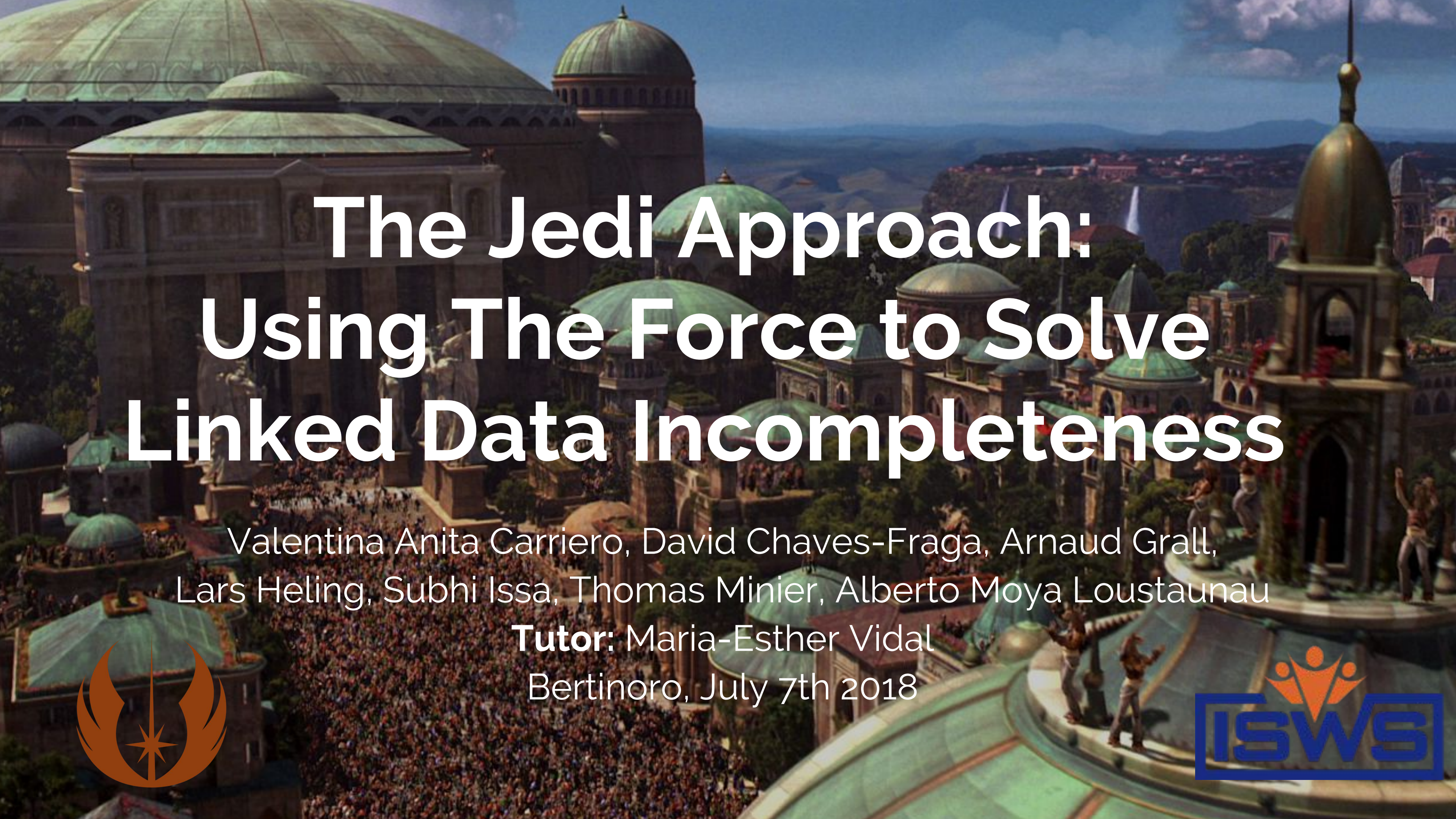


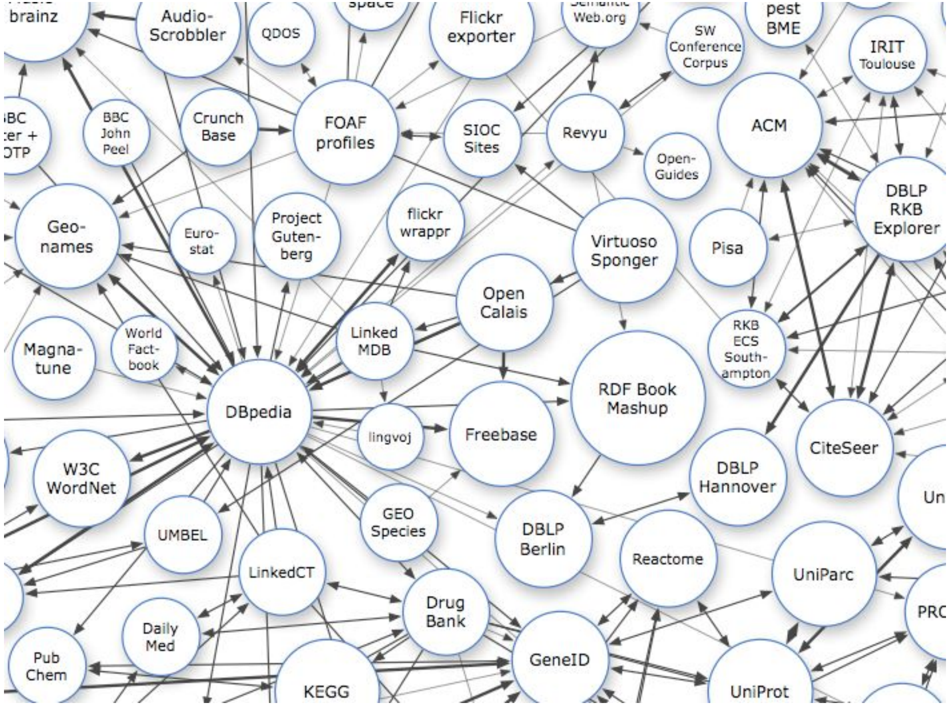
The Jedi Approach: Using The Force to Solve Linked Data Incompleteness

Valentina Anita Carriero, David Chaves-Fraga, Arnaud Grall,
Lars Heling, Subhi Issa, Thomas Minier, Alberto Moya Loustaunau
Tutor: Maria-Esther Vidal

Bertinoro, July 7th 2018

Introduction

- Following the **Linked Data principles** [1], data providers have made available hundreds of RDF datasets
- **Federated SPARQL queries** allow data consumers to evaluate SPARQL queries over several RDF datasets

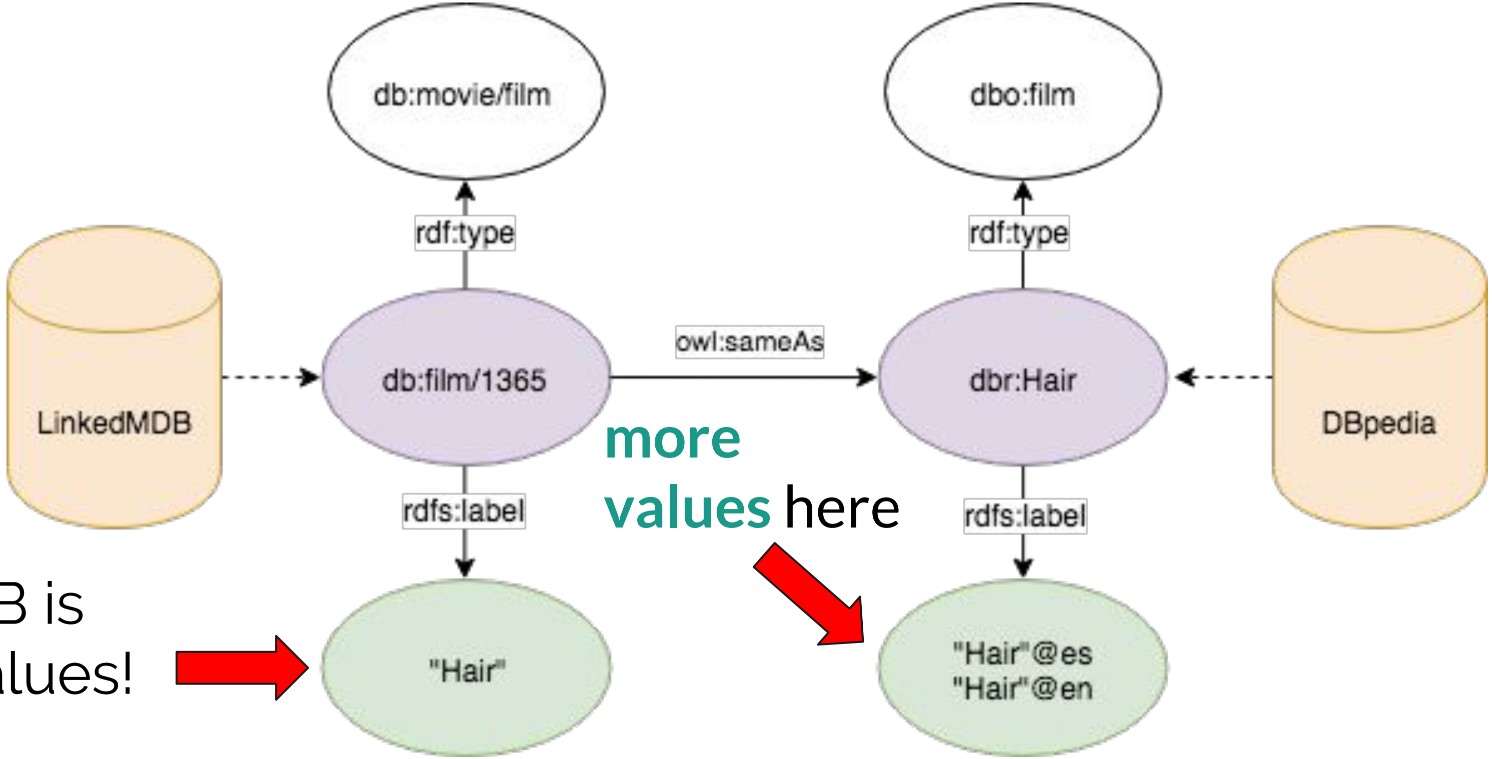


[1] Max Schmachtenberg, Christian Bizer, and Heiko Paulheim. Adoption of the Linked Data best practices in different topical domains. In ISWC, pages 245–260. 2014.

Linked Data Validity

- **Incompleteness** is one of the issues of **Linked Data validity**
- Many datasets have **missing values** for multiple RDF resources

Incompleteness in LinkedMDB & DBpedia

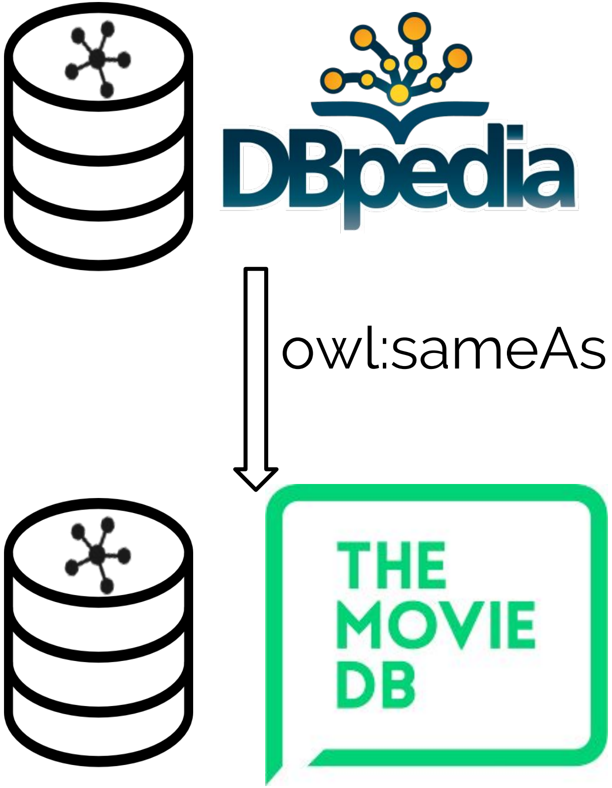


more values here

LinkedMDB is **missing** values!

Fixing incompleteness using Linked Data

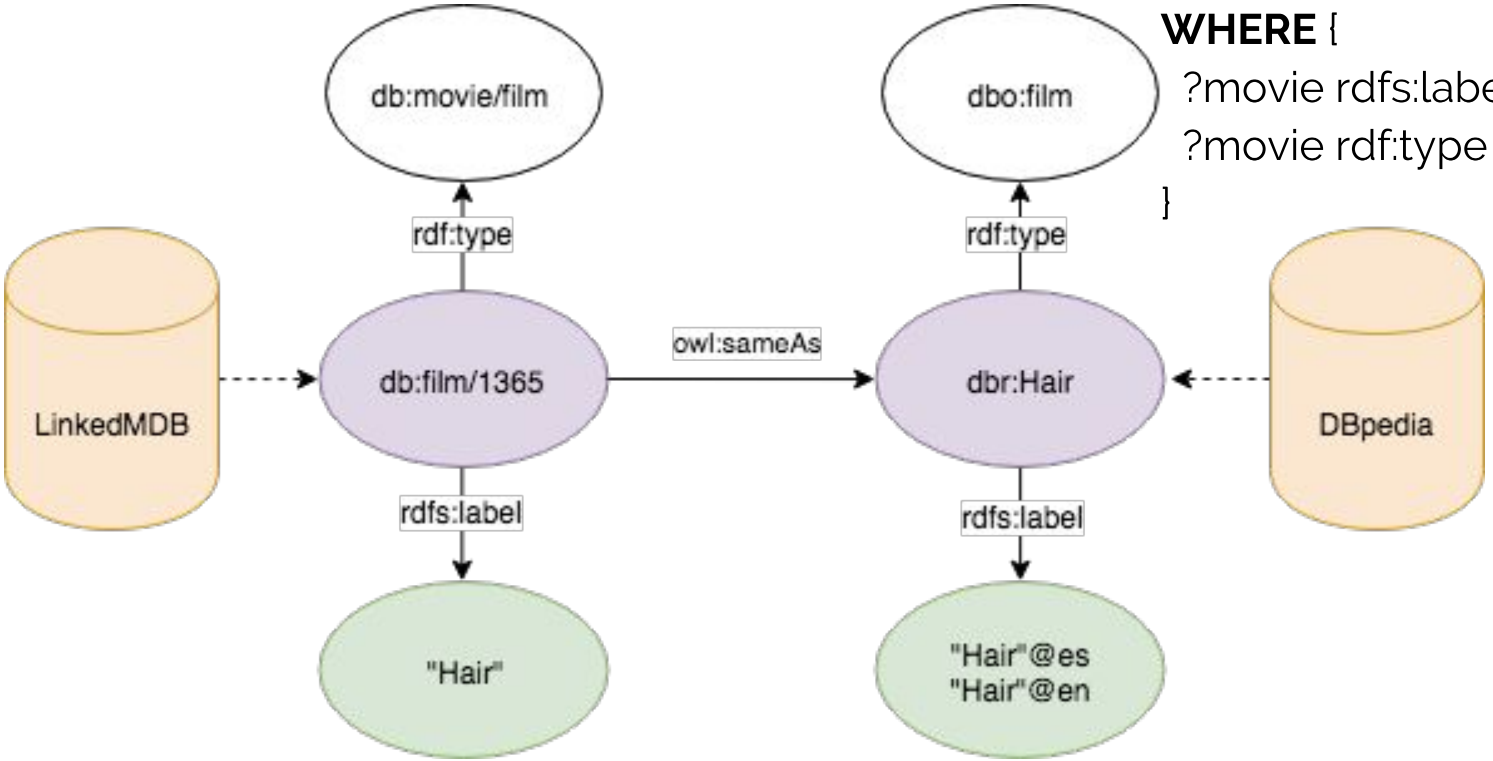
- In the LOD, entities are **linked** across remote RDF datasets
- These datasets can be used to **improve completeness**
- Let’s use them with a **federated SPARQL query**!



owl:sameAs

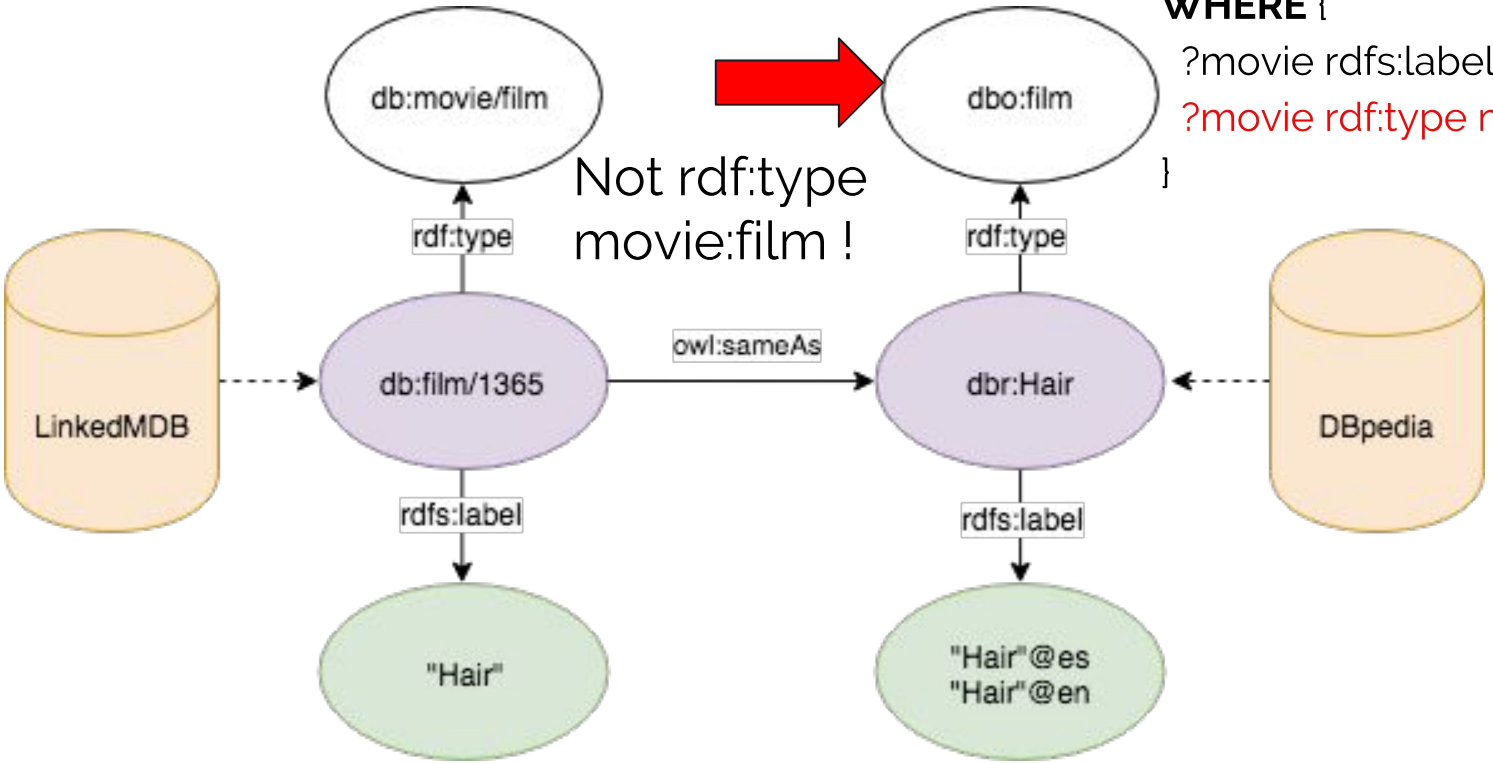
Motivating example

```
SELECT ?movie ?label
WHERE {
  ?movie rdfs:label ?label.
  ?movie rdf:type movie:film.
}
```



Semantic heterogeneity

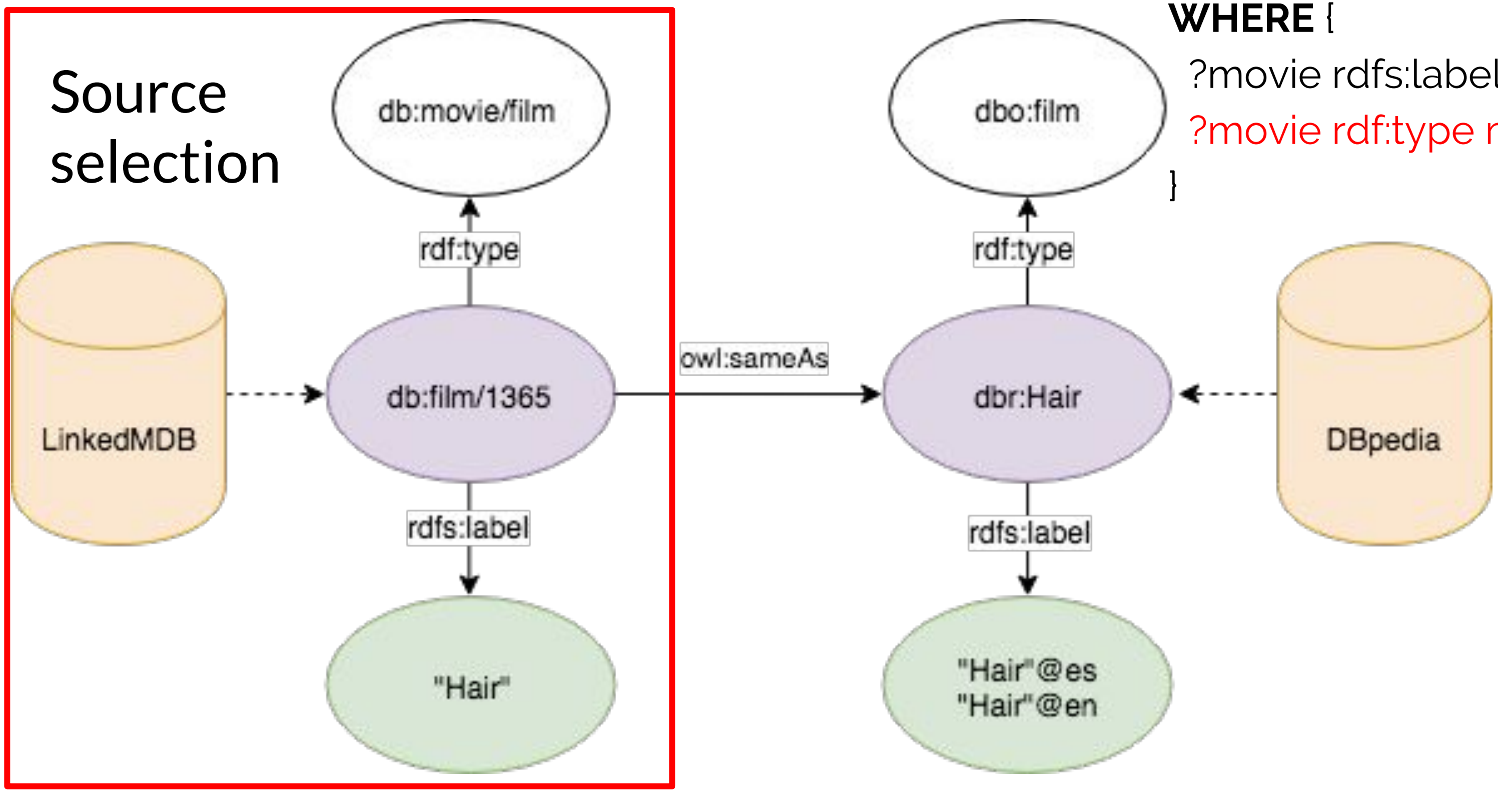
```
SELECT ?movie ?label
WHERE {
  ?movie rdfs:label ?label.
  ?movie rdf:type movie:film.
}
```



Not rdf:type movie:film !

There is still incompleteness!

```
SELECT ?movie ?label
WHERE {
  ?movie rdfs:label ?label.
  ?movie rdf:type movie:film.
}
```



Research problem

Find the **minimal set of sources** from a federation of SPARQL endpoints to use **during query execution** in order to **maximize answer completeness**.

Related Work

Detecting incompleteness in the LOD

- HARE [2], a hybrid SPARQL engine that uses a **model** to **estimate the completeness** of RDF dataset.
- Finds missing values via microtask crowdsourcing.
- However, it **cannot detect incompleteness in a federation.**

[2] Acosta, M., Simperl, E., Flöck, F., Vidal, M. E.: Enhancing answer completeness of SPARQL queries via crowdsourcing. Web Semantics: Science, Services and Agents on the World Wide Web, 45, 41-62.

Describing RDF datasets using RDF-MTs

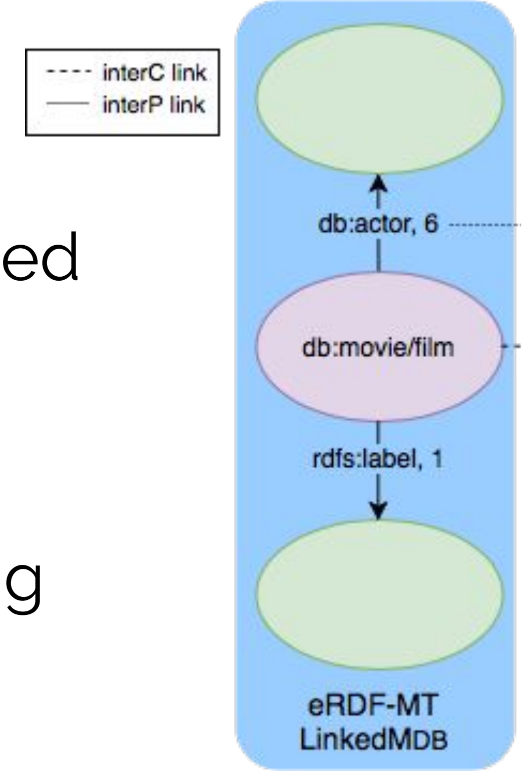
- MULDER [3] is a federated SPARQL query engine which describes RDF datasets using **RDF Molecules Templates**
- Properties are associated with entities of the same class
- **Links** between entities across datasets are included

[3] Endris, K. M., Galkin, M., Lytra, I., Mami, M. N., Vidal, M. E., Auer, S: MULDER: querying the linked data web by bridging RDF molecule templates. In International Conference on Database and Expert Systems Applications

The Jedi Approach

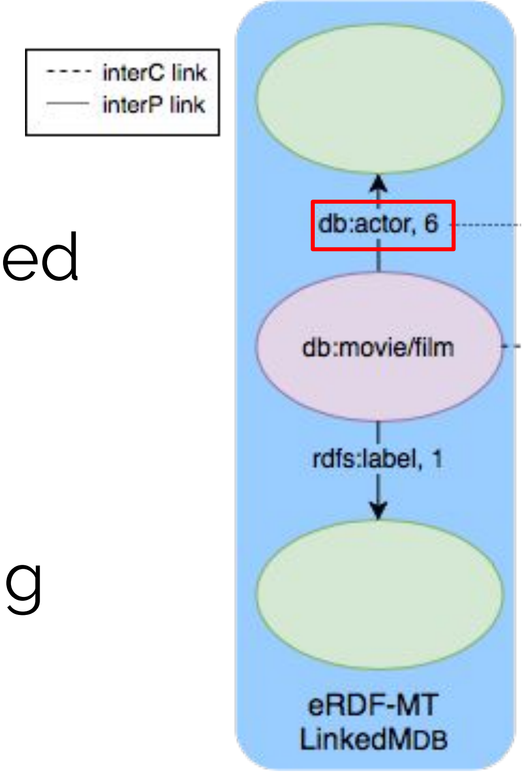
Extended RDF Molecule Templates (eRDF-MTs)

- Based on MULDER's RDF-MTs
- Properties are annotated with their **aggregated multiplicity**
- Allow to **detect incompleteness** during query execution



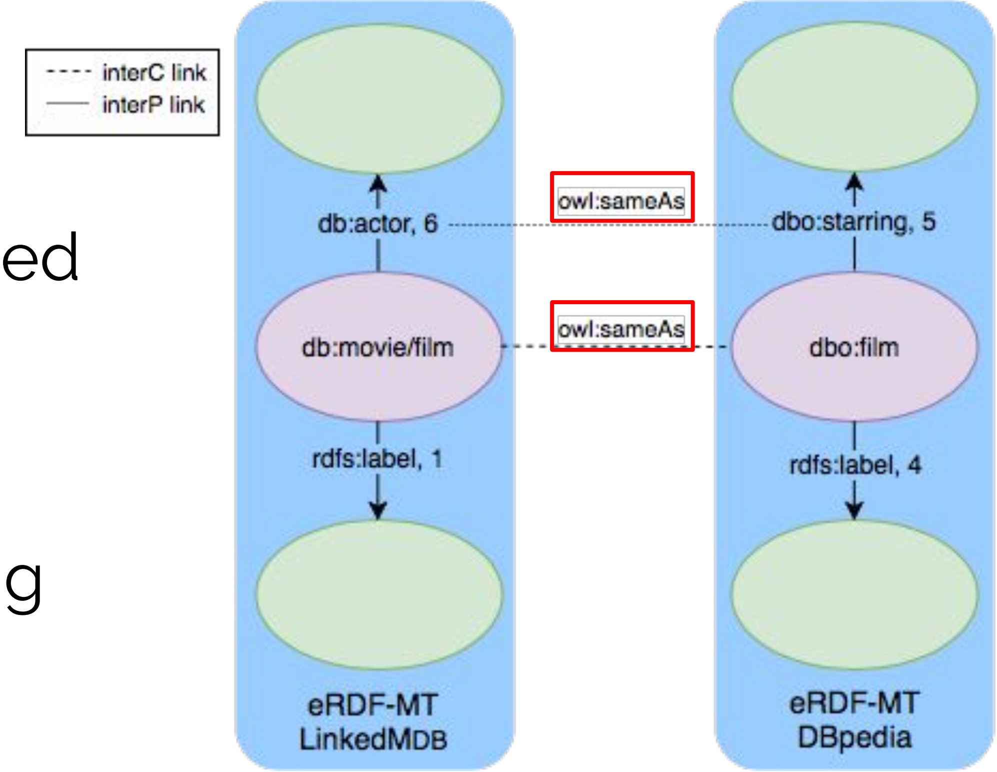
Extended RDF Molecule Templates (eRDF-MTs)

- Based on MULDER's RDF-MTs
- Properties are annotated with their **aggregated multiplicity**
- Allow to **detect incompleteness** during query execution



Extended RDF Molecule Templates (eRDF-MTs)

- Based on MULDER's RDF-MTs
- Properties are annotated with their **aggregated multiplicity**
- Allow to **detect incompleteness** during query execution

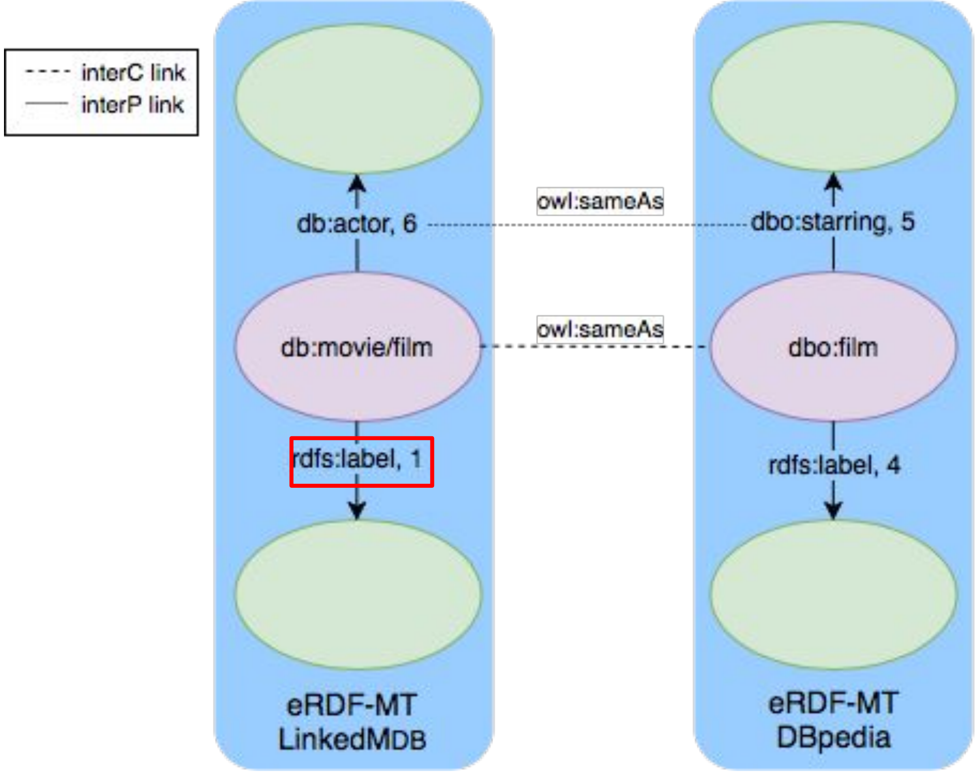


The Jedi cost-model & operator

- The Jedi **cost-model** is used to **select relevant RDF datasets** to improve answer completeness
- The **Jedi operator** allows to evaluate a triple pattern across a federation
  - Uses both eRDF-MTs & the cost-model

Jedi query rewriting

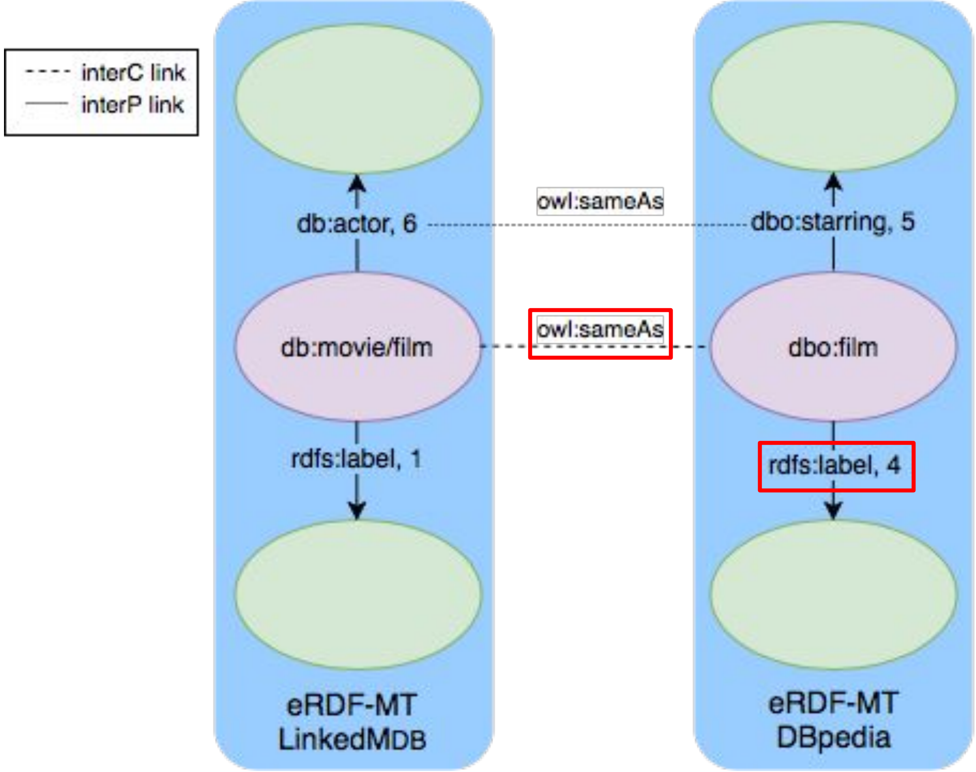
```
SELECT ?movie ?label
WHERE {
  ?movie rdfs:label ?label.
  ?movie rdf:type movie:film.
}
```



Jedi query rewriting

```
SELECT ?movie ?label
WHERE {
  ?movie rdfs:label ?label.
  ?movie rdf:type movie:film.
}
```

```
SELECT ?movie ?label
WHERE {
  ?movie rdf:type movie:film.
  ?movie owl:sameAs ?cc.
  ?cc rdfs:label ?label.
}
```



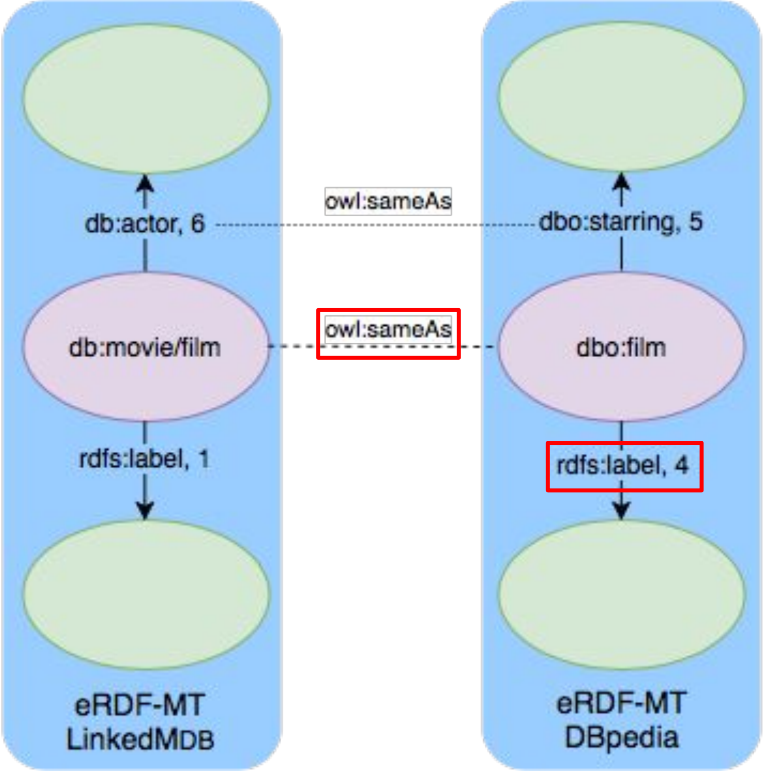
Jedi query rewriting

```
SELECT ?movie ?label
WHERE {
  ?movie rdfs:label ?label.
  ?movie rdf:type movie:film.
}
```

```
SELECT ?movie ?label
WHERE {
  ?movie rdf:type movie:film.
  ?movie owl:sameAs ?cc.
  ?cc rdfs:label ?label.
}
```

Rewritten the query has been

- - - - interC link
—— interP link



Preliminary Experimental Results

Domain	Query	DBpedia	DBpedia + Wikidata
Sport	q1	0	42
Movies	q2	3	6
Culture	q3	0	31
Drugs	q4	0	482
Life Sciences	q5	0	9

Conclusion

- Jedi is able improve answer completeness **using the Linked Data** during query execution
- Rely on **source description**, a **cost-model** and a **physical query operator**
- Can be easily integrated in any state-of-art federated SPARQL query engine

Future Works

- Try to compute eRDT-MTs client-side
  - **Less dependence** on the data providers
- Implement Jedi and perform a **complete experimental study**

Questions?

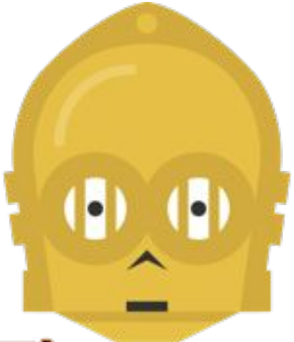
Lars

Maria-Esther

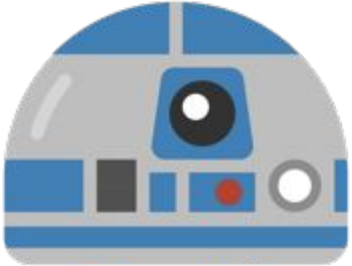
Arnaud

Alberto

Subhi



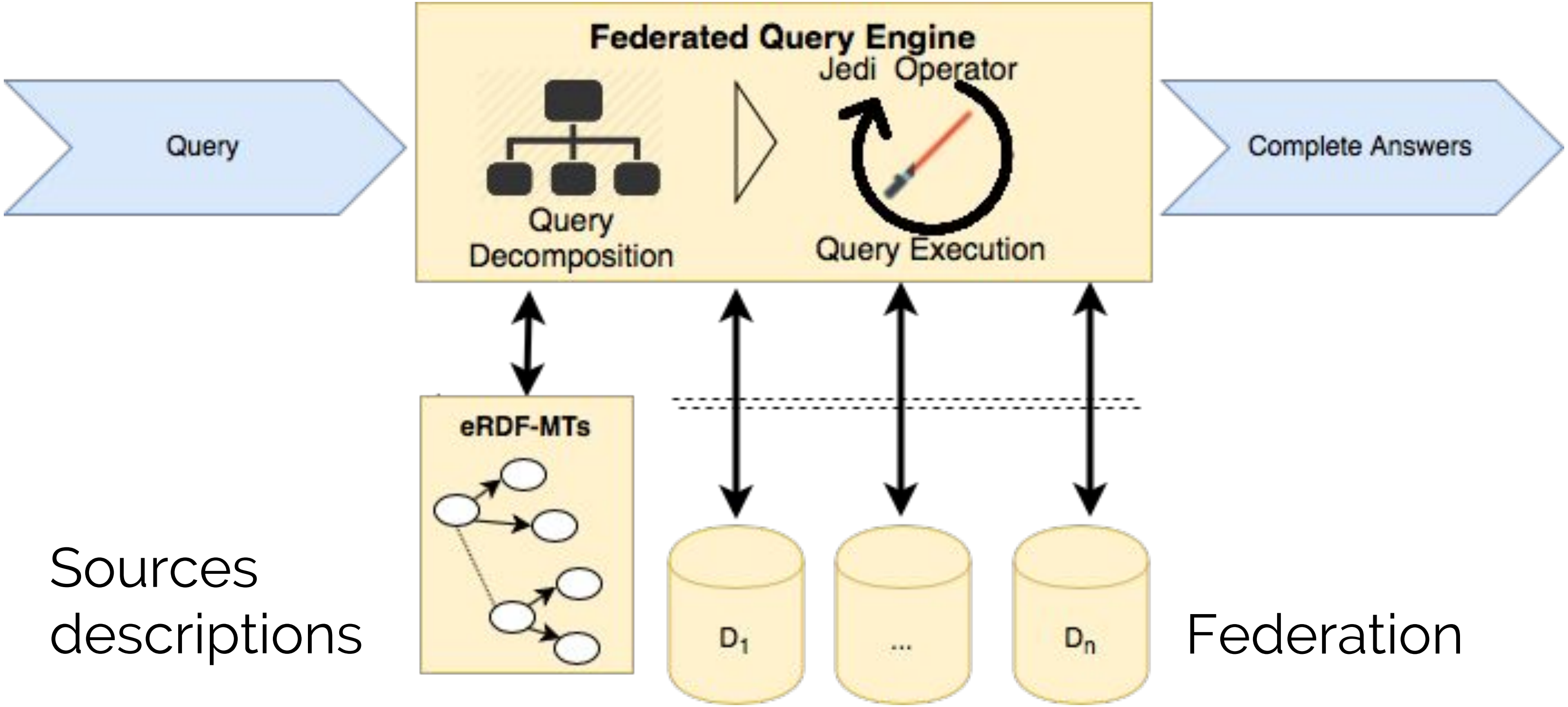
Thomas



David

Valentina

The Jedi Architecture



Sources descriptions

Federation