How to Best Find a Partner? 
An Evaluation of R2RML Editing Approaches

Christoph Pinkel¹, Carsten Binnig², Peter Haase¹, Clemens Martin², Kunal Sengupta³, Johannes Trame¹

¹fluid Operations AG - ²Cooperative State University Mannheim - ³Wright State University Dayton, OH
Relational DB to RDF Mappings
Relational DB to RDF Mappings
Relational DB to RDF Mappings
Relational DB to RDF Mappings

Christoph Pinkel — fluid Operations AG
Example: MusicBrainz — Music Ontology

- 149 tables
- 271 Fks
- 100 classes
- 169 object properties
Outline

Introduction & Background

R2RML Editing Approaches

User Study & Study Questions

Observations

Conclusion
R2RML Editing Approaches
R2RML Mapping Rules (*TriplesMaps*)
R2RML Mapping Rules (*TriplesMaps*)
R2RML Mapping Rules *(TriplesMaps)*
R2RML Mapping Rules (*TriplesMaps*)
R2RML Mapping Rules (*TriplesMaps*)
R2RML Mapping Rules (*TriplesMaps*)
R2RML Mapping Rules (*TriplesMaps*)

![Diagram](image-url)
R2RML Mapping Rules (*TriplesMaps*)
R2RML Mapping Rules (*TriplesMaps*)

:McCartney a mo:MusicArtist; foaf:name “Paul McCartney”
R2RML Mapping Rules (*TriplesMaps*)

:McCartney a mo:MusicArtist; foaf:name “Paul McCartney”
R2RML Mapping Rules (*TriplesMaps*)

![Diagram showing R2RML mapping rules with a specific example: 
:McCartney a mo:MusicArtist; foaf:name “Paul McCartney”]
R2RML Mapping Rules (*TriplesMaps*)

- LogicalTable
- SubjectMap
- PredicateObjectMap

:McCartney a mo:MusicArtist;
  foaf:name “Paul McCartney”
R2RML Mapping Rules (*TriplesMaps*)

```
:McCartney a mo:MusicArtist;
 foaf:name "Paul McCartney"
```

**LogicalTable**

**SubjectMap**

**PredicateObjectMap**

**TriplesMap**

**table:** artist

---

Christoph Pinkel — fluid Operations AG
**R2RML Mapping Rules (TriplesMaps)**

SELECT * FROM artist
JOIN artist_name

:McCartney a mo:MusicArtist;
foaf:name "Paul McCartney"

LogicalTable

SubjectMap

PredicateObjectMap
R2RML Mapping Rules (*TriplesMaps*)

```sql
SELECT * FROM artist 
JOIN artist_name 
```

```
:McCartney a mo:MusicArtist; 
foaf:name "Paul McCartney"
```
R2RML Mapping Rules (*TriplesMaps*)

```
:McCartney a mo:MusicArtist;
foaf:name "Paul McCartney"
```

![Diagram of R2RML Mapping Rules](image)
R2RML Mapping Rules (TriplesMaps)

It is important where you start

:McCartney a mo:MusicArtist;
  foaf:name "Paul McCartney"
Where to Start?

PredicateObjectMap

SubjectMap

LogicalTable
Where to Start?

:McCartney a mo:MusicArtist;
foaf:name “Paul McCartney”

1. PredicateObjectMap
2. SubjectMap
3. LogicalTable
Where to Start?

:McCartney a fo:MusicArtist;
foaf:name “Paul McCartney”

1. PredicateObjectMap
2. SubjectMap
3. LogicalTable

Ontology Driven Approach
Where to Start?

1. LogicalTable
2. SubjectMap
3. PredicateObjectMap

table: artist
Where to Start?

1. LogicalTable
2. SubjectMap
3. PredicateObjectMap

Database Driven Approach

table: artist
Editors: Supported Editing Approach
Editors: Supported Editing Approach
R2RML Editor
User Study
& Study Questions
User Study

Mapping **MusicBrainz** and **Music Ontology**

Three small mapping tasks, different characteristics

Each task consists of two steps: (a) concept, (b) property

43 participants

Different background knowledge (by self assessment)

Tasks & approaches shuffled
Mapping Tasks

- Artists with Name
  - Tricky Join

- Recording and Duration
  - Semantic Subtleties

- Tracks w/ Number on Album
  - Confusing Table Data
Question (1) — General Case Champion

In the general case, no approach outperforms the other.
Question (2) — User Background

The background of users influences results.
Question (3) — Task Characteristics

Task characteristics influence which approach works better.

Subtle Semantics?

Join?
Observations
Observations — General Case

(a) Time Taken

(b) Correctness

(c) Average Rating

\[ \sigma = 339 \quad \sigma = 364 \]

\[ \sigma = 0.46 \quad \sigma = 0.41 \]

\[ \sigma = 1.29 \quad \sigma = 1.31 \]
Observations — User Background

(a) By User Expertise  (b) By Ontology Skills  (c) By DB Skills
Observations — Task Characteristics

(a) Average Time

(b) Average Correctness
Conclusions

Editing approach influences result quality & time taken

No approach seems to be a “general champion”

Dependencies on user background and task characteristics

Editors should offer either approach, depending on task/user

Future work: how to identify task characteristics, mind set?

Thank you!