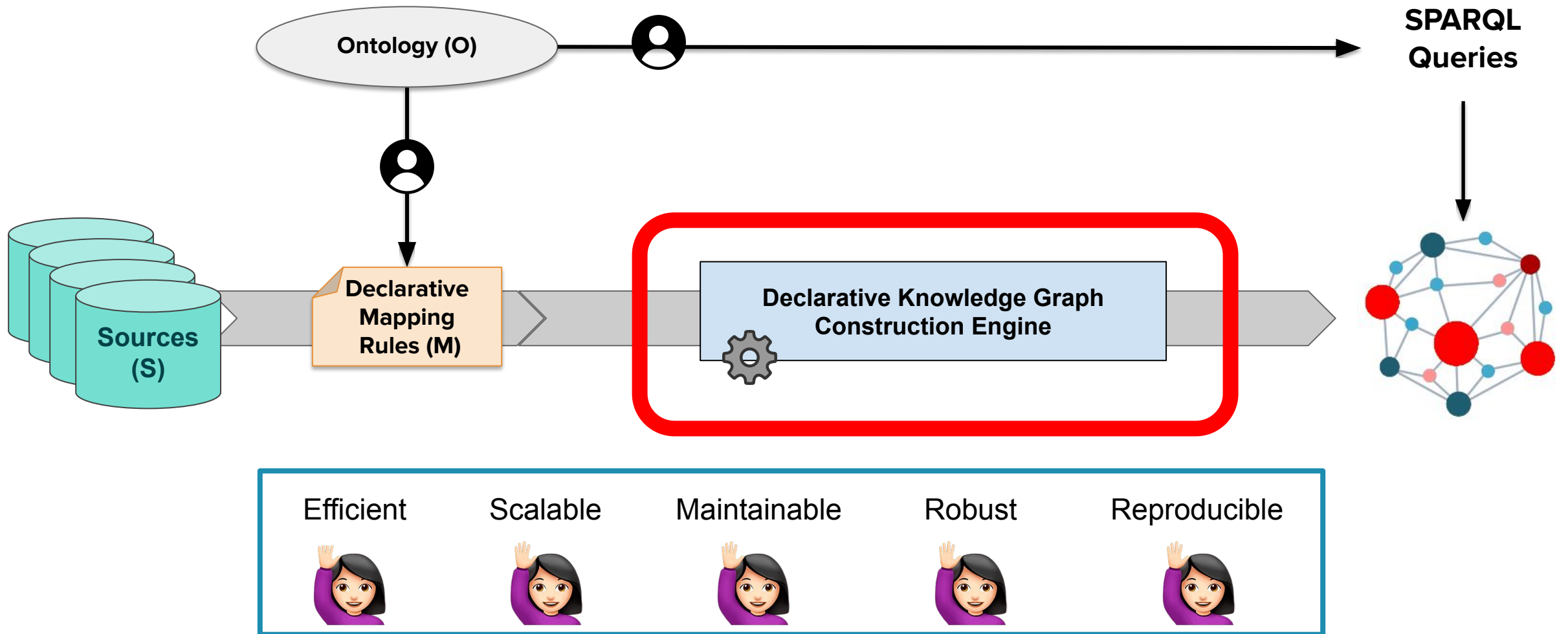
DB2triples (https://github.com/antidot/db2triples)
R2RML Parser (https://github.com/nkons/r2rml-parser)
XSPARQL (http://xsparql.sourceforge.net/)
Morph-KGC (https://github.com/oeg-upm/morph-kgc)
R2RML-F (https://github.com/chrdebru/r2rml)

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

**homogeneous
data sources**

**heterogeneous
data sources**

R2RML
(RDBs)

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

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)

Morph-xR2RML (https://github.com/frmichel/morph-xr2ml)
Squerall (https://github.com/EIS-Bonn/Squerall)
Ontario (https://github.com/SDM-TIB/Ontario/)

virtualisation

* Adapted from Knowledge Graph Construction Tutorial @ ESWC 2022 (Anastasia Dimou)

KU LEUVEN

Morph-KGC: Scalable KG construction
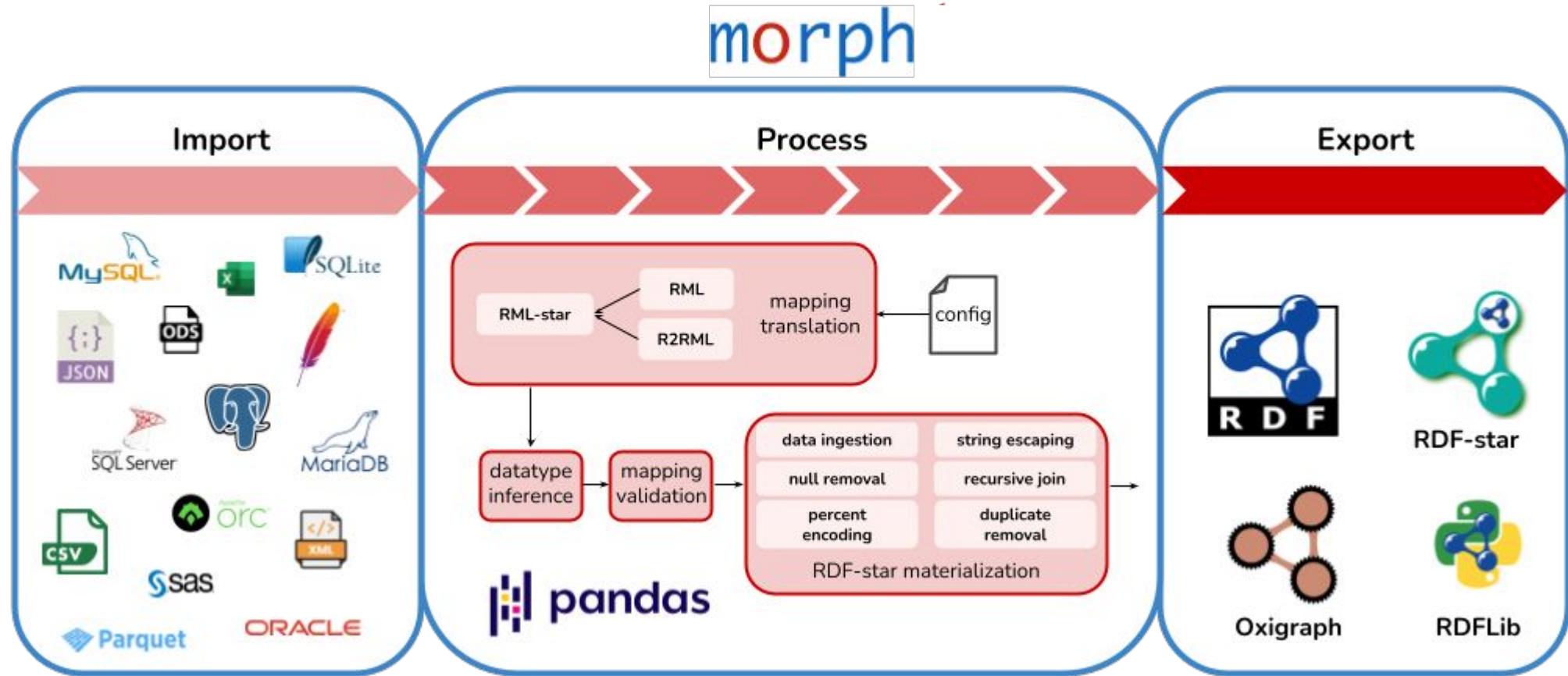
Arenas-Guerrero, J., Chaves-Fraga, D., Toledo, J., S. Perez, M., & Corcho, O. (2022). Morph-KGC: Scalable knowledge graph materialization with mapping partitions. *Semantic Web Journal (http://dx.doi.org/10.3233/SW-223135)*

Arenas-Guerrero, J., Iglesias-Molina, A., Chaves-Fraga, D., Garijo, D., Corcho, O. & Dimou, A. (2022). Morph-KGC[star]: Declarative generation of RDF-star graphs from heterogeneous data. *Under Review at Semantic Web Journal*

KU LEUVEN

Hands-on time!

morph

https://github.com/oeg-upm/morph-kgc/

https://pypi.org/project/morph-kgc/

https://morph-kgc.readthedocs.io/

https://short.upm.es/umdvm

All info about Morph suite: **https://morph.oeg.fi.upm.es/**

KU LEUVEN