Knowledge Graph Construction and Evolution using Declarative Mapping Languages

PhD Defense

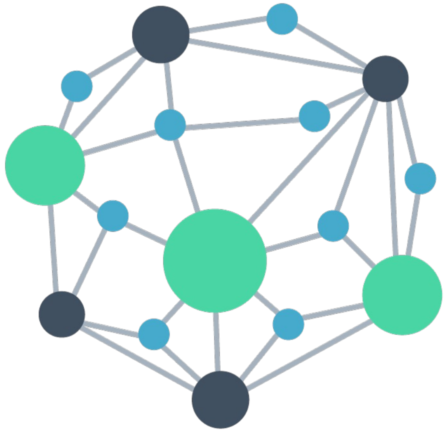
Ana Iglesias Molina
Ontology Engineering Group, Universidad Politécnica de Madrid

Supervised by Oscar Corcho

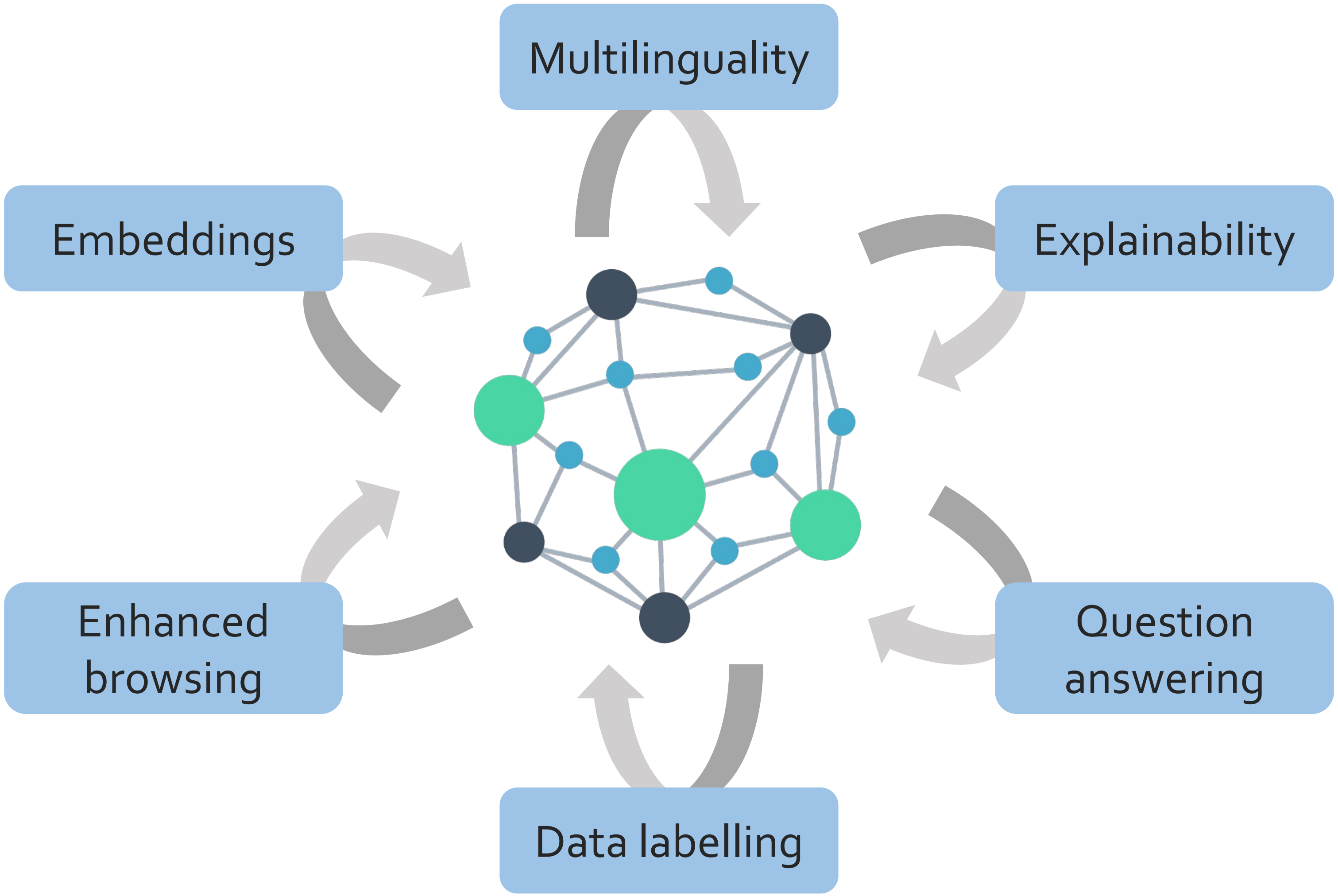
ana.iglesiasm@upm.es @_aieme 6/11/2024 ETSIInf

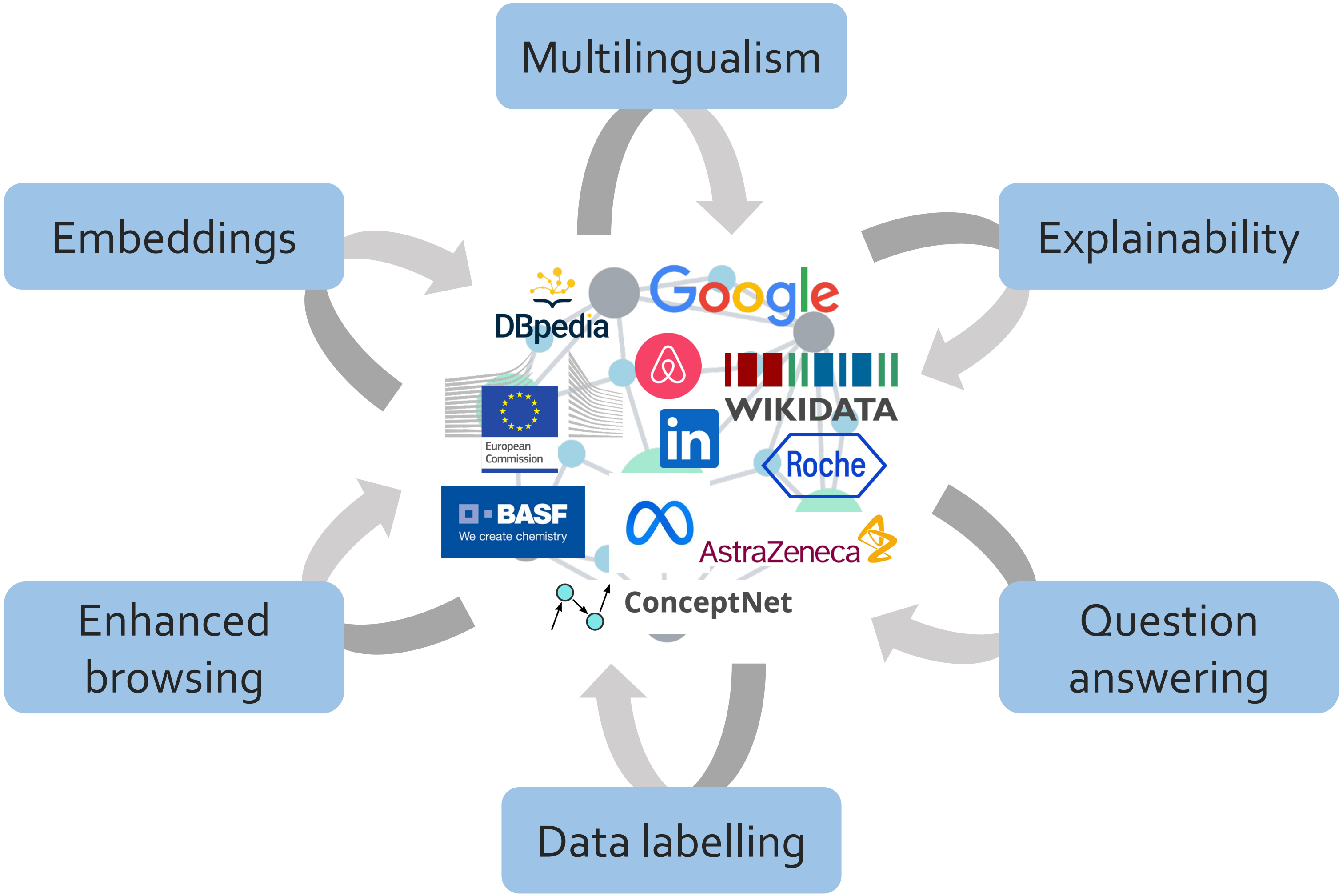
- **Introduction**
- State of the Art
- Thesis Objectives
- Contributions
- Conclusions

"A graph of data intended to accumulate and convey knowledge of the real world, whose nodes represent entities of interest and whose edges represent relations between these entities"

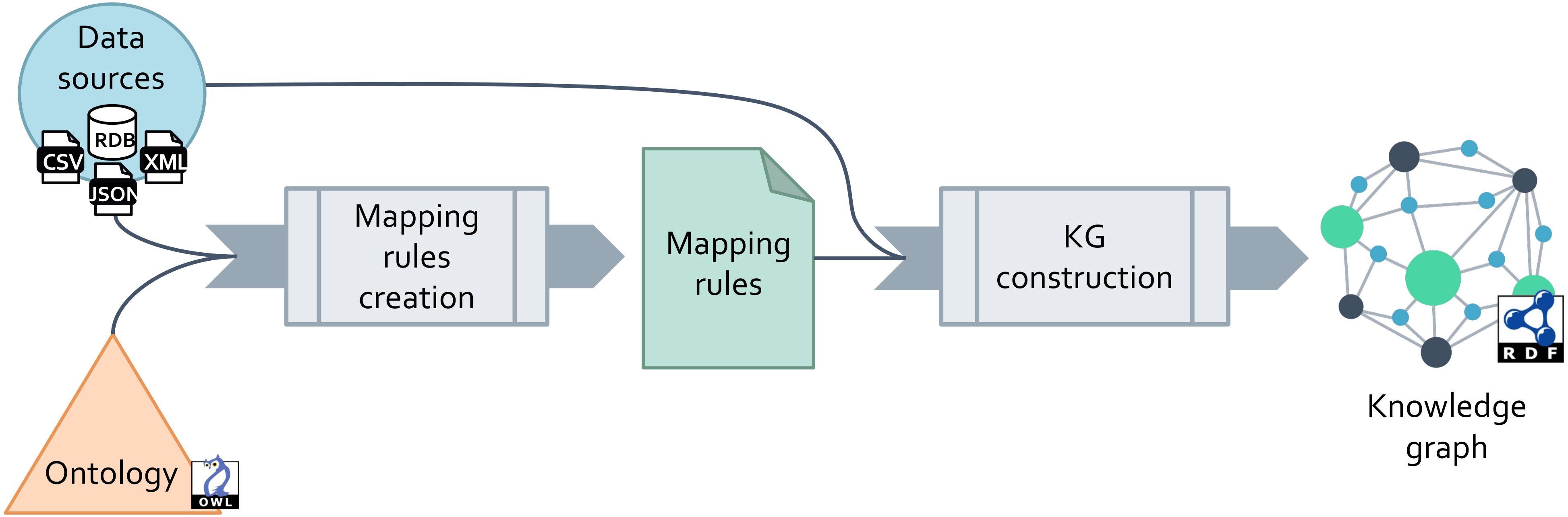


Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., Melo, G. D., Gutierrez, C., Kirrane, S., Labra-Gayo, J. E., Navigli, R., Neumaier, S., et al. (2021). Knowledge graphs. *ACM Computing Surveys (CSUR)*, 54(4), 1–37.


Multilinguality
Embeddings
Explainability
Enhanced browsing
Question answering
Data labelling


Multilingualism
Embeddings
Explainability
DBpedia
Google
WIKIDATA
European Commission
Roche
BASF
We create chemistry
AstraZeneca
ConceptNet
Enhanced browsing
Question answering
Data labelling

Knowledge Graph Construction



- Efficient
- Scalable
- Maintainable
- Robust
- Understandable
- Reproducible

Thesis objective
To improve the understanding (features and limitations) and operational management of declarative KG construction languages

Introduction

State of the Art

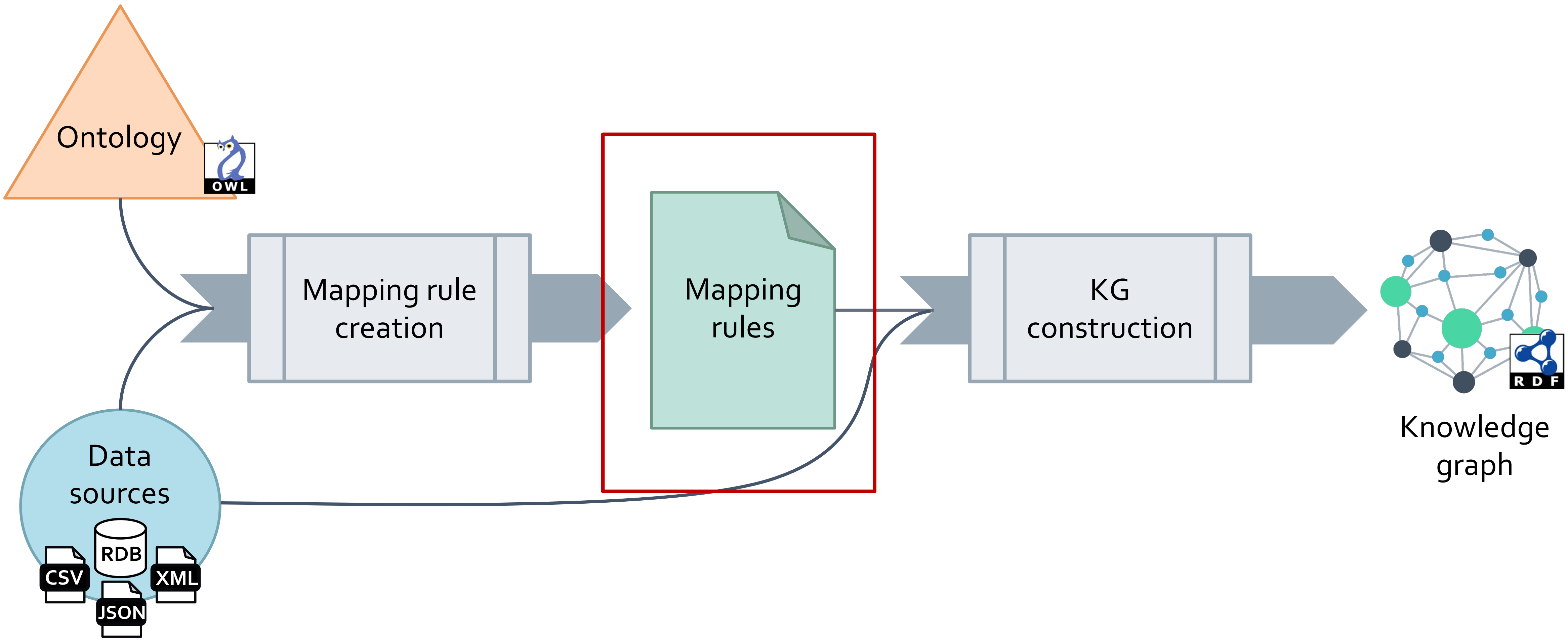
- Mapping languages for knowledge graph construction
- User-friendly approaches
- Knowledge graph life cycle

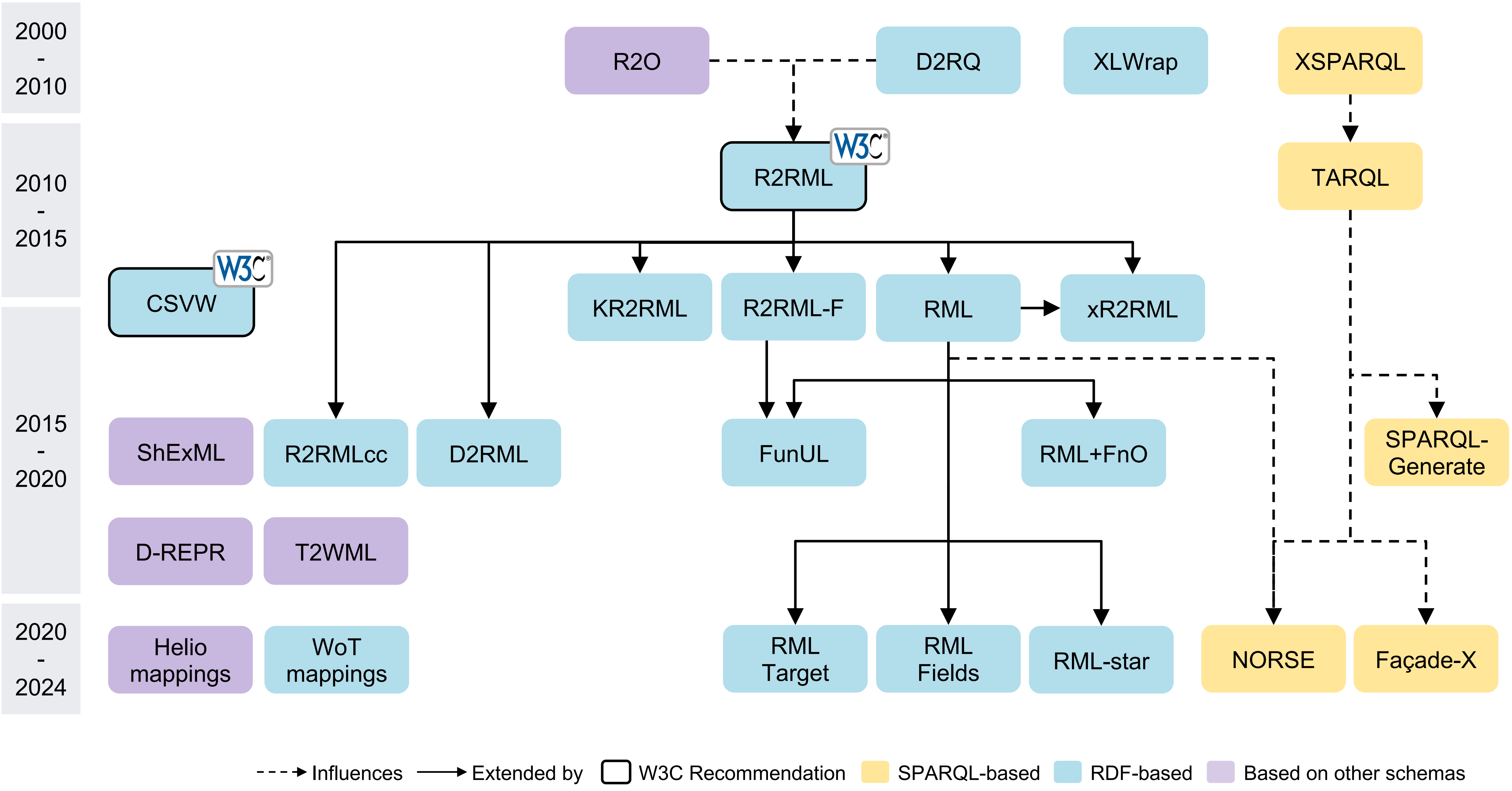
Thesis Objectives

Contributions

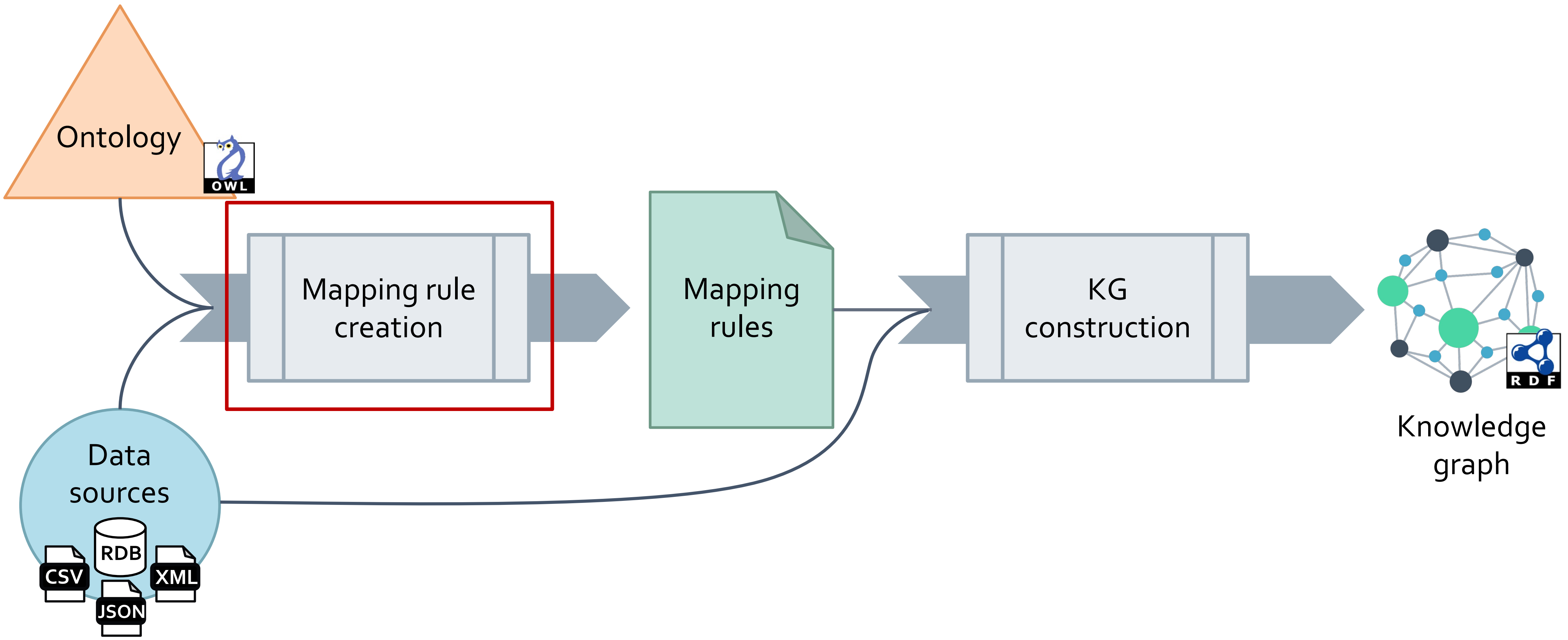
Conclusions

Knowledge Graph Construction

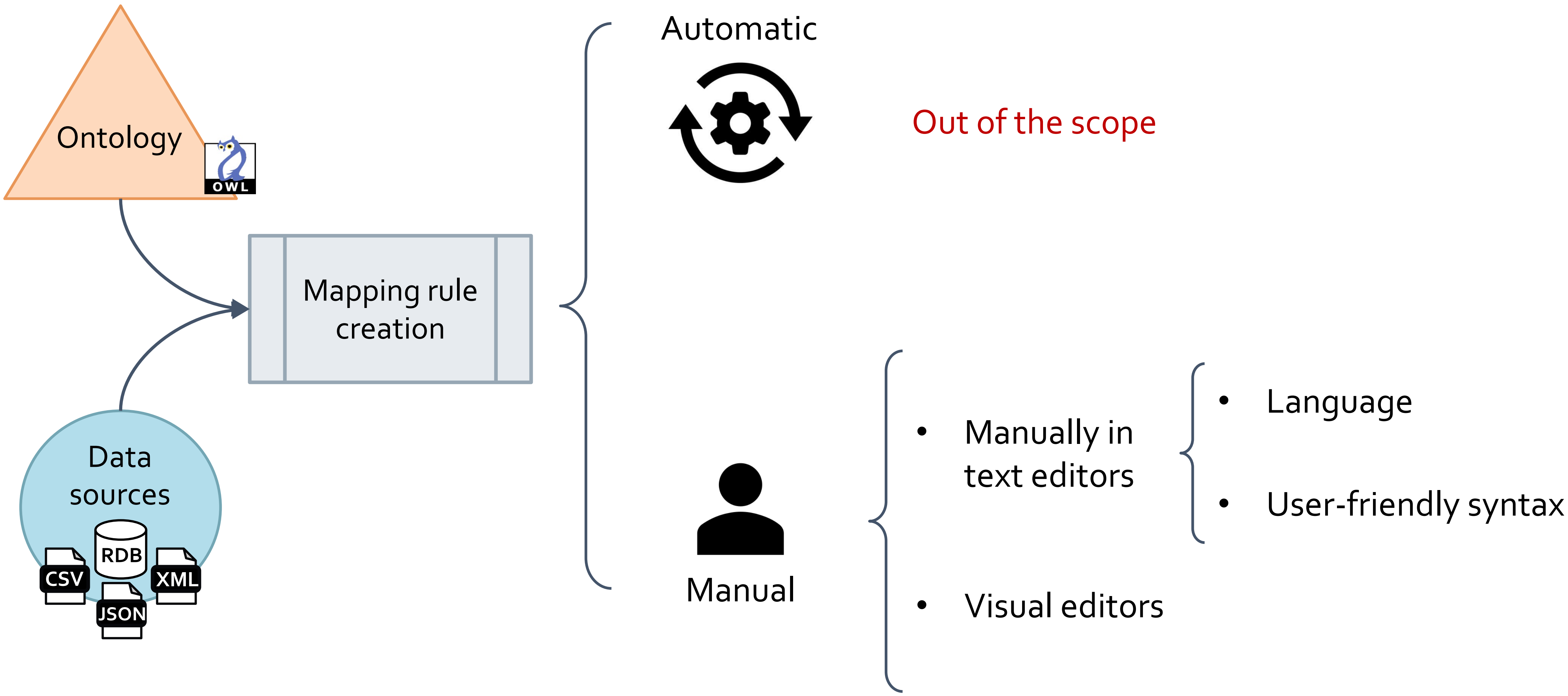



2000 - 2010
2010 - 2015
2015 - 2020
2020 - 2024
R2O
D2RQ
XLWrap
XSPARQL
R2RML
TARQL
CSVW
KR2RML
R2RML-F
RML
xR2RML
ShExML
R2RMLcc
D2RML
FunUL
RML+FnO
SPARQL-Generate
D-REPR
T2WML
Helio mappings
WoT mappings
RML Target
RML Fields
RML-star
NORSE
Façade-X
Influences
Extended by
W3C Recommendation
SPARQL-based
RDF-based
Based on other schemas

Mapping Rule Creation



Mapping Rule Creation



User-friendly Mapping Rule Creation

Text editor: Languages

```
@prefix rr: <http://www.w3.org/ns/r2rml#>.
@prefix ex: <http://example.com/ns#>.

<#athletesTM>
  rr:logicalTable [ rr:tableName "ATH" ];
  rr:subjectMap [
    rr:template "http://example.com/athlete/{RANK}";
    rr:class ex:Athlete;
  ];
  rr:predicateObjectMap [
    rr:predicate ex:name;
    rr:objectMap [ rr:column "ENAME" ];
  ] ;
  rr:predicateObjectMap [
    rr:predicate ex:rank;
    rr:objectMap [ rr:column "RANK" ];
  ] ;
  rr:predicateObjectMap [
    rr:predicate ex:mark;
    rr:objectMap [ rr:column "MARK" ];
  ].
```

R2RML

User-friendly Mapping Rule Creation

Text editor: User-friendly syntaxes

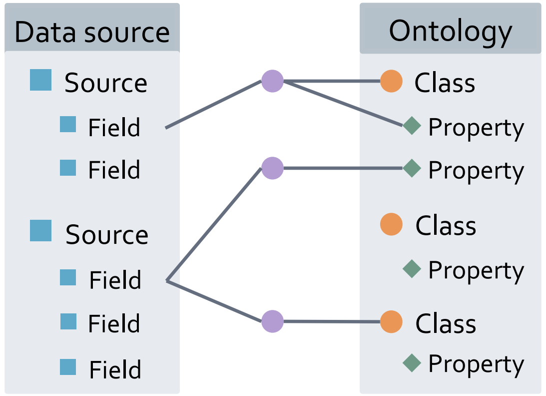
```
prefixes:
 ex: "http://example.com/ns#"

mappings:
 athletesTM:
  sources:
   - [ "data.json~jsonpath", "$.*" ]
  s: http://example.com/athlete/$(RANK)
  po:
   - [ a, ex:Athlete ]
   - [ ex:name, $(NAME) ]
   - [ ex:rank, $(RANK) ]
   - [ ex:mark, $(MARK) ]
```

- R2RML ▶ **SML** (Stadler et al., 2015)
- R2RML ▶ **Ontop obda** (Calvanese et al, 2017)
- rml.io ▶ **YARRRML** (Heyvaert et al., 2018)
- R2RML ▶ **SMS2**
- R2RML, rml.io, CSVW ▶ **XRM**

Visual Editors

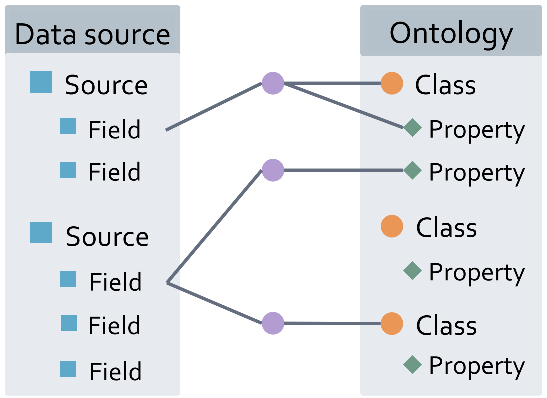
Tree layout



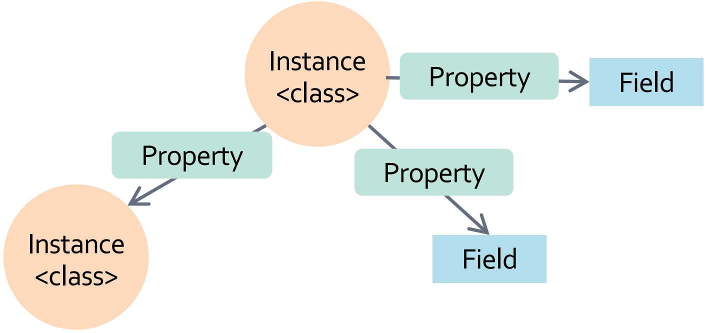
- ▶ **ODEMapster** (Barrasa & Gómez-Pérez, 2006)
- ▶ **Karma** (Gupta et al., 2012)
- ▶ **RBA** (Neto et al., 2013)

Visual Editors

Tree layout



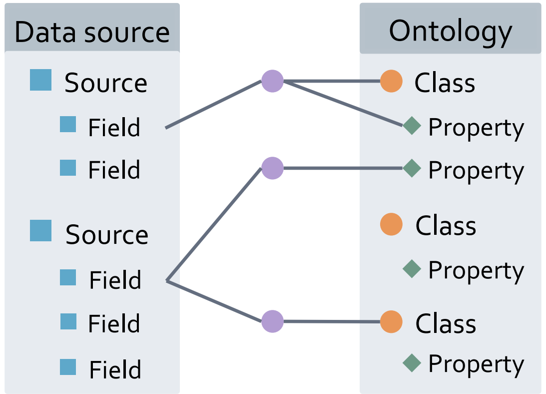
Graph layout



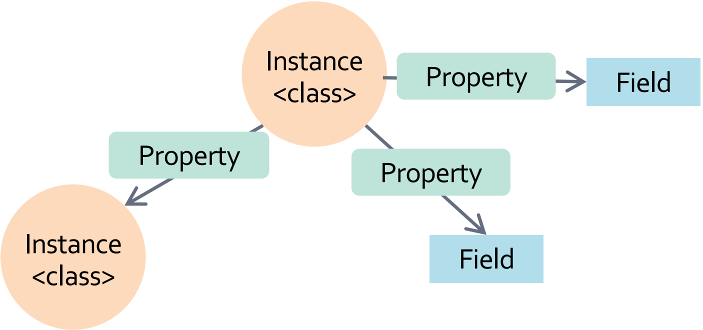
- R2RML ▶ Lembo et al., 2014
- ▶ **RMLEditor** (Heyvaert et al., 2016)
- R2RML ▶ **SQuaRE** (Blinkiewicz & Bak, 2016)
- R2RML ▶ **Map-On** (Sicilia et al., 2017)
- R2RML ▶ **gra.fo**
- SMS2 ▶ **Stardog designer**

Visual Editors

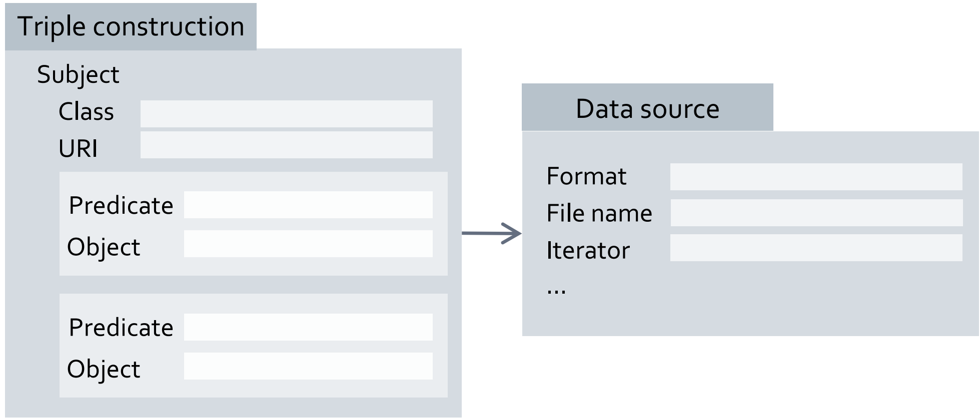
Tree layout



Graph layout

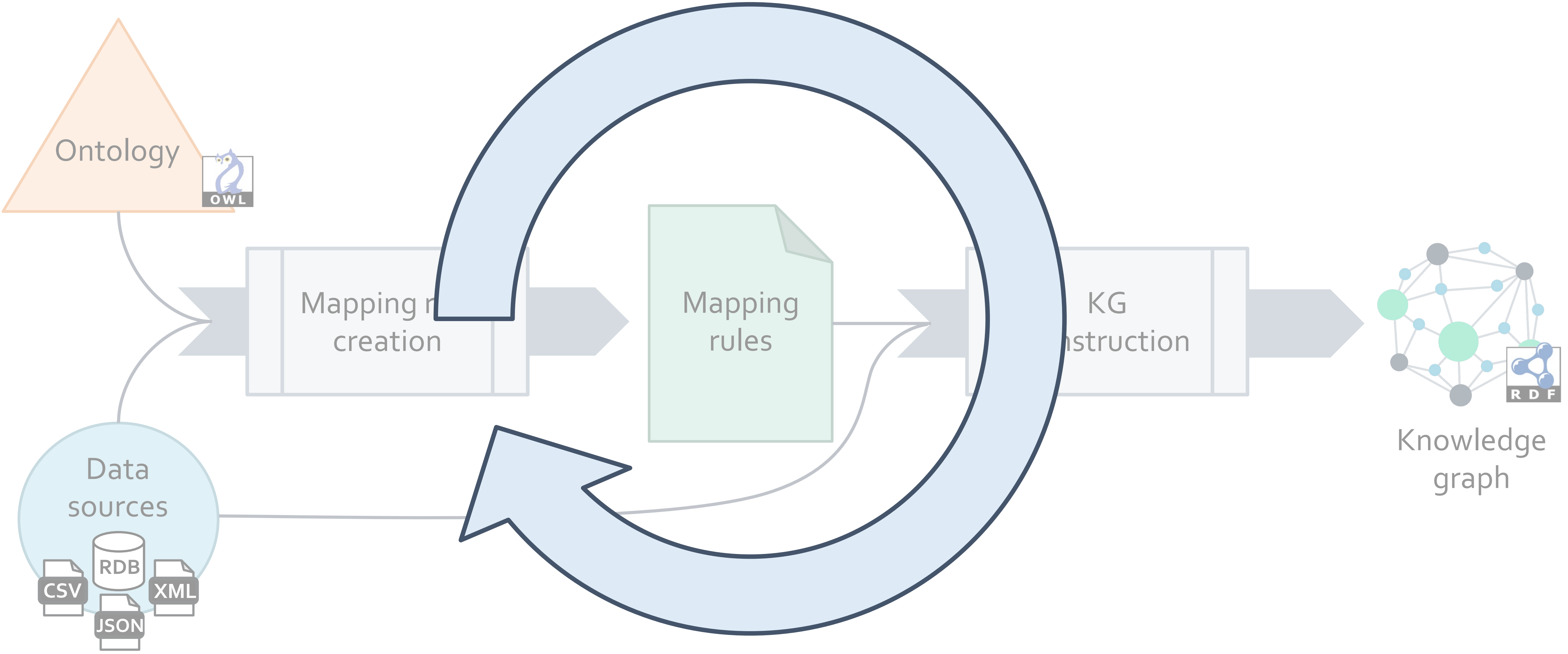


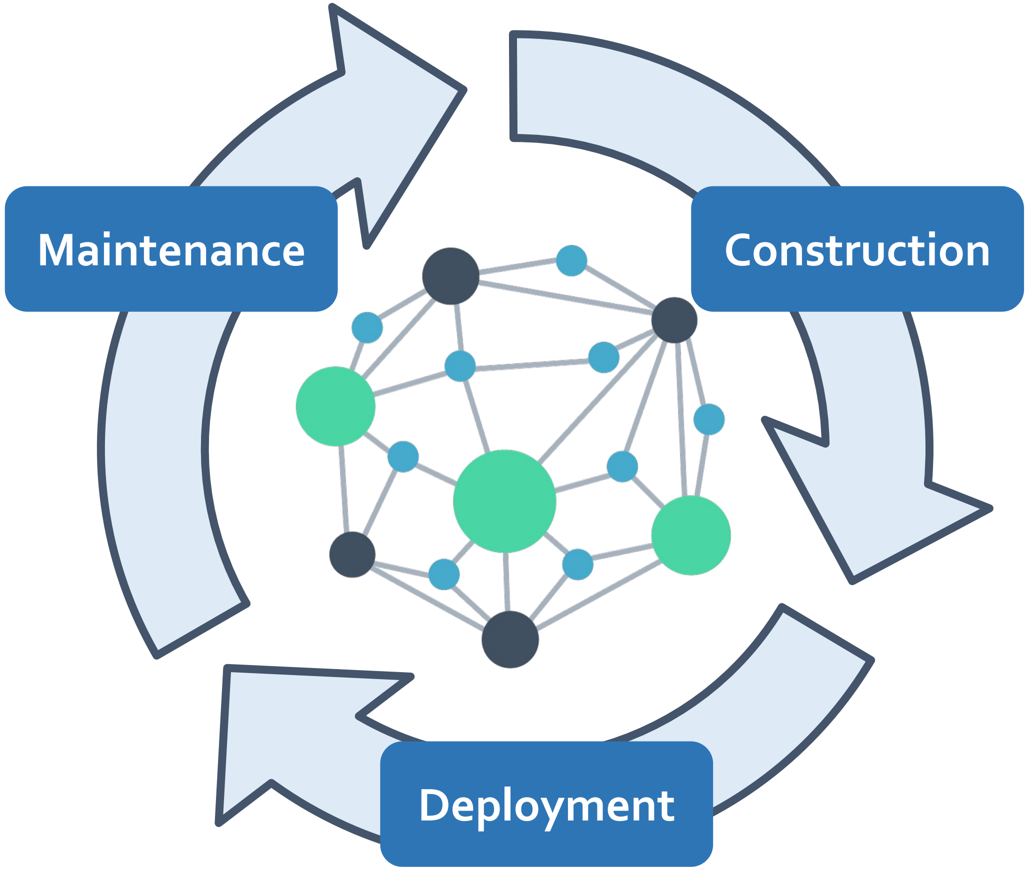
Building blocks



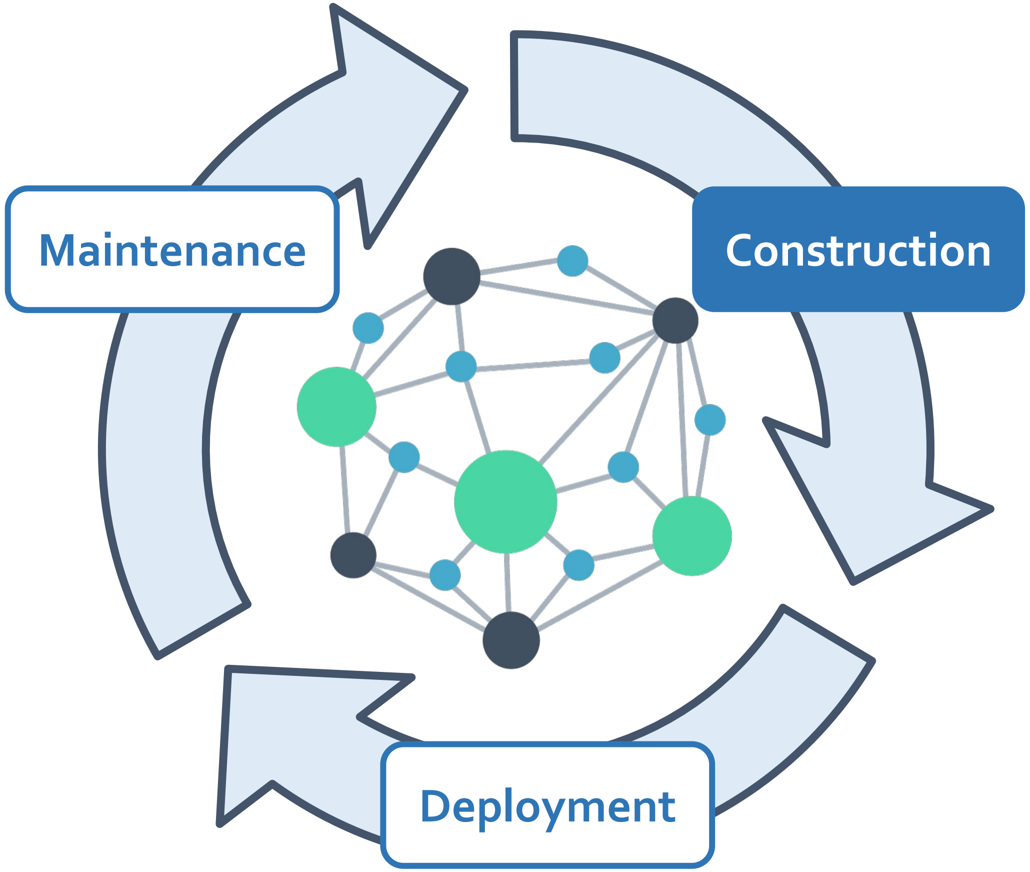
- ▶ Sengupta et al., 2013
- ▶ **OntopPro** (Calvanese et al., 2017)
- ▶ **Juma** (Crotti Junior et al., 2017)
- ▶ **RMLx** (Aryan et al., 2017)
- ▶ **Ontopic Studio**
- ▶ **Eccenca Corporate Memory**
- ▶ **OpenRefine**

Knowledge Graph Lifecycle



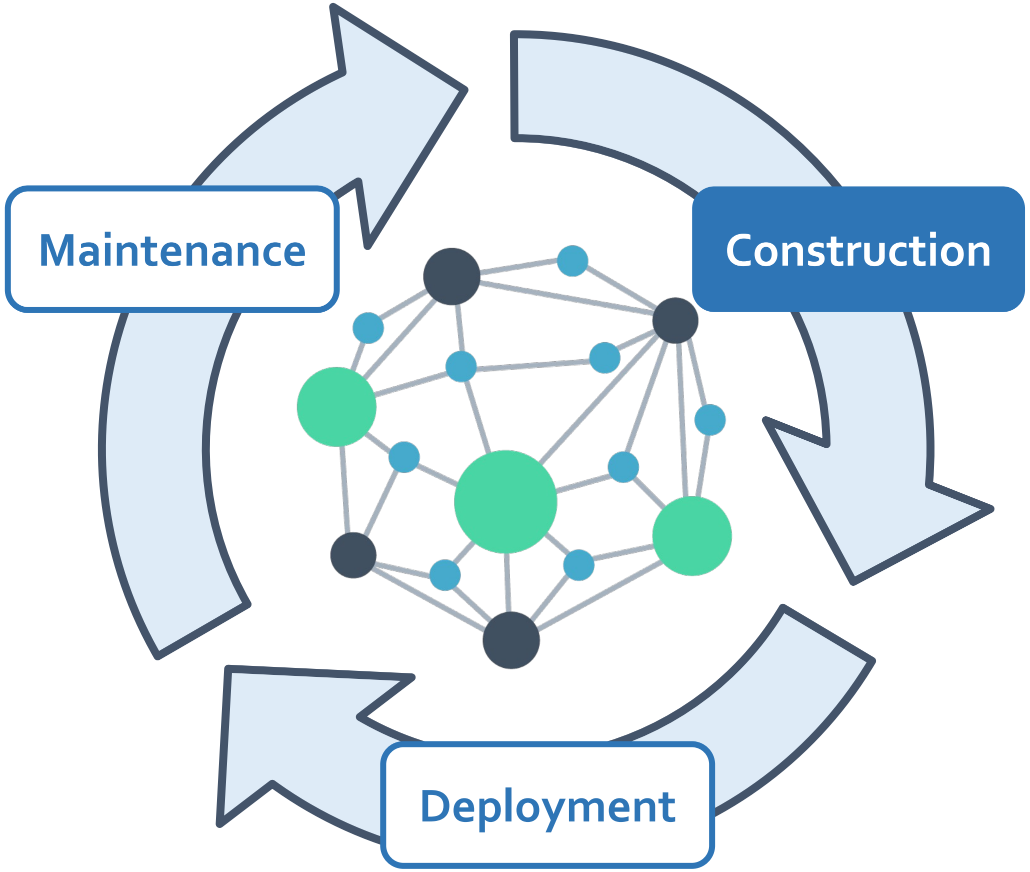

Maintenance
Construction
Deployment

Mapping Rules Support in the KG Construction Step



- Increasing mapping **expressiveness**
- **Facilitating** KG construction

Mapping Rules Support in the KG Construction Step



- Increasing mapping **expressiveness**
- **Facilitating** KG construction
- Resource usage **optimization** in KGC engines

Arenas-Guerrero, J., Chaves-Fraga, D., Toledo, J., Pérez, M. S., & Corcho, O. (2024). Morph-KGC: Scalable Knowledge Graph Materialization with Mapping Partitions. *Semantic Web*, *15*, 1–20.

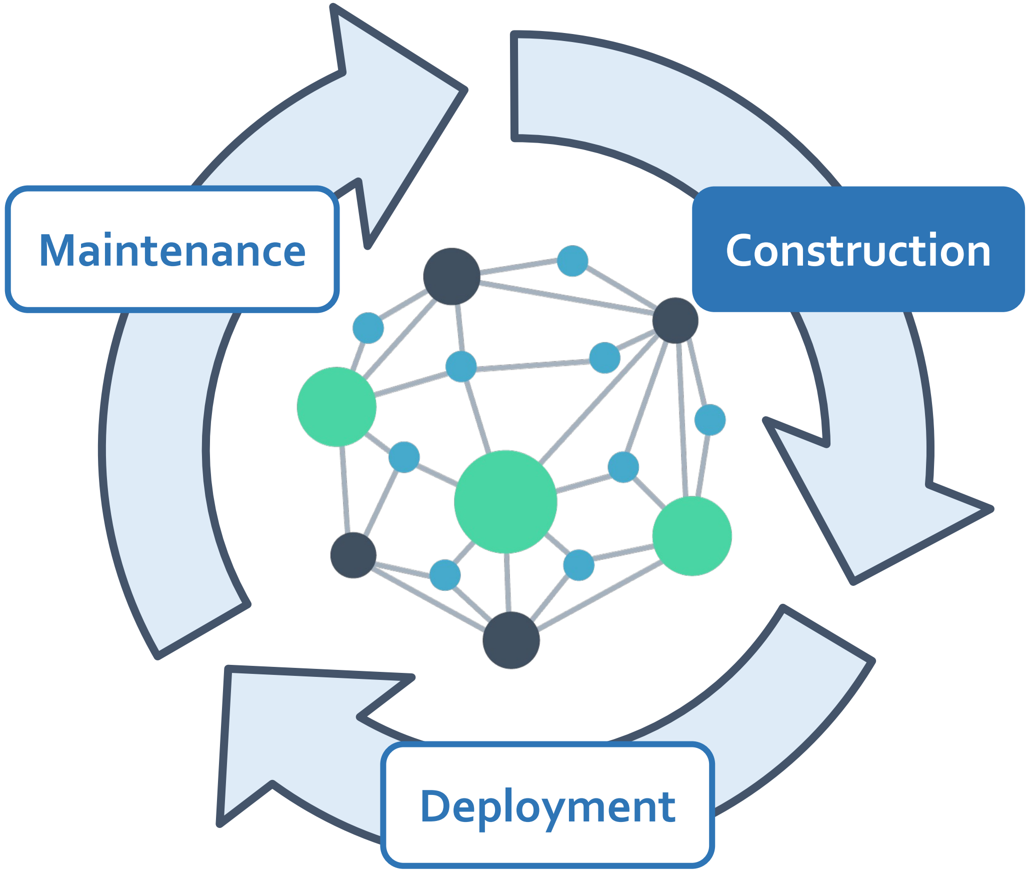
Iglesias, E., Jozashoori, S., & Vidal, M.-E. (2023). Scaling up Knowledge Graph Creation to Large and Heterogeneous Data Sources. *Journal of Web Semantics*, *75*, 100755.

Jozashoori, S., & Vidal, M.-E. (2019). MapSDI: A scaled-up semantic data integration framework for knowledge graph creation. *CoopIS, ODBASE, C&TC 2019, Proceedings*, 58–75.

Xiao, G., Lanti, D., Kontchakov, R., Komla-Ebri, S., Güzel-Kalaycı, E., Ding, L., Corman, J., Cogrel, B., Calvanese, D., & Botoeva, E. (2020). The virtual knowledge graph system ontop. *International Semantic Web Conference*, 259–277.

Chaves-Fraga, D., Endris, K. M., Iglesias, E., Corcho, O., & Vidal, M.-E. (2019). What are the Parameters that Affect the Construction of a Knowledge Graph? *CoopIS, ODBASE, C&TC 2019, Proceedings*, 695–713.

Mapping Rules Support in the KG Construction Step

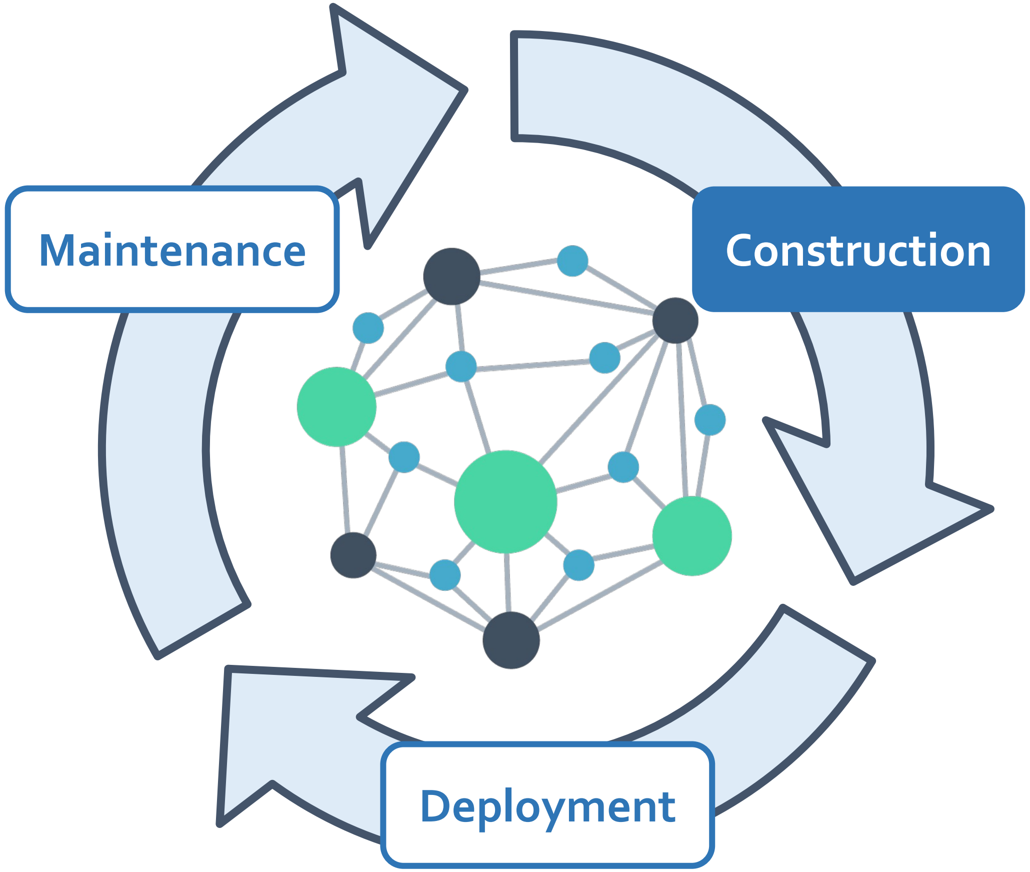


- Increasing mapping **expressiveness**
- **Facilitating** KG construction
- Resource usage **optimization** in KGC engines
- **Metadata annotation**

Chaves-Fraga, D., Ruckhaus, E., Priyatna, F., Vidal, M.-E., & Corcho, O. (2021). Enhancing virtual ontology-based access over tabular data with Morph-CSV. *Semantic Web*, *12* (6), 869–902.

Vidal, M.-E., Sakor, A., Jozashoori, S., Niazmand, E., Purohit, D., Iglesias, E., Aisopos, F., Vogiatzis, D., Menasalvas, E., Gonzalez, A. R., et al. (2023). Knowledge Graphs for Enhancing Transparency in Health Data Ecosystems. *Semantic Web*, *14* (5), 943–976.

Mapping Rules Support in the KG Construction Step



- Increasing mapping **expressiveness**
- **Facilitating** KG construction
- Resource usage **optimization** in KGC engines
- **Metadata annotation**
- Data **processing** and cleaning

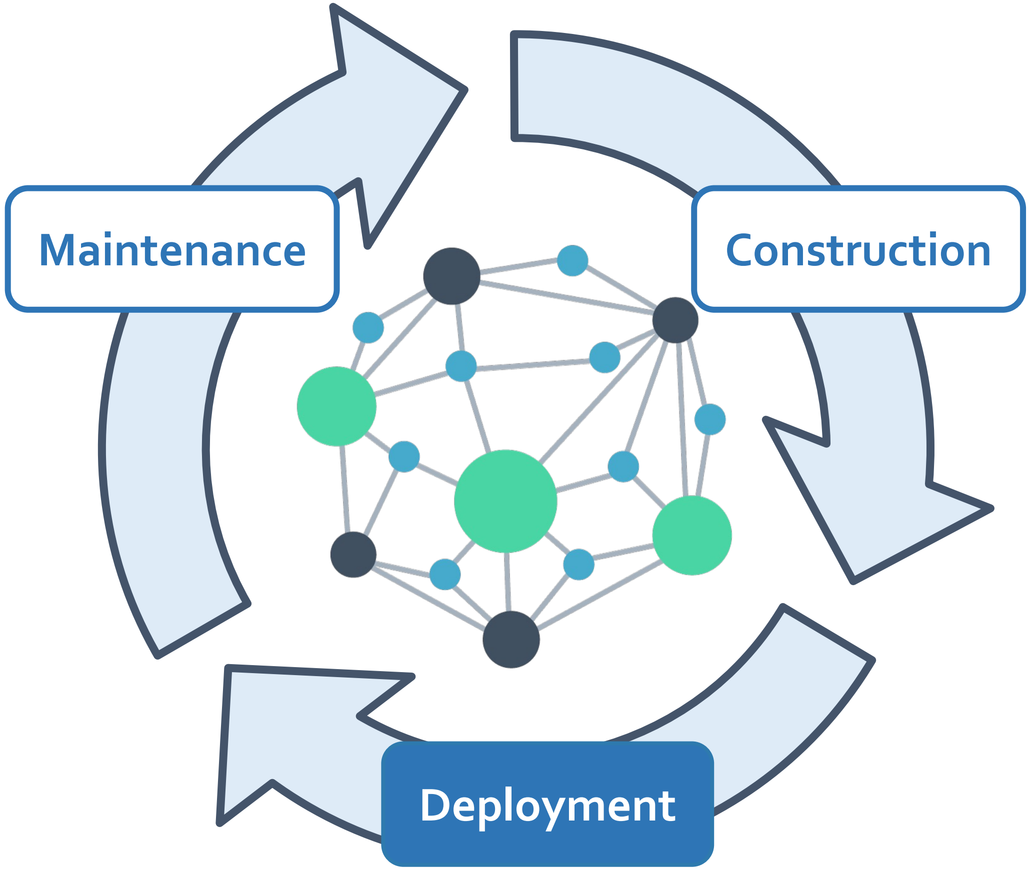
Crotti Junior, A., Debruyne, C., Brennan, R., & O'Sullivan, D. (2016). FunUL: a method to incorporate functions into uplift mapping languages. *iiWAS '16*, 267–275.

De Meester, B., Maroy, W., Dimou, A., Verborgh, R., & Mannens, E. (2017). Declarative data transformations for linked data generation: The case of DBpedia. *ESWC 2017, Proceedings, Part I, 10250*, 33–48.

Debruyne, C., & O'Sullivan, D. (2016). R2RML-F: towards sharing and executing domain logic in R2RML mappings. *LDOW@WWW 2016, 1593*.

Jozashoori, S., Chaves-Fraga, D., Iglesias, E., Vidal, M.-E., & Corcho, O. (2020). FunMap: Efficient Execution of Functional Mappings for Knowledge Graph Creation. *ISWC 2020, Proceedings, Part I*, 276–293.

Mapping Rules Support in the KG Deployment Step

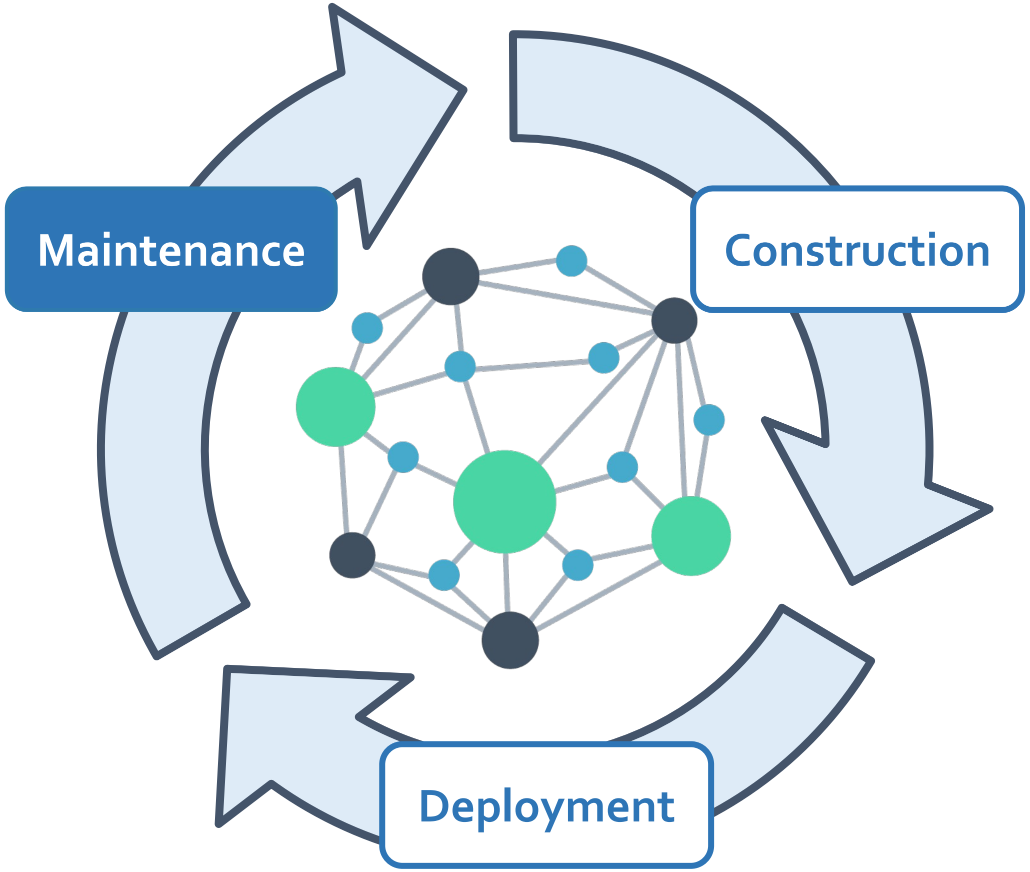


- **Output** graph description

Van Assche, D., Haesendonck, G., De Mulder, G., Delva, T., Heyvaert, P., De Meester, B., & Dimou, A. (2021). Leveraging Web of Things W3C Recommendations for Knowledge Graphs Generation. *ICWE 2021*, 337–352.

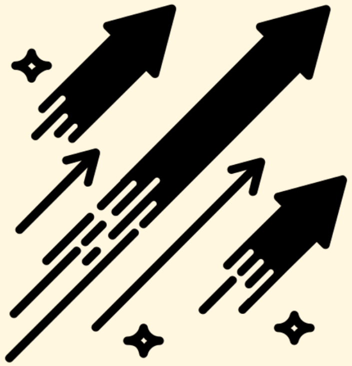
Lefrançois, M., Zimmermann, A., & Bakerally, N. (2017). A SPARQL extension for generating RDF from heterogeneous formats. *ESWC 2017, Proceedings, Part I*, 35–50.

What about in the maintenance step?



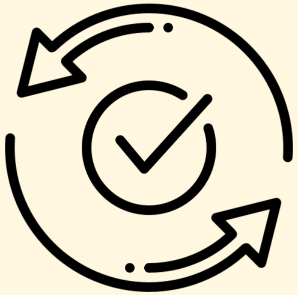
Limitations of the State of the Art

Lack of in-depth assessment of expressiveness and capabilities of mapping languages



Mapping languages are outdated w.r.t. new needs in knowledge graph construction

Limited access for mapping rule writing depending on the user profile



Missing assessment on the role of declarative mapping rules in knowledge graph evolution

Introduction

State of the Art

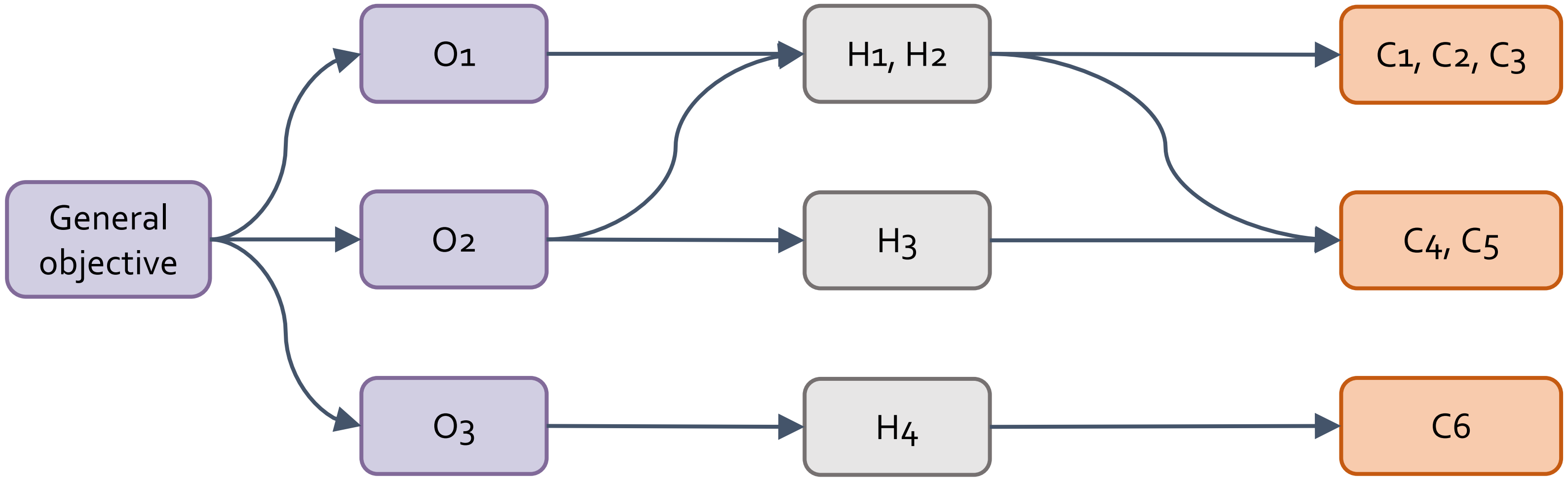
Thesis Objectives

- Methodology
- Objectives and contributions
- General overview with assumptions, hypothesis and restrictions

Contributions

Conclusions

Identification of Limitations → Definition of objectives → Definition of hypothesis → Development of contributions



O1: To analyze the needs for KG construction from heterogeneous data sources to support advances in mapping languages

H1: Current mapping languages lack some expressiveness to construct KGs from heterogeneous data sources for any use case

H2: It is possible to update current mapping languages with new features that address the current needs in the construction of KGs

C1: Comparison framework of current mapping languages

C2: Definition and implementation of requirements for KG construction

C3: Development of new features for mapping languages up to date with current KG construction needs

O2: To help knowledge engineers and domain experts with building mappings in a user-friendly manner

H3: Writing mapping rules in spreadsheet environments facilitates the process for practitioners of diverse backgrounds for mapping writing, whilst reducing errors

C4: Design of a user-friendly approach for writing mapping rules based on spreadsheets

H1: Current mapping languages lack some expressiveness to construct KGs from heterogeneous data sources for any use case

H2: It is possible to update current mapping languages with new features that address the current needs in the construction of KGs

C5: Update of a user-friendly syntax for mapping languages with extended features

O3: To assess declarative KG construction technologies in terms of their benefits for KG evolution

H4: Declarative KG construction technologies brings benefits in the evolution of knowledge graphs within their full life cycle

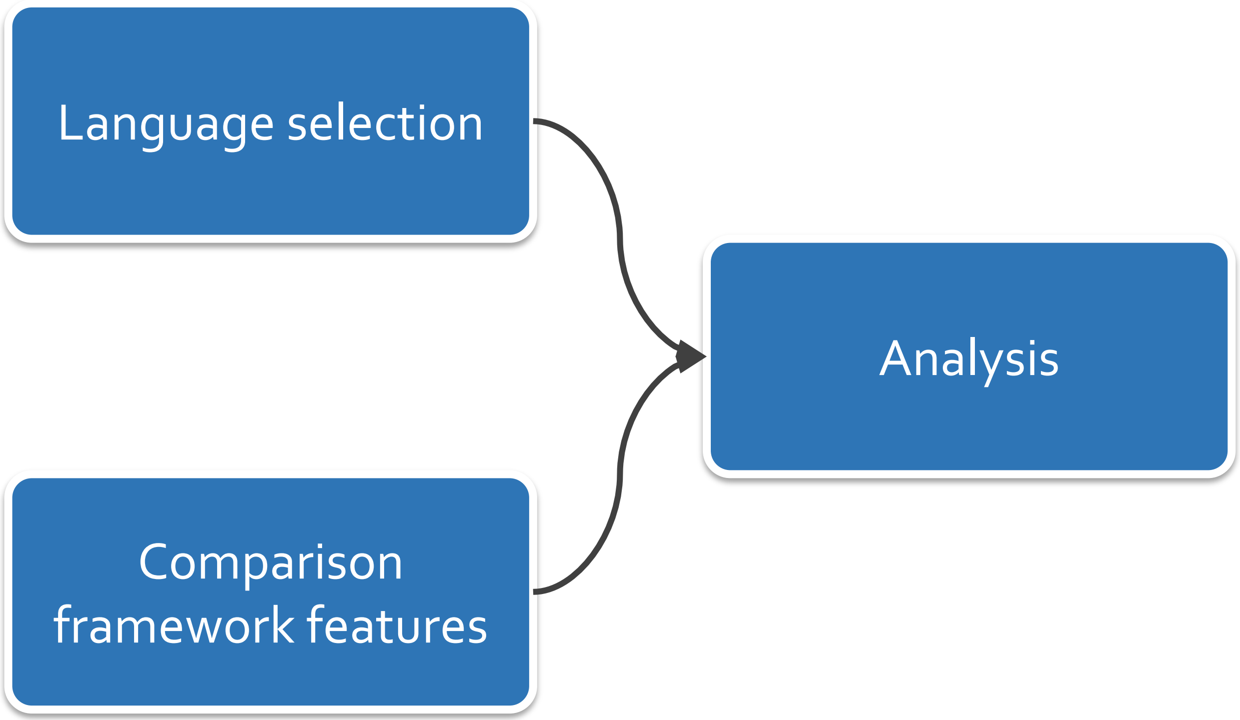
C6: Analysis of scenarios where declarative KG construction technologies support KG refactoring

- Introduction
- State of the Art
- Thesis Objectives
- **Contributions**
  - **O1. Understanding, representing and extending the expressiveness of declarative mapping languages**
  - O2. Enhancing the creation of mapping rules
  - O3. Representation impact on knowledge graph refactoring
- Conclusions

C1 Comparison framework of current mapping languages

Iglesias-Molina, A., Cimmino, A., Ruckhaus, E., Chaves-Fraga, D., García-Castro, R., & Corcho, O. (2024). An ontological approach for representing declarative mapping languages. *Semantic Web*, *15* (1), 191–221.

Methodology



- 17 selected languages
  - Widely **used**, **relevant** and/or include novel or **unique** features
  - **Original languages**, not their corresponding user-friendly serializations
  - Not tailored for a specific use case
- Languages analyzed based on their **official documentation**
  - Specific implementations or extensions not in official documentation are not considered

Classification	Language
RDF-based	R2RML
	RML (2014)
	KR2RML
	xR2RML
	R2RML-F
	FunUL
	XLWrap
	CSVW
	D2RML
	R2RML collections and containers
SPARQL-based	XSPARQL
	TARQL
	SPARQL-Generate
	Facade-X
Based on other schemas	ShExML
	Helio
	D-REPR

Comparison framework features

Data source description

Retrieval of data

- Streams
- Synchronous
- Asynchronous

Describing data sources

- Security terms
- Encoding
- MIME Type
- Features describing data
- Retrieval protocol
- Data format

Triple generation

Subject

- Constant generation
- Dynamic generation
  - RDF Resource kinds
  - Data reference
  - Data sources
  - Hierarchy iteration
  - Functions

Predicate

- Constant generation
- Dynamic generation
  - RDF Resource kinds
  - Data reference
  - Data sources
  - Hierarchy iteration
  - Functions

Object

- Constant generation
- Dynamic generation
  - RDF Resource kinds
  - Data reference
  - Data sources
  - Hierarchy iteration
  - Functions
- Datatype and language

General features

Statements

- Assign to named graphs
- Retrieve data from one source
- Retrieve data from one or more sources
- Allow conditions

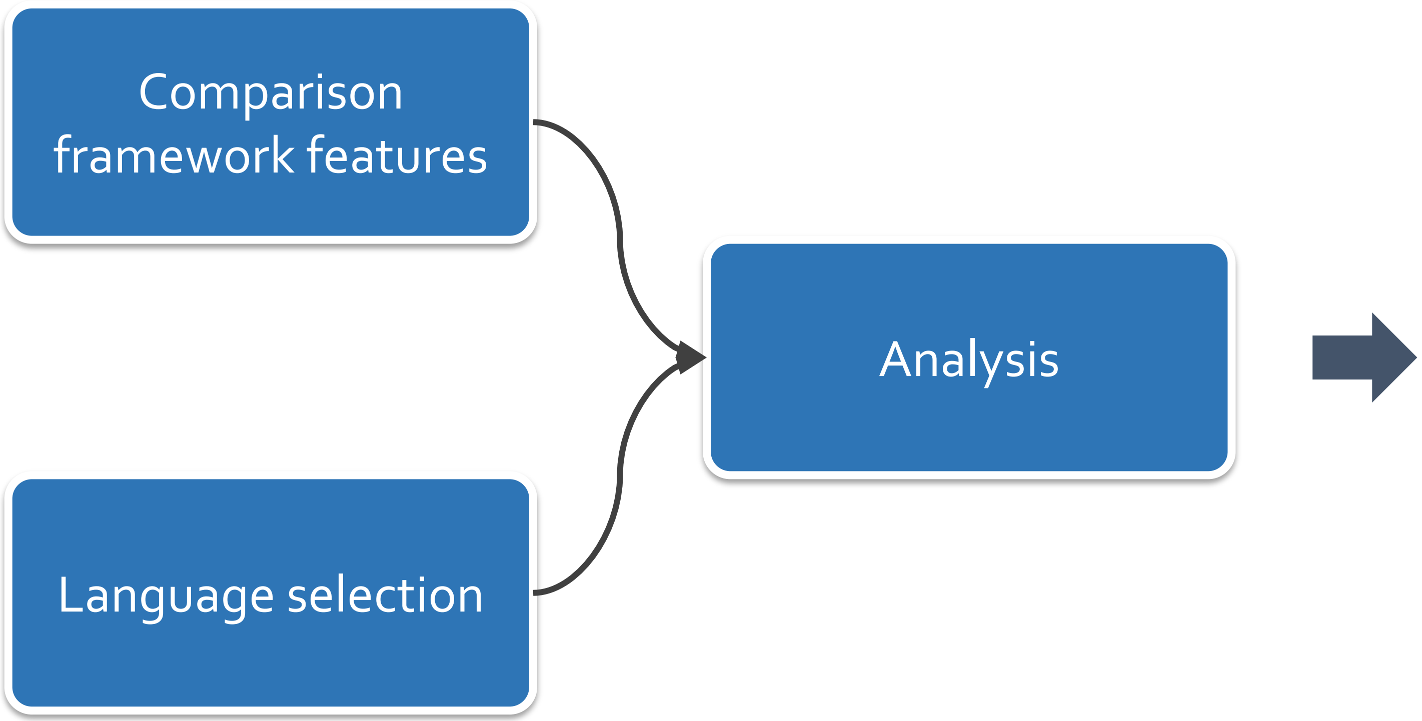
Linking rules

- One data reference
- One or more data references
- No condition to link
- Link with one condition
- Link with one or more conditions
- Use only equal function
- Use any similarity function

Functions

- Cardinality
- Nested functions
- Functions belong to a specification
- Declare own functions

Methodology



- High **heterogeneity** in expressiveness
- **Different capabilities** of same language in different implementations
- Languages **do not cover all** features

C2

Definition and implementation of requirements for KG construction

Iglesias-Molina, A., Cimmino, A., Ruckhaus, E., Chaves-Fraga, D., García-Castro, R., & Corcho, O. (2024). An ontological approach for representing declarative mapping languages. *Semantic Web*, *15* (1), 191–221.

Methodology

Comparison framework → Set of requirements for KG construction → Implementation in an ontology

- 17 languages
- 43 analyzed features organized in 3 categories

- 28 core requirements

- Conceptual Mapping ontology
- Following the LOT methodology

Requirements Specification

Data source description

ID	Requirement
cm-r1	A mapping can describe data sources retrieved synchronously, asynchronously and as streams
cm-r2	A data source can describe data in different formats specifying its Media Type
cm-r6	A data source can specify the encoding
cm-r9	Multiple data sources can be described with combined frames and nested frames
cm-r10	A source frame describes exactly one data source
cm-r26	A frame can have nested frames to access multi-value references

Data source access

ID	Requirement
cm-r3	A data source may have a specified data access service
cm-r4	A data access service can specify security terms
cm-r5	A data access service can specify up to one retrieval protocol

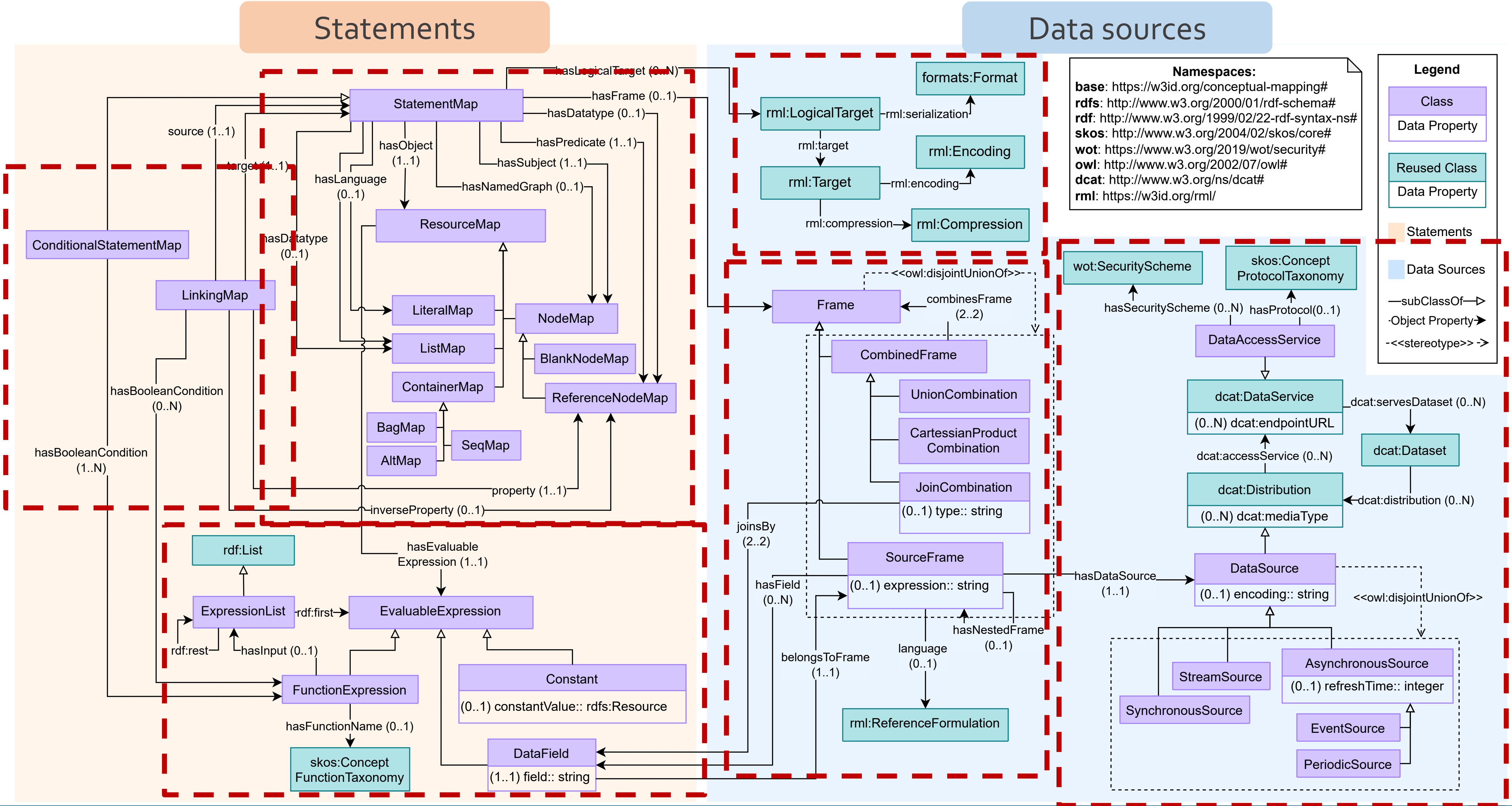
Statement generation

ID	Requirement
cm-r7	A mapping can describe data sources and their access
cm-r22	A statement map can be subject to a condition
cm-r23	A statement map can be linked to another statement map with none, one or several conditions
cm-r24	The condition to link statements may be any boolean condition

Triple elements

ID	Requirement
cm-r8	A mapping specifies statement maps to declare the transformation of frames into RDF triples
cm-r11	A statement map can describe constant and dynamic subjects, predicates, objects
cm-r12	A statement map can describe constant and dynamically named graph, datatype and language tag
cm-r13	A resource map can reference one or more data fields from one or more data sources
cm-r14	A resource map may contain data with different levels of iteration of a source
cm-r15	A resource map may contain functions
cm-r16	A subject can only be expressed as a blank node or an IRI
cm-r17	A predicate can only be expressed as an IRI
cm-r18	An object can only be expressed as a blank node, an IRI, a literal, a container or a list
cm-r19	A named graph can only be expressed as an IRI or blank node
cm-r20	A datatype tag can only be expressed as an IRI or a List
cm-r21	A language tag can only be expressed as a Literal or a List
cm-r25	A function may have nested functions
cm-r27	A statement map can have objects of type literal using data from different sources using combined frames
cm-r28	A statement map can have datatype and language tags using data from different sources using combined frames

Implementation: Conceptual Mapping ontology model



Technical implementation, publication and maintenance

Implementation

- Conceptualization with **diagrams** → Chowlk
- Encoded in **OWL2** →
  - Definition of **concepts and relationships**
- **Evaluation:**
  - Pitfalls → oops!
  - Inconsistencies → HermiT
  - Requirements coverage → THEMIS
  - Peer-reviewed by experts

Publication

- **Documentation** → WIDOCO
- **Publication** → OnToology
  - **W3ID**: w3id.org/conceptual-mapping

Maintenance

- Supported by the **GitHub issue tracker**

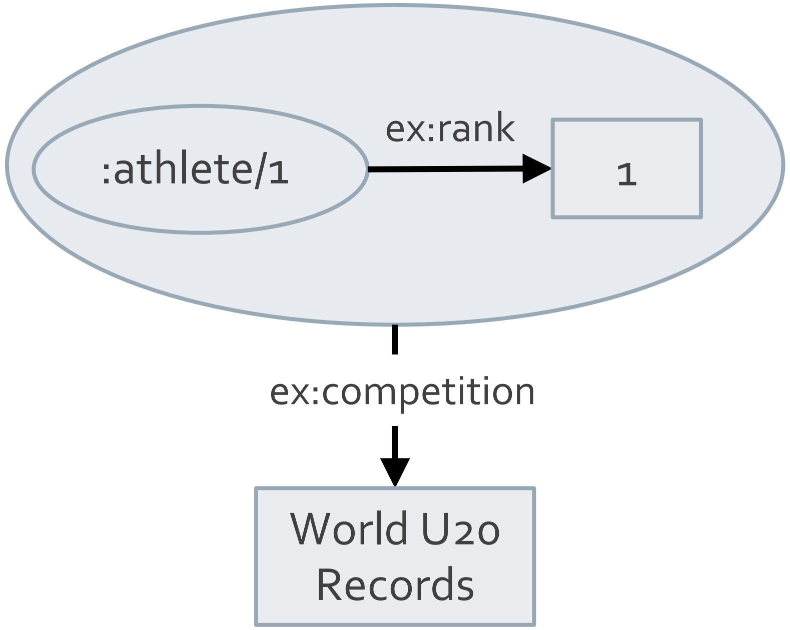
https://github.com/oeg-upm/Conceptual-Mapping

C3

Development of new features for mapping languages up to date with current KG construction needs

Delva, T., Arenas-Guerrero, J., **Iglesias-Molina, A**., Corcho, O., Chaves-Fraga, D., & Dimou, A. (2021). RML-star: A Declarative Mapping Language for RDF-star Generation. *ISWC 2021: Posters, Demos, and Industry Tracks, Virtual Event, October 24–28, 2021.*

Arenas-Guerrero, J.*, **Iglesias-Molina, A.***, Chaves-Fraga, D., Garijo, D., Corcho, O., & Dimou, A. (2024). Declarative generation of RDF-star graphs from heterogeneous data. *Semantic Web*, in press.




```
<< :athlete/1 ex:rank 1 >> ex:competition “World U20 Records”
```

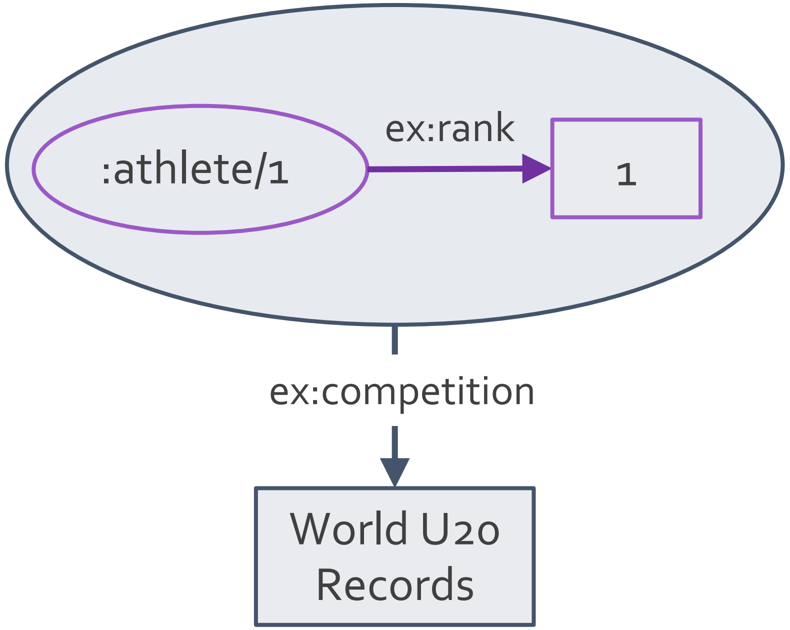
- Extension of the RDF syntax
- Part of the RDF 1.2 release (under development)

Mapping languages need to adapt their syntax accordingly to produce RDF-star graphs

RML-star

http://w3id.org/rml/star/ https://doi.org/10.5281/zenodo.7919845 https://github.com/kg-construct/rml-star

```
ID, RANK, COMP
1 , 1    , World U20 Records
```



RML-star mapping

```
<#rankTM>
  a rml:NonAssertedTriplesMap ;
  rml:logicalSource :ranks ;
  rml:subjectMap [
    rml:template ":athlete{ID}"];
  rml:predicateObjectMap [
    rml:predicate ex:rank ;
    rml:objectMap [
      rml:reference “RANK” ] ] .
```

```
<#competitionTM>
  a rml:TriplesMap ;
  rml:logicalSource :ranks ;
  rml:subjectMap [
    rml:quotedTriplesMap <#rankTM>];
  rml:predicateObjectMap [
    rml:predicate ex:competition ;
    rml:objectMap [
      rml:reference “COMP” ] ] .
```

```
<< :athlete/1 ex:rank 1 >> ex:competition “World U20 Records”
```

http://w3id.org/rml/star/ https://doi.org/10.5281/zenodo.7919845 https://github.com/kg-construct/rml-star

RML-star validation

RML-star test cases

- **16 test cases**
- Adapted from the **RDF-star test cases** and extended
- Two tests per case:
  - **One input** data source
  - **Two or more input** data sources

Use cases

- **Scientific software metadata extraction** (SOMEF)
  - 79 Triples Maps
  - 36 Quoted Triples Maps
- **Biomedical research literature** (SemMedDB)
  - 10 Triples Maps
  - 5 Quoted Triples Maps

Tested in morph-kgc^{star}

https://doi.org/10.5281/zenodo.6518802

- **High heterogeneity** in mapping languages and their capabilities
- **Limitations** for complex and new use cases

Comparison framework for mapping languages

Fine-grained **analysis** of languages expressiveness that can **help practitioners** to find **language suitable for each need**

Set of requirements for KG construction

Openly published, available for supporting **new developments** (e.g. mapping translation or knowledge conversion)

RML-star extension for constructing RDF-star graphs

Increasing support for easier **statement annotation**, tested and implemented in different engines

- Introduction
- State of the Art
- Thesis Objectives
- **Contributions**
  - O1. Understanding, representing and extending the expressiveness of declarative mapping languages
  - **O2. Enhancing the creation of mapping rules**
  - O3. Representation impact on knowledge graph refactoring
- Conclusions

C4

Design of a user-friendly approach for writing mapping rules based on spreadsheets

Iglesias-Molina, A., Chaves-Fraga, D., Priyatna, F. and Corcho, O. (2019) Towards the Definition of a Language-independent Mapping Template for Knowledge Graph Creation. *Workshop on Capturing Scientific Knowledge co-located with K-CAP 2019, November 19–21, 2019, Marina del Rey, US.*

Iglesias-Molina, A., Pozo-Gilo, L., Doña, D., Ruckhaus, E., Chaves-Fraga, D., & Corcho, O. (2020). Mapeathor: Simplifying the Specification of Declarative Rules for Knowledge Graph Construction. *ISWC 2020 Demos and Industry Tracks, Virtual Event, November 1–6, 2020*.

Visual Editors

Manually writing mappings (language or syntax)

Mapping rules in spreadsheets

- **Familiar** environment
- **Wide support** (Excel, GSpreadhseets)
- **No syntax**, only essential elements
- Convenient for **large mappings**

Prefix sheet

Prefix	URI
ex	http://ex.com/
@base	http://ex.com/base/

Subject sheet

ID	Class	URI	Graph
PERSON	ex:Person	http://ex.com/Person/{ID}	http://ex.com/graph
SPORT	ex:Sport	http://ex.com/Sport/{ID}	

Source sheet

ID	Feature	Value
PERSON	source	data/people.csv
PERSON	format	CSV
SPORT	source	data/sports.csv
SPORT	format	CSV

Function sheet

FunctionID	Feature	Value
<date-fun>	executes	grel:toDate
<date-fun>	returns	grel:output_date
<date-fun>	ex:valueParam1	{Birthdate}
<date-fun>	ex:valueParam2	yyyyMMdd
<date-fun>	ex:valueParam3	d/M/y

Predicate_Object sheet

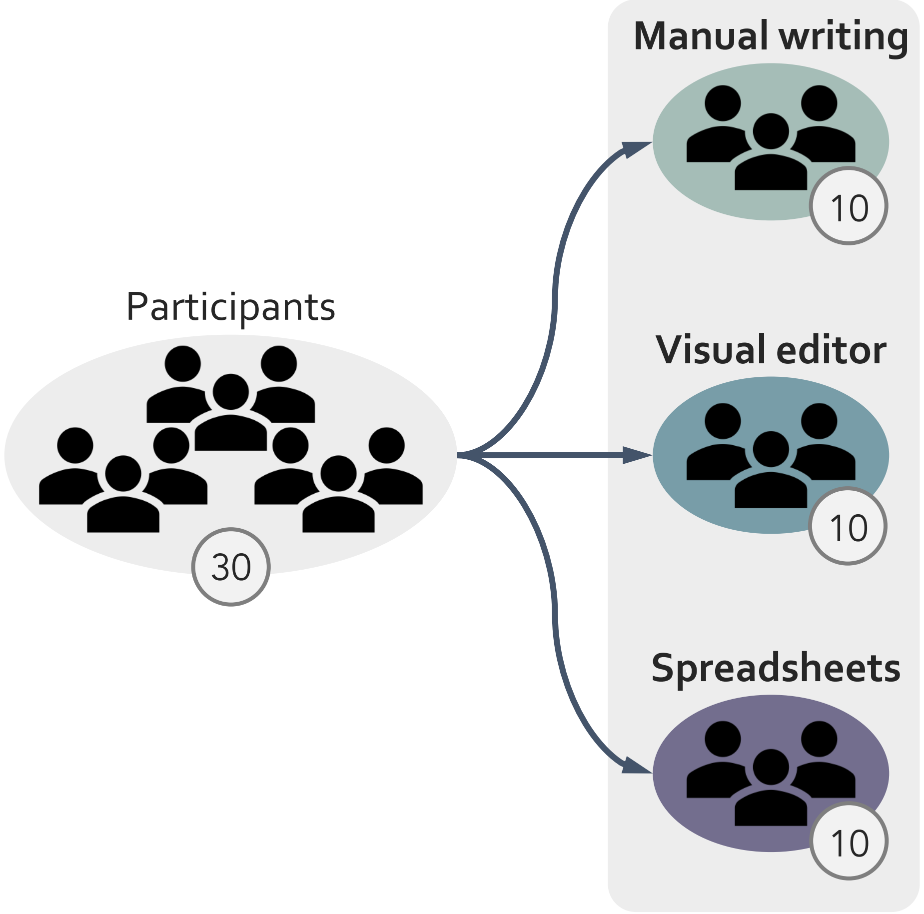
ID	Predicate	Object	DataType	Language	ReferenceID	InnerRef	OuterRef
SPORT	ex:name	{Sport}	string	en			
PERSON	ex:name	{Name}	string				
PERSON	ex:plays				SPORT	{SportID}	{ID}
PERSON	ex:birthdate	<date-fun>					

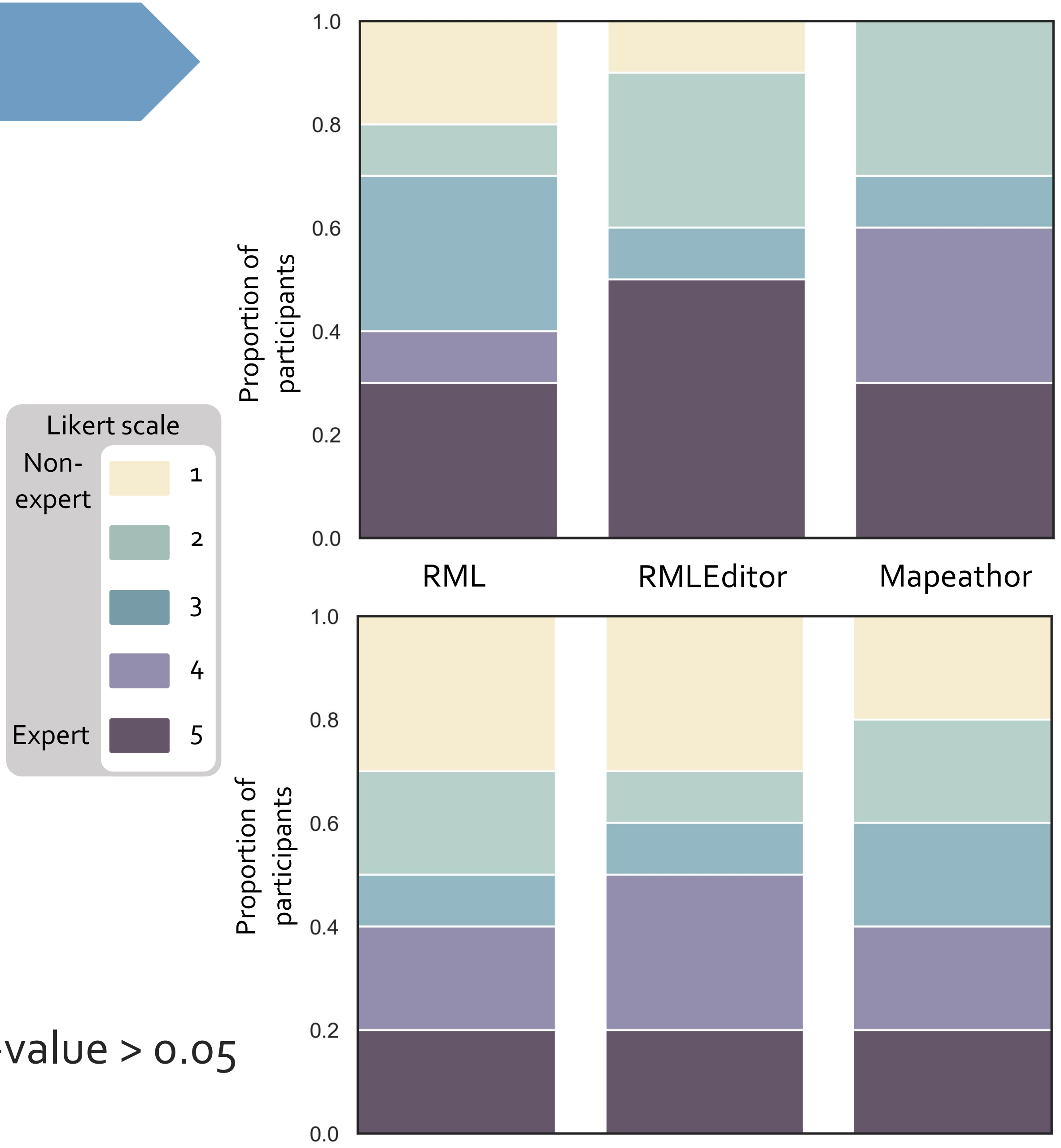
Mapeathor

- **Spreadsheet parser** that writes mapping files
  - Processes **Excel** and **Google Spreadsheets**
  - Produces mappings in **R2RML**, **RML**, **YARRRML**
- Available as PyPi package, web service and with Docker

https://doi.org/10.5281/zenodo.5973906 https://github.com/oeg-upm/mapeathor

Participant sign-up → Group Assignment



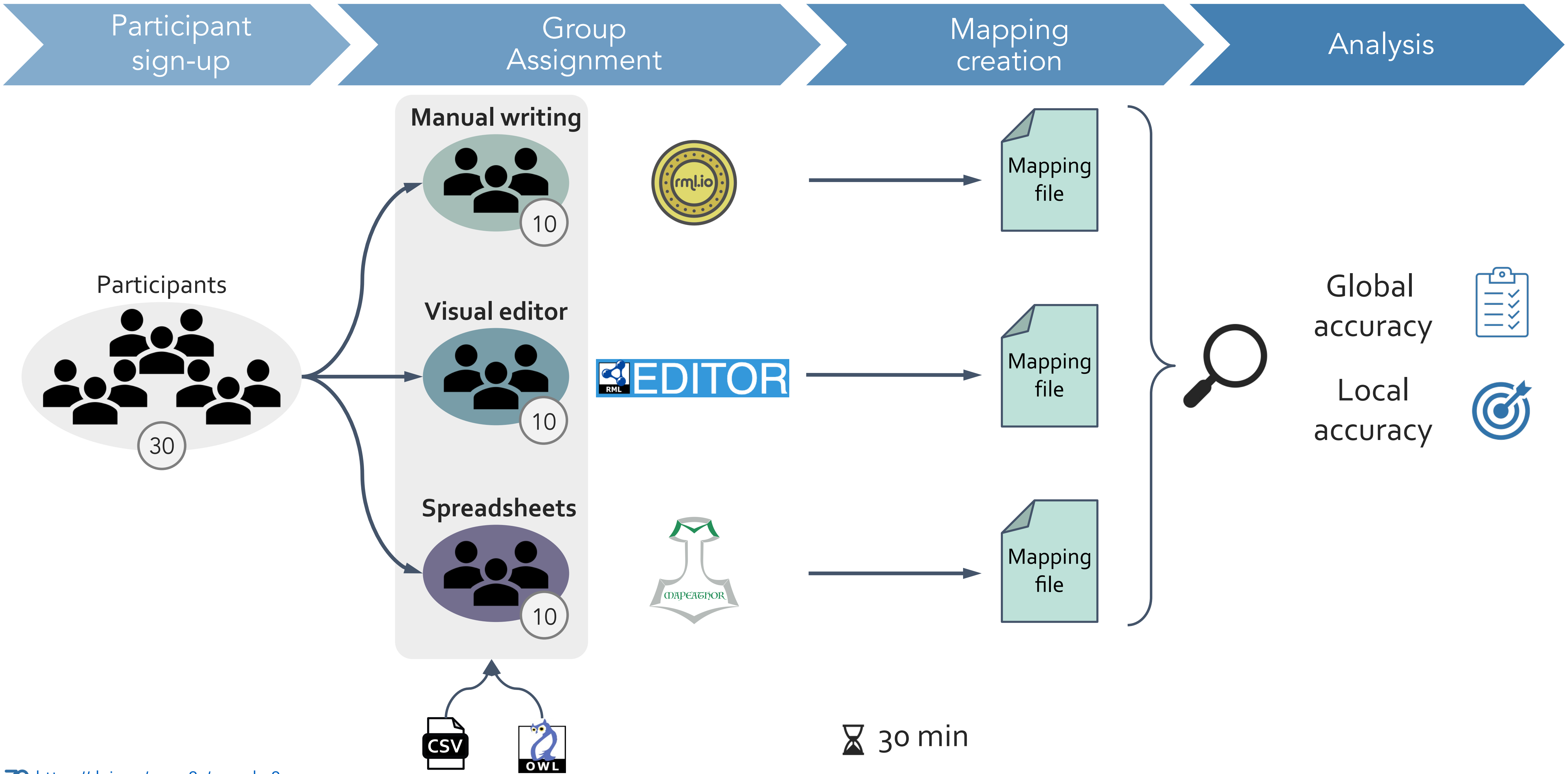


Expertise in Linked Data

Expertise in mappings

Chi-square test for homogeneity → p-value > 0.05

https://doi.org/10.5281/zenodo.8154522

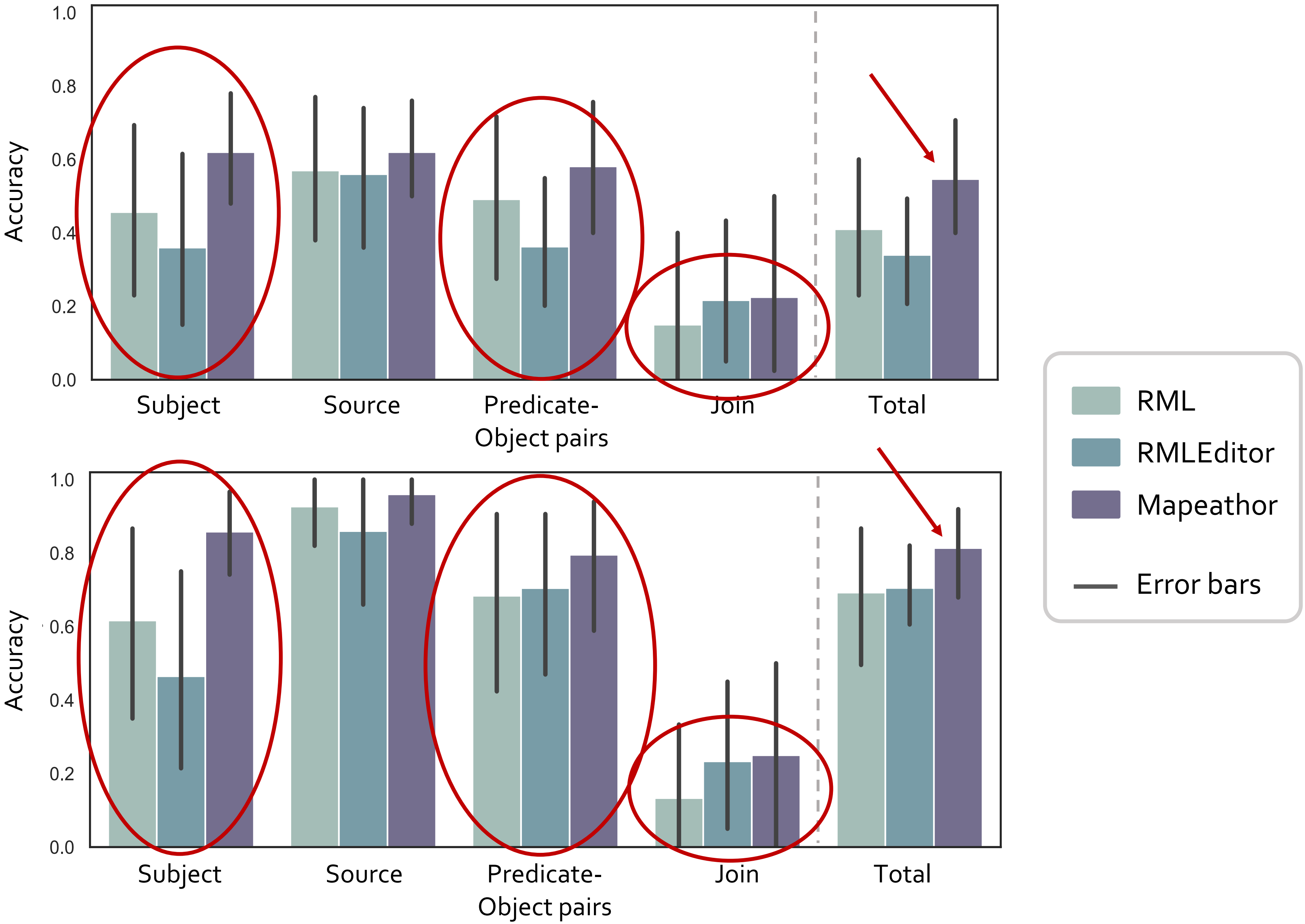


https://doi.org/10.5281/zenodo.8154522

Results

Global accuracy

Local accuracy

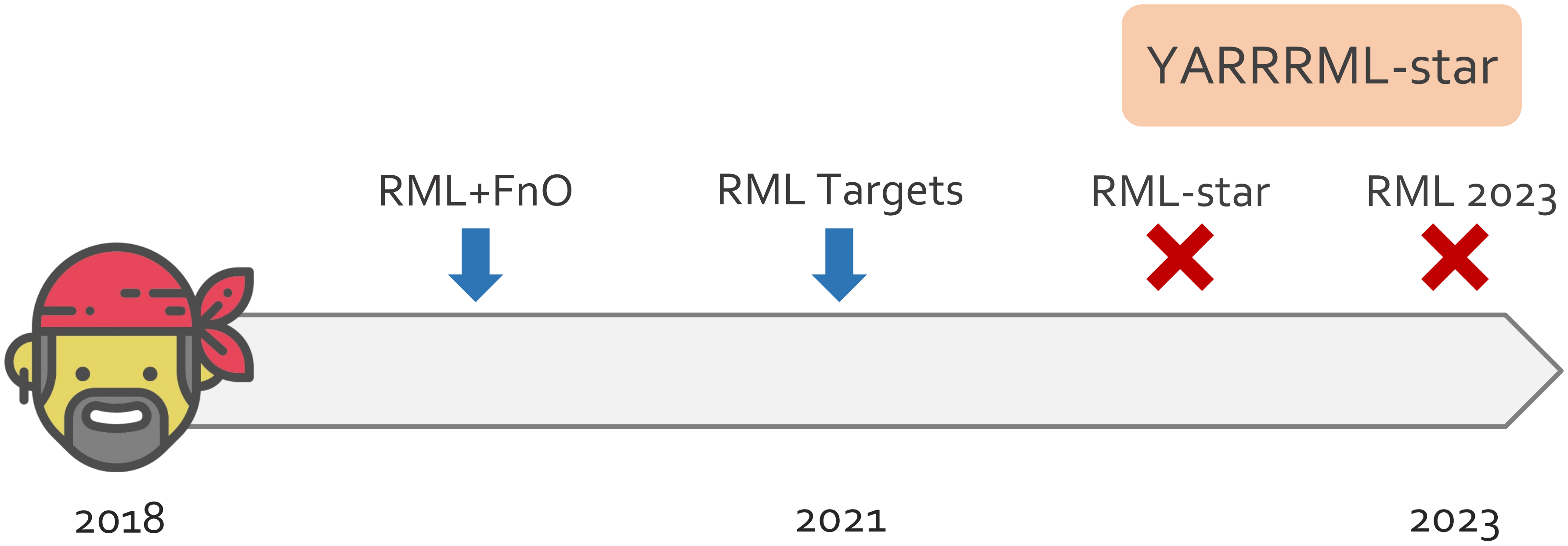


C5

Update of a user-friendly syntax for mapping languages with extended features

Iglesias-Molina, A., Chaves-Fraga, D., Dasoulas, I. and Dimou, A. (2023) Human-Friendly RDF Graph Construction: Which One Do You Chose?. *In Proceedings of the 23rd International Conference on Web Engineering 2023 (ICWE2023), June 6–9, 2023 Alicante, Spain.*

- User-friendly **serialization** for **RML**
- **Evolved with RML extensions** over the years



https://oeg-dataintegration.github.io/yarrrml-spec/

YARRRML-star

```
<#rankTM>
  a rml:NonAssertedTriplesMap ;
  rml:logicalSource :ranks ;
  rml:subjectMap [
    rml:template ":athlete/{ID}"];
  rml:predicateObjectMap [
    rml:predicate ex:rank ;
    rml:objectMap [
      rml:reference “RANK” ] ] .

<#competitionTM>
  a rml:TriplesMap ;
  rml:logicalSource :ranks ;
  rml:subjectMap [
    rml:quotedTriplesMap <#rankTM>];
  rml:predicateObjectMap [
    rml:predicate ex:competition ;
    rml:objectMap [
      rml:reference “COMP” ] ] .
```

```
mappings:
  rankTM:
    source:
      - [...]
    subject: ":athlete/{ID}”
    po:
       - [ ex:rank, $(RANK) ]

  competitionTM:
    source:
      - [...]
    subject:
      quotedNonAsserted: rankTM
    po:
       - [ ex:competition, $(COMP) ]
```

https://oeg-dataintegration.github.io/yarrrml-spec/

Additional extensions:

- Dynamic datatypes
- Dynamic language tags
- Inline joins (reducing verbosity)

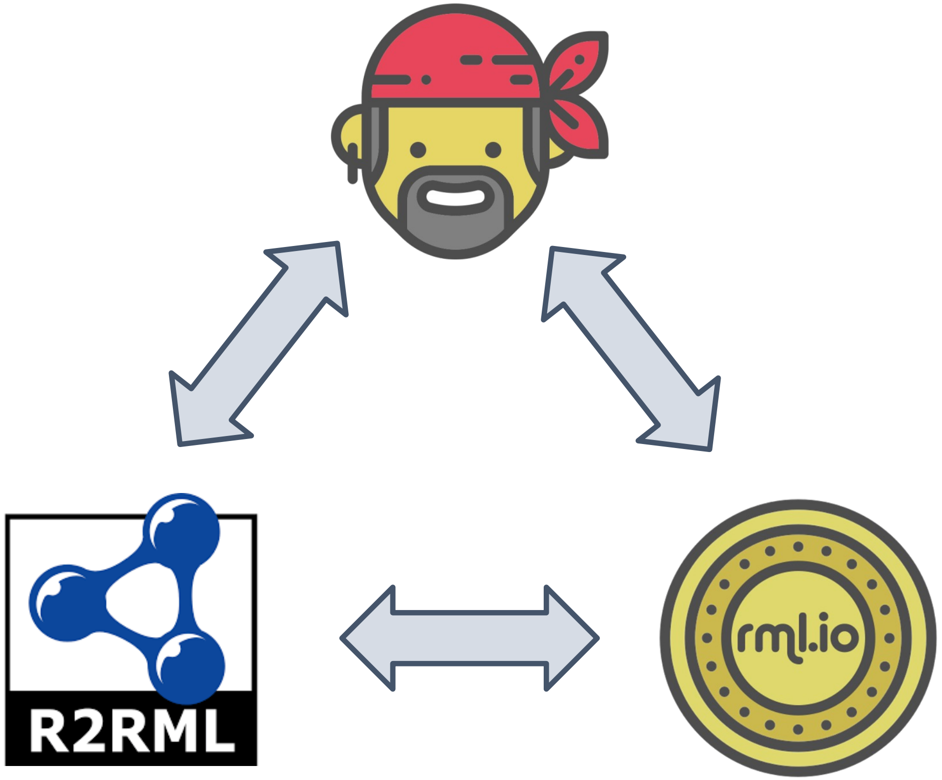
```
mappings:
 rankTM:
  source:
   - [...]
  subject: ":athlete/{ID}"
  po:
   - [ ex:rank, $(RANK) ]
   - [ ex:birthdate, $(DATE), xsd:$(DATATYPE) ]
   - [ ex:name, $(NAME), $(COUNTRY)~lang ]
   - predicates: ex:competesIn
     objects:
      - function: join( mapping=competitionTM,
                        equal(child=$(ID), parent=$(ID) )

 competitionTM:
  source:
   - [...]
  subject:
   quotedNonAsserted: rankTM
  po:
   - [ ex:competition, $(COMP) ]
```

https://oeg-dataintegration.github.io/yarrrml-spec/

Yatter

- Bi-directional **mapping translator**
  - R2RML, RML YARRRML
- Available as PyPi package
- Compliance with **modules**:
  - RML-star
  - RML-IO
  - RML-FNML
  - RML-core (partially)



https://doi.org/10.5281/zenodo.7024500 https://github.com/oeg-upm/yatter/

Test cases

- **> 50 test cases**
- Extended from [R2]RML test cases
- **Coverage**:
  - R2RML, RML-core, RML-FNML, RML-star and RML-IO
- Considers **shortcuts**
- **Available for any YARRRML engine**

Serializations comparison

- **Comparison** of YARRRML-star with other **user friendly serializations**:
  - ShExML, SMS2, XRM
- Assessment of **15 features**

https://github.com/oeg-upm/yarrrml-validation

Conclusions for Contributions of O2

Steep learning curve for writing mappings for KG construction

Spreadsheet-based approach for writing mapping rules in different languages

Novel yet familiar approach for **easing the mapping writing** for users with different backgrounds

The **YARRRML-star** extension

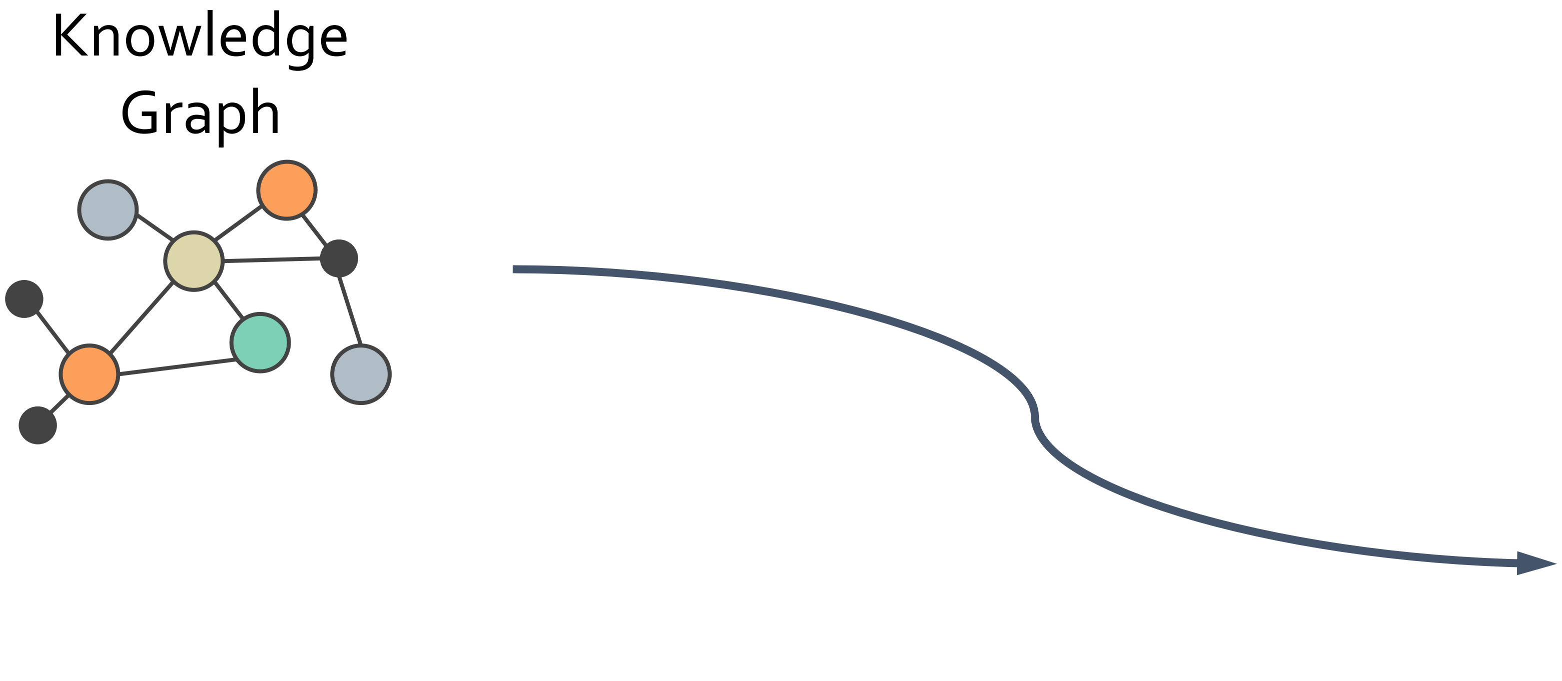
Empower a well-known serialization with the latest updates from RML

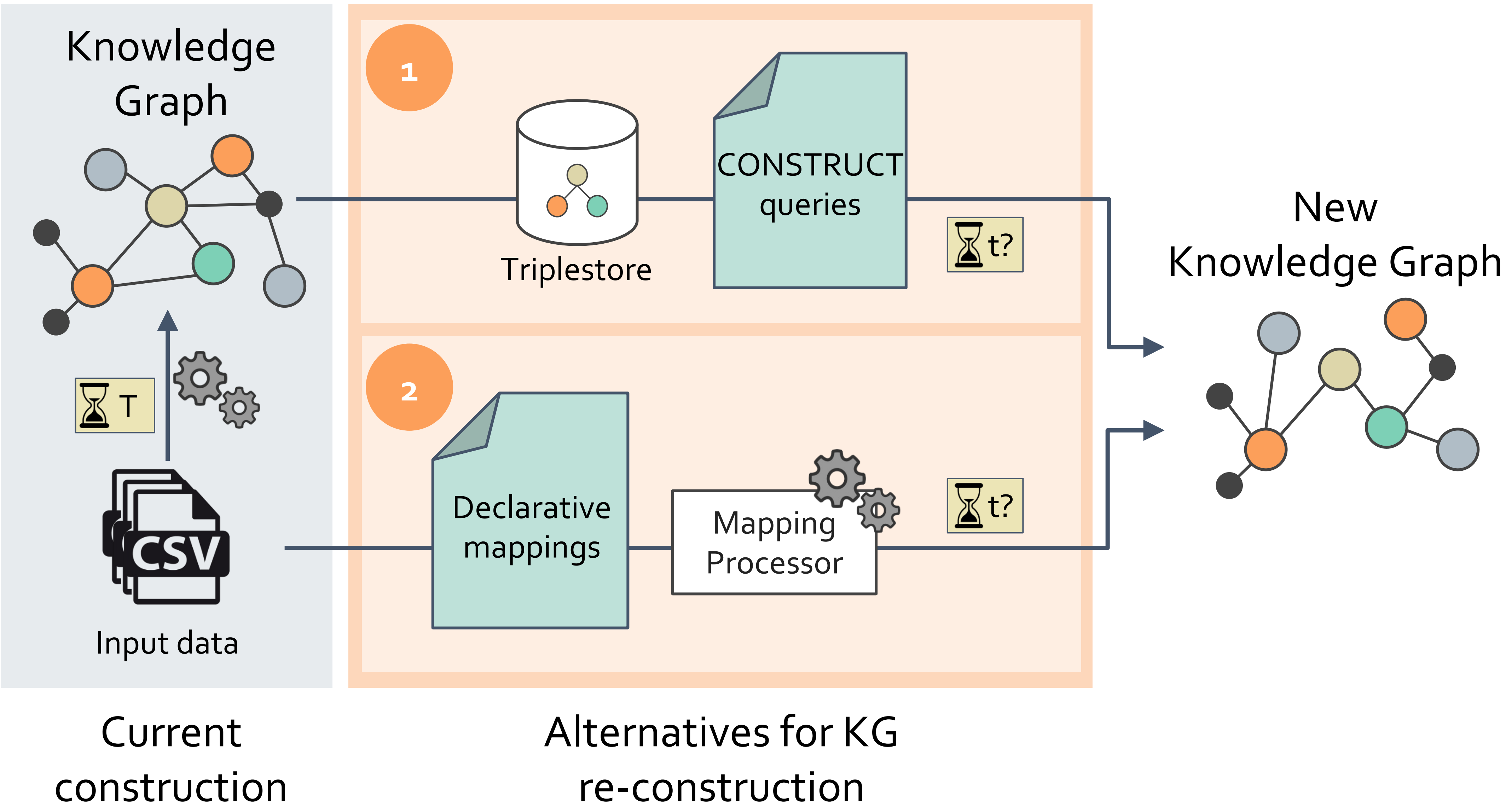
- Introduction
- State of the Art
- Thesis Objectives
- **Contributions**
  - O1. Understanding, representing and extending the expressiveness of declarative mapping languages
  - O2. Enhancing the creation of mapping rules
  - **O3. Representation impact on knowledge graph refactoring**
- Conclusions

C6

Analysis of scenarios where declarative KG construction technologies support KG evolution

Iglesias-Molina A., Toledo J., Corcho O. and Chaves-Fraga D. (2023) Re-Construction Impact on Metadata Representation Models. *In Proceedings of The Twelfth International Conference on Knowledge Capture (K-CAP23)*, December 5–7, 2023, Pensacola, US.


Knowledge
Graph
New
Knowledge Graph

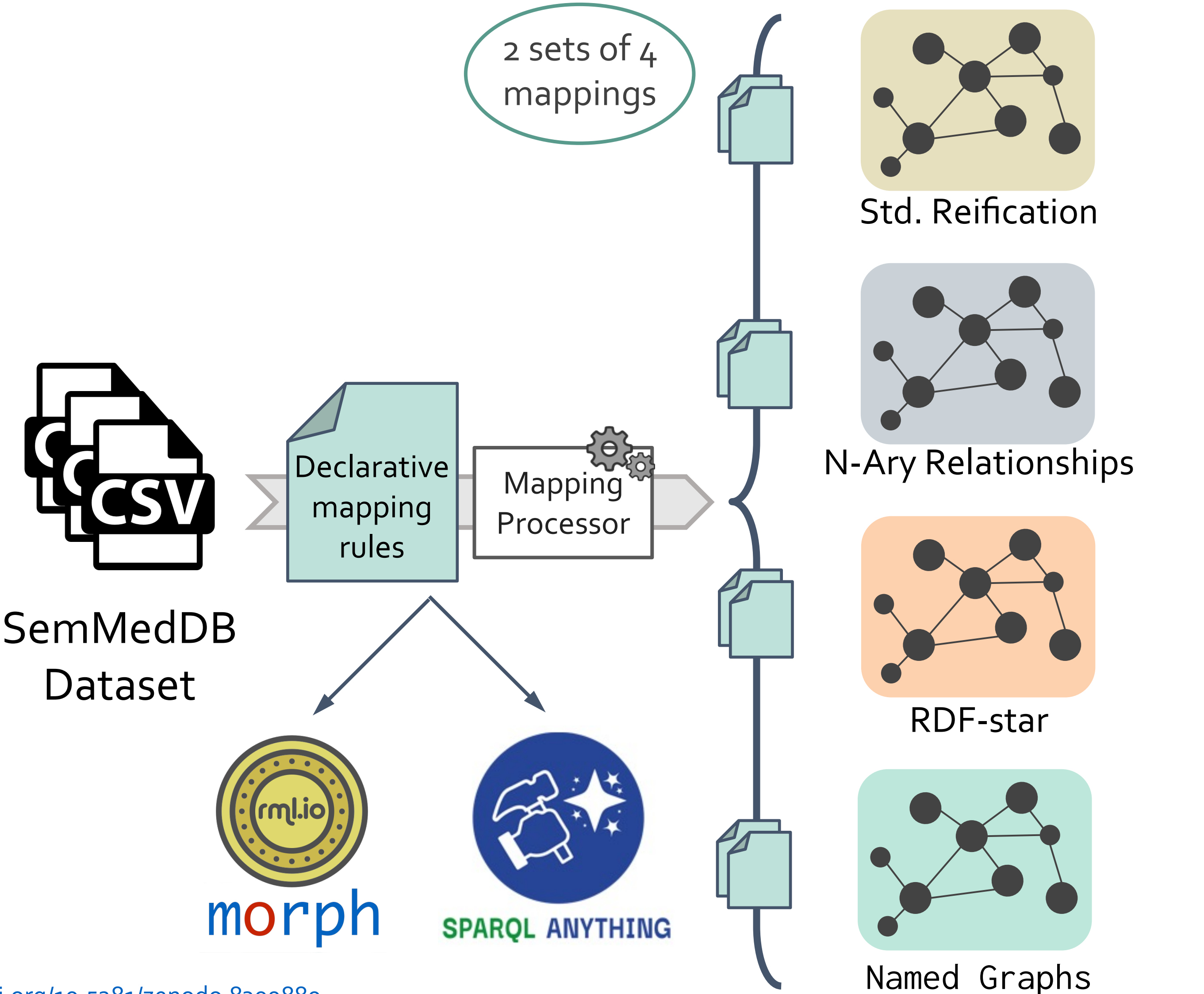

Knowledge Graph
T
Input data
1
Triplestore
CONSTRUCT queries
t?
2
Declarative mappings
Mapping Processor
t?
New Knowledge Graph
Current construction
Alternatives for KG re-construction

- **Biomedical entities and relationships**
- Suitable for testing metadata representations:
  - Entities are assigned a *semantic type* with a *score* value
  - Extraction of entities is assigned a *timestamp*
- Available in CSV and RDB
- Used with **4 representations in 4 scales**

	1K	10K	100K	1M
Std. Reification	25 000	249 997	2 499 966	24 999 607
Named Graphs	10 000	99 994	999 932	9 999 190
N-Ary Relationships	15 000	149 997	1 499 966	14 999 595
RDF-star	8 485	78 655	710 588	6 503 388

Alternative 1: Re-construct from source data

Knowledge Graphs in different representations



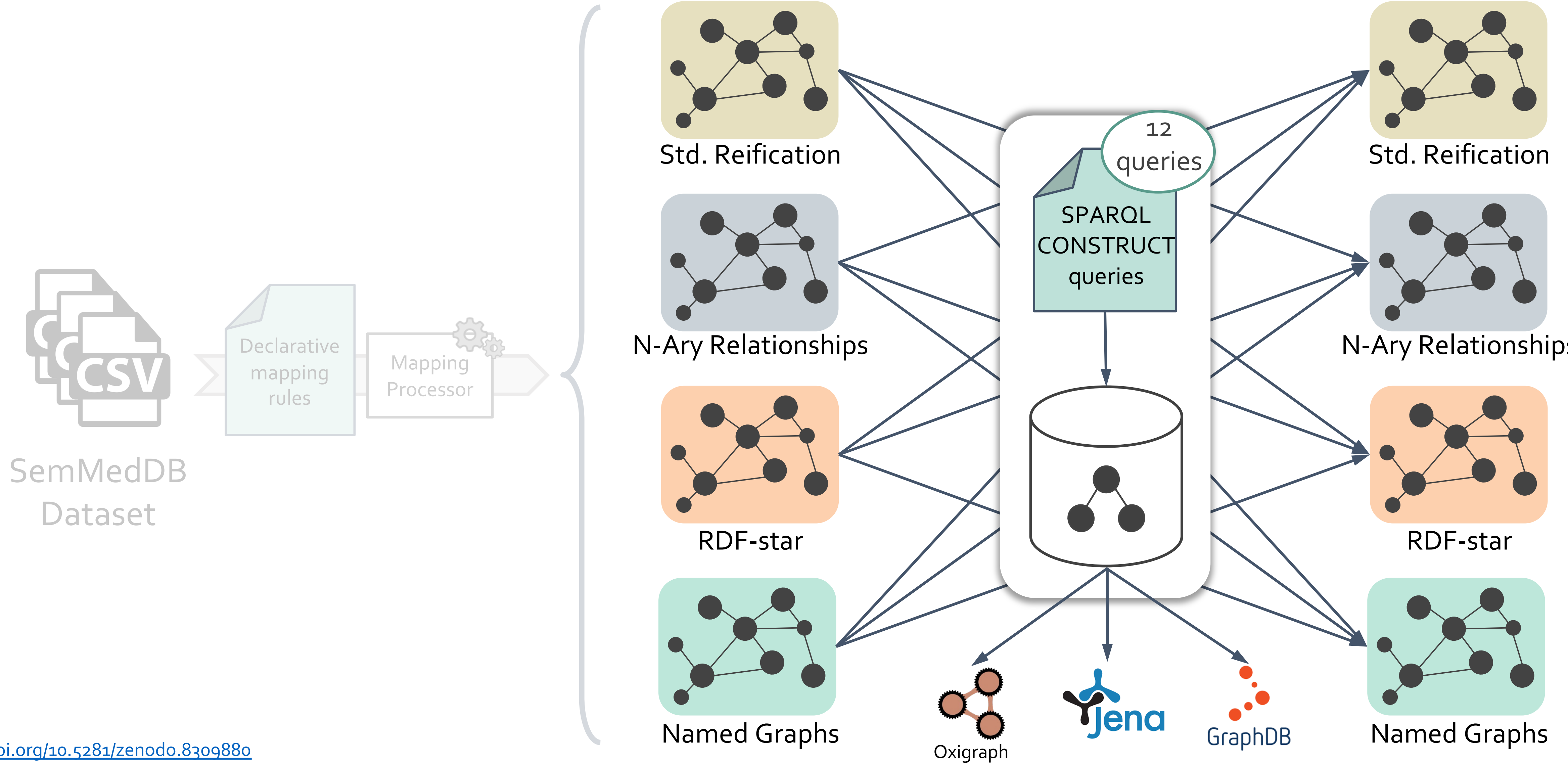
https://doi.org/10.5281/zenodo.8309880

Experimental setup

Alternative 1: Re-construct from source data

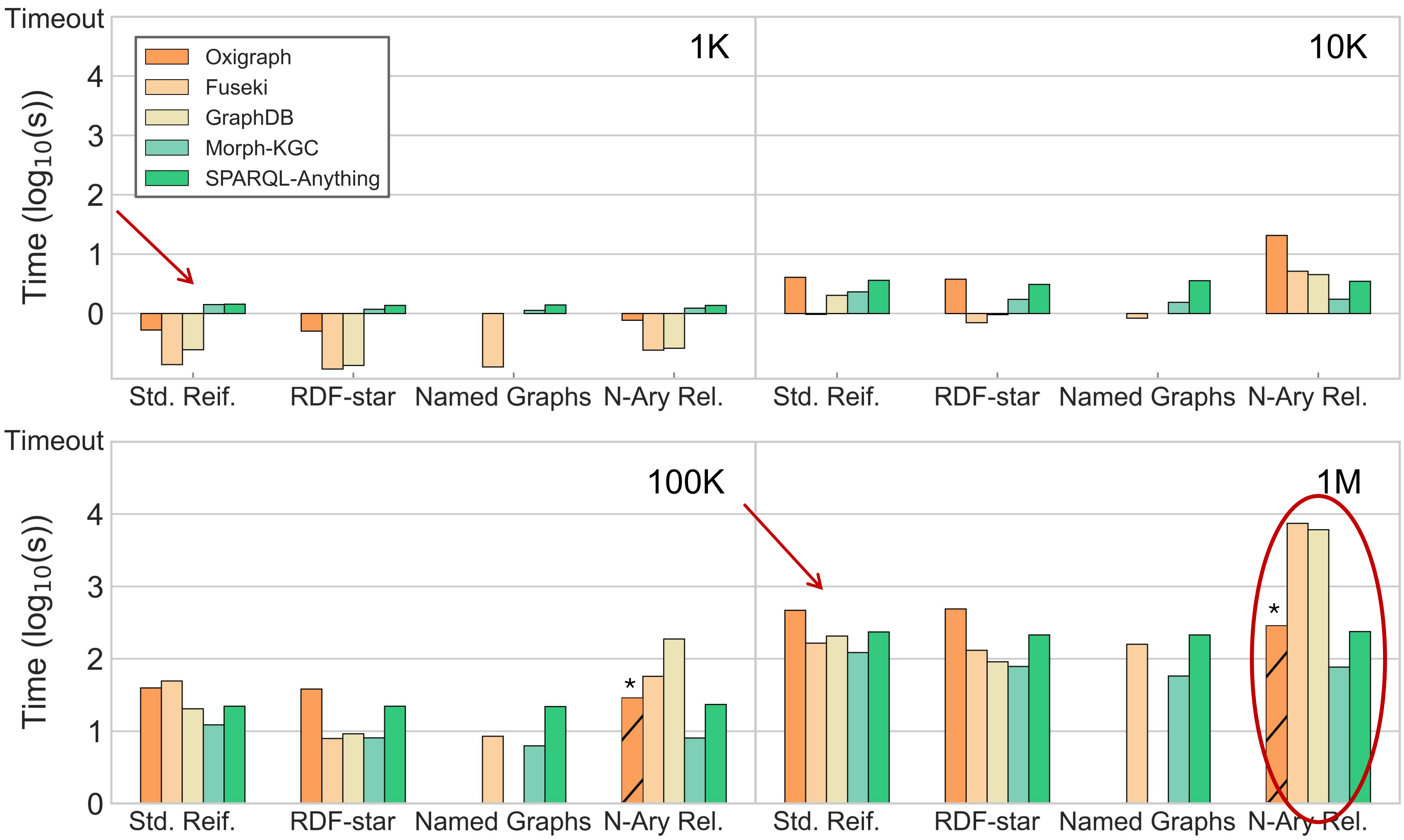
Knowledge Graphs in different representations

Alternative 2: Re-construct within triplestore



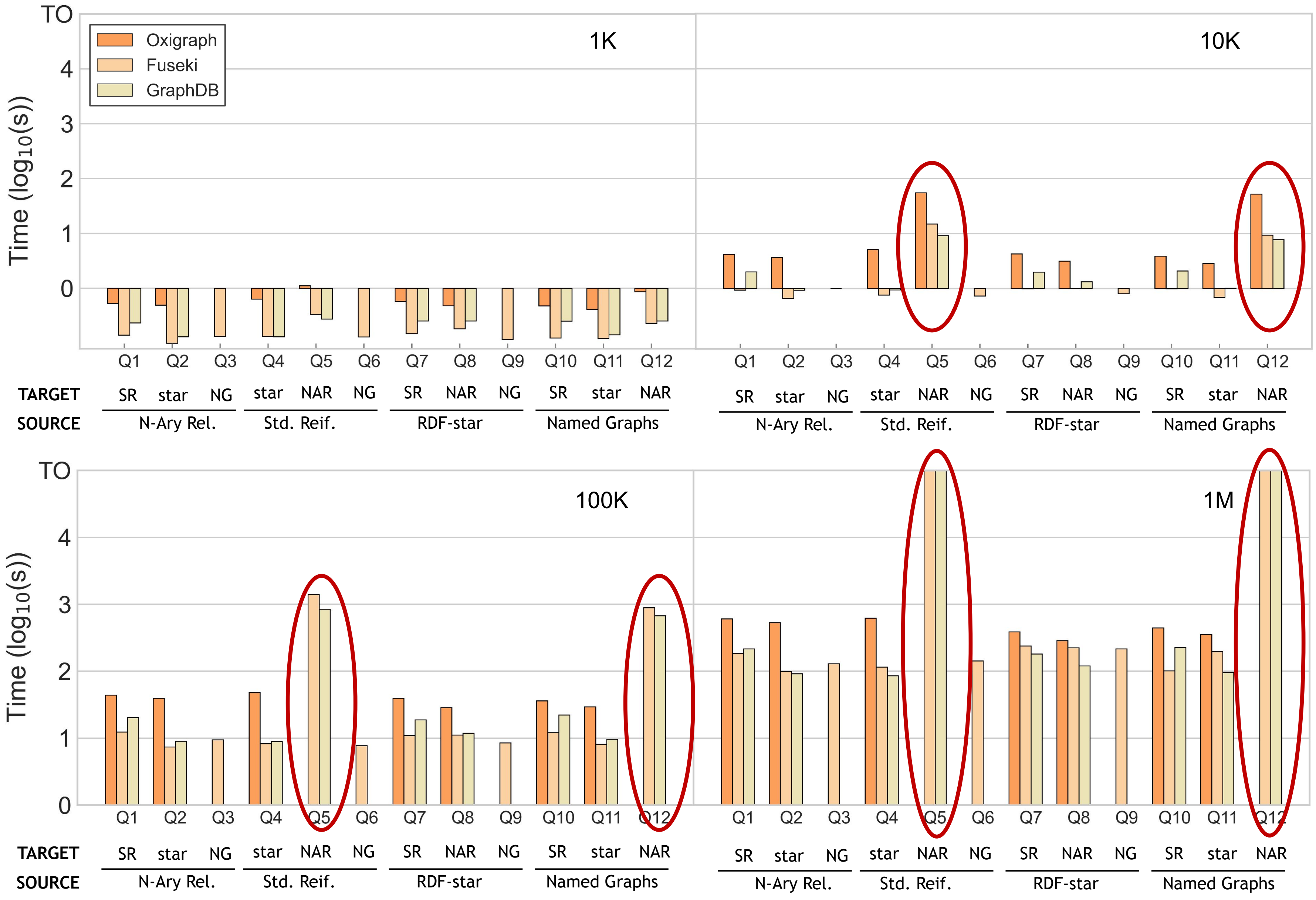
https://doi.org/10.5281/zenodo.8309880

Construction from scratch vs re-construction in triplestore



*Bars with diagonal patterns contain out-of-memory errors produced in some queries

Pairs of translations within triplestores



How can declarative KG construction workflows can **assist KG maintenance** in a re-factoring scenario?

Evaluation for re-constructing a KG with a different schema with **mappings** and **triplestores**

Identification where **mapping-based workflows enhance the performance** for KG re-construction

Analysis of the **influence of schemas** in triplestores and the **limitations** of the SPARQL CONSTRUCT clause

- Introduction
- State of the Art
- Thesis Objectives
- Contributions
- **Conclusions**
  - Summary of contributions and conclusions
  - Future steps

Conclusions

O1: To **analyze the needs** for KG construction from heterogeneous data sources to **support advances** in mapping languages

- C1 — **Understand** in detail the mapping languages **characteristics** and **limitations**
- C2 — Gathered the characteristics of **what mapping languages are desired to express**, implement and publish them
- C3 — **Empower** a widely used **mapping language for constructing RDF-star** graphs for easing statement annotation

O2: To **help** knowledge engineers and domain experts with building mappings in a **user-friendly manner**

- C4, C5 — Assist practitioners with different backgrounds to **facilitate their access and writing** of mapping files
- C4, C5 — Study and analyze the main **difficulties in mapping writing** and how they can be **overcome** in **future** developments

O3: To **assess** declarative KG construction technologies in terms of their benefits for **KG evolution**

- C6 — KGC workflows using mappings can **facilitate** long-term **maintenance** of knowledge graphs
- C6 — Highlight the **limitations** of long-established technologies for these needs

Future lines of research

User needs in the center: **evolve** the languages **as new requirements appear** with surrounding technology and challenges

Identify the factors involved in long-term **KG maintenance** to keep up, develop mechanisms for **propagating updates** and **governance guidelines**

Assess how **LLMs can assist KG construction** while maintaining the workflows **reproducible**, **maintainable** and **ensuring KG quality**

Publications

Journal | Conference | Workshop | Poster/Demo

J O1 — Arenas-Guerrero, J.*, **Iglesias-Molina, A.***, Chaves-Fraga, D., Garijo, D., Corcho, O., & Dimou, A. (2024). Declarative generation of RDF-star graphs from heterogeneous data. *Semantic Web*, in press (Q2).

J O1 — **Iglesias-Molina, A.**, Cimmino, A., Ruckhaus, E., Chaves-Fraga, D., García-Castro, R., & Corcho, O. (2024). An ontological approach for representing declarative mapping languages. *Semantic Web*, *15* (1), 191–221 (Q2).

C O3 — **Iglesias-Molina A.**, Toledo J., Corcho O. and Chaves-Fraga D. (2023) Re-Construction Impact on Metadata Representation Models. *K-CAP 2023* (Core B).

C O1 — **Iglesias-Molina, A.**, Van Assche, D., Arenas-Guerrero, J., De Meester, B., Debruyne, C., Jozashoori, S., Maria, P., Michel, F., Chaves-Fraga, D. and Dimou, A. (2023) The RML Ontology: A Community-Driven Modular Redesign After a Decade of Experience in Mapping Heterogeneous Data to RDF. *ISWC 2023* (Core A).

C O3 — **Iglesias-Molina, A.**, Ahrabian, K., Ilievski, F., Pujara, J. and Corcho, O. (2023) Comparison of Knowledge Graph Representations for Consumer Scenarios. *ISWC 2023* (Core A).

C O2 — **Iglesias-Molina, A.**, Chaves-Fraga, D., Dasoulas, I. and Dimou, A. (2023) HumanFriendly RDF Graph Construction: Which One Do You Chose?. ICWE 2023 (Core B).

W O2 — **Iglesias-Molina, A.**, Chaves-Fraga, D., Priyatna, F. and Corcho, O. (2019) Towards the Definition of a Language-independent Mapping Template for Knowledge Graph Creation. *Workshop on Capturing Scientific Knowledge co-located with K-CAP 2019.*

W O1 — **Iglesias-Molina, A.**, Cimmino, A., Corcho, O. (2022) Devising Mapping Interoperability with Mapping Translation. *Workshop on Knowledge Graph Construction, co-located with ESWC 2022.*

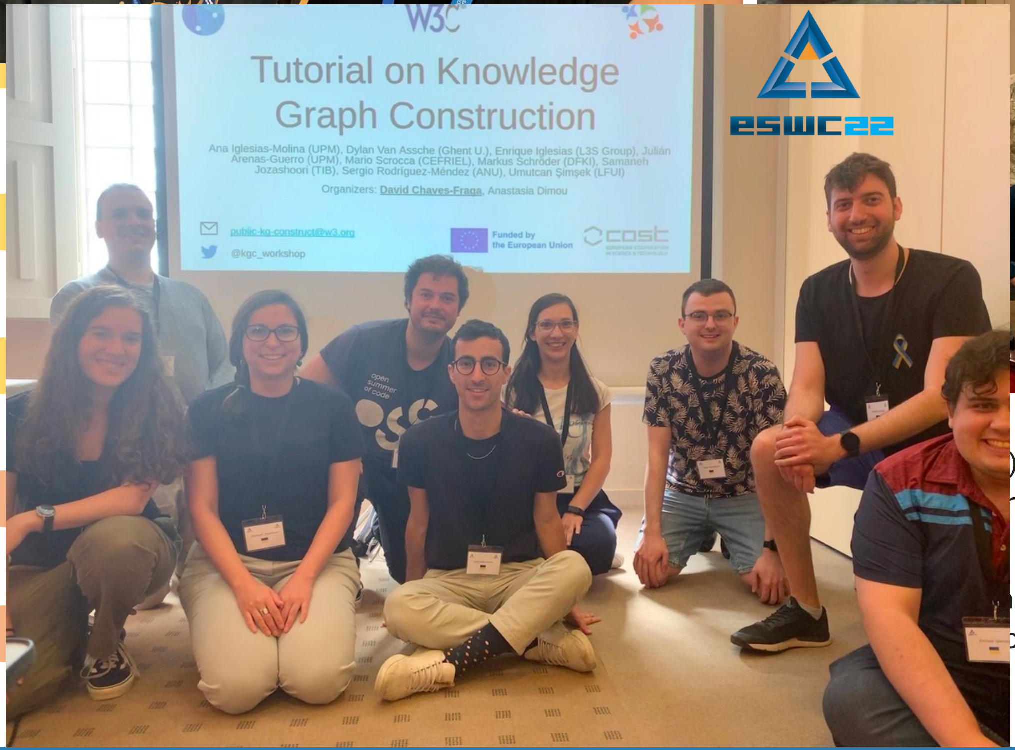
P O1 — **Iglesias-Molina, A**. and Garijo D. (2023) Towards Assessing FAIR Research Software Best Practices in an Organization Using RDF-star. *Semantics 2023 Posters and Demos Track.*

P O1 — Delva, T., Arenas-Guerrero, J., **Iglesias-Molina, A.**, Corcho, O., Chaves-Fraga, D., and Dimou, A. (2021) RML-star: A declarative mapping language for RDF-star generation. *ISWC 2021 Posters, Demos and Industry Tracks.*

D O2 — **Iglesias-Molina, A.**, Pozo-Gilo, L., Doña, D., Ruckhaus, E., Chaves-Fraga, D., & Corcho, O. (2020). Mapeathor: Simplifying the Specification of Declarative Rules for Knowledge Graph Construction. *ISWC 2020 Demos and Industry Tracks*.

Related Publications and Activities

os, y construcción de grafo de



ruction.

s-

bitation

ECAI 2024: How do ...
Anastasia Dimou an

- K-CAP 2023: Tutoria
  Graphs. Ana Iglesia
- ESWC 2022: Knowle
  Anastasia Dimou, E
  Dylan Van Assche, U
- ISWC 2020: Tutorial
  Rules. David Chaves