

# Updating Wikipedia via DBpedia Mappings and SPARQL

Vadim.Savenkov@wu.ac.at

joint work with **Albin Ahmeti**, **Javier Fernández** and **Axel Polleres**

European Semantic Web Conference @ Portoroz, Slovenia, Mai 30, 2017

**Armand Jean du Plessis de Richelieu**



Richelieu, par Philippe de Champaigne.

**Biographie**

**Naissance** 9 septembre 1585  
Paris (France)

**Décès** 4 décembre 1642 (à 57 ans)  
Paris (France)

**Cardinal de l'Église catholique**

**Créé cardinal** 5 septembre 1622  
par le pape Grégoire XV

**Titre cardinalice** Cardinal-prêtre

**Évêque de l'Église catholique**

**Consécration épiscopale** 17 avril 1607  
Par S.É. Anne de Pérusse d'Escars de Givry

**Évêque de Luçon**

1605 (confirmé le 18 décembre 1606) – 29 avril 1624

Alphonse-Louis du Plessis de Richelieu Emery de Bragelongne<sup>1</sup>

**Abbé de Cîteaux**  
(coadjuteur à partir de 1627)

19 novembre 1635 – 4 décembre 1642

Pierre III Nivelles Claude Vaussin

declarative mappings

D About: Cardinal Richelieu × +

dbpedia.org/page/Cardinal\_Richelieu

**About: Cardinal Richelieu**

An Entity of Type : cleric, from Named Graph : http://dbpedia.org, within Data Space : dbpedia.org

Armand Jean du Plessis, Cardinal Duke of Richelieu and of France (furmã zã du plesil; 9 September 1585 – 4 December 1642) was a French statesman, diplomat, and cardinal. He was the chief minister to Louis XIII and Louis XIV, and was instrumental in the formation of the French state. He was often known by the title of the King's "Chief Minister" or "First Minister". He sought to consolidate

S	P	O
dbr:Cardinal_Richelieu	dbo:religion	dbr:Catholic_Church
dbr:Concino_Concini	dbo:successor	dbr:Cardinal_Richelieu
dbr:Château_de_Richelieu	dbp:owner	dbr:Cardinal_Richelieu

**Property**

dbo:abstract

Armand-Jean du Plessis, Premier Duc de Richelieu (\* 9. September 1585 in Paris; † 4. Dezember 1642 in Paris), kurz Kardinal Richelieu, war ein französischer Aristokrat, Kirchenfürst und Staatsmann. Von 1624 bis zu

- ✓ Semantically enriches the exported data with a TBox
- ✓ Partly materialized (membership for superclasses)
- ✓ References external ontologies

# Motivation for batch updates

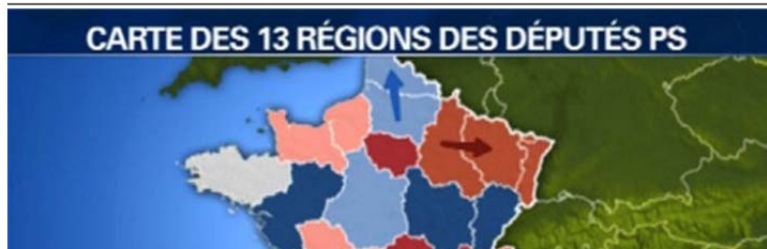
## The Telegraph



HOME > NEWS > WORLD NEWS > EUROPE > FRANCE

Plans to redraw map of France's regions described as 'butchery' as parliamentary debate opens

France's National Assembly begins delicate task of debating cutting the country's regions from 22 to as few as 13, as critics say it amounts to "butchery"



```
DELETE {
```

```
?X dbo:region dbr:Upper_Normandy .  
?Y dbo:region dbr:Lower_Normandy . }
```

```
INSERT {
```

```
?X dbo:region dbr:Normandy .  
?Y dbo:region dbr:Normandy }
```

```
WHERE {
```

```
{?X dbo:region dbr:Upper_Normandy}  
UNION  
{?Y dbo:region dbr:Lower_Normandy}  
}
```

## *Ontology Based Data Management (OBDM)*

Ontology as a unified interface for querying and updating the data in external sources.

# Updates+Entailment: First Steps

Without mappings: updates can be unambiguously resolved for some simple ontology languages.

## Example: $\text{RDFS}_{\neg}$ = RDFS core + Class disjointness

[ESWC 2016: Ahmeti, Calvanese, Polleres, S.]

- **Subclass**  $A \sqsubseteq B$ , **subproperty**  $p \sqsubseteq q$

- **Domain and range restrictions**

domain of  $p$  is restricted to  $A$ :  $\exists p \sqsubseteq A$

range of  $p$  is restricted to  $A$ :  $\exists p^{-1} \sqsubseteq A$

- **Class disjointness**  $A \sqsubseteq \neg B$

## RDFS $\neg$ +

- Inverse properties  $p \sqsubseteq r^-$
- Property disjointness  $p \sqsubseteq \neg r$
- Functionality (**data properties only**)
- DifferentFrom  $c_1 \neq c_2$
- sameAs  $c_1 = c_2$  (to link **external ontologies**)
- redirectsTo: equality with the **canonical representative** of each equivalence class

# French Catholic Clergy mapping

```
Mapping fr: Infobox Prélat catholique{ { ConditionalMapping
| cases =
  {{Condition | templateProperty = titre | operator = equals | value = Pape
    | mapping = {{TemplateMapping | mapToClass = Pope | mappings =
      {{IntermediateNodeMapping | nodeClass = PersonFunction | correspondingProperty = occupation
        | mappings =
          {{ConstantMapping | ontologyProperty = title | value = Pape }}
          {{PropertyMapping | templateProperty = début pontificat | ontologyProperty = activeYearsStartYear }}
          {{PropertyMapping | templateProperty = fin pontificat | ontologyProperty = activeYearsEndYear }}
          ...
          {{PropertyMapping | templateProperty = prédécesseur pape | ontologyProperty = predecessor }}
          {{PropertyMapping | templateProperty = successeur pape | ontologyProperty = successor }}
        }}
      }}
    }}
  }}
  {{Condition | templateProperty = titre | operator = equals | value = Évêque
    | mapping = {{TemplateMapping | mapToClass = ChristianBishop }}
  }}
  ...
  {{Condition | operator = otherwise | mapping = {{TemplateMapping | mapToClass = Cleric }} }}
}

| defaultMappings =
  {{PropertyMapping | templateProperty = nom | ontologyProperty = foaf:name }}
  ...
  {{PropertyMapping | templateProperty = nom naissance | ontologyProperty = birthName }}
  {{PropertyMapping | templateProperty = devise | ontologyProperty = motto }}
  ...
  {{PropertyMapping | templateProperty = bénédiction abbatiale | ontologyProperty = abbeychurchBlessing }}
}}
```

# TemplateMapping and PropertyMapping

Infobox type

```
Mapping fr: Infobox Prélat catholique { { ConditionalMapping
| cases =
  {{Condition | templateProperty = titre | operator = equals | value = Pape
    ...
  }}
  {{Condition | templateProperty = titre | operator = equals | value = Évêque
    | mapping = {{TemplateMapping | mapToClass = ChristianBishop }}
  }}
  {{Condition | templateProperty = titre | operator = equals | value = Cardinal
    | mapping = {{TemplateMapping | mapToClass = Cardinal }}
  }}
  ...
  {{Condition | operator = otherwise
    | mapping = {{TemplateMapping | mapToClass = Cleric }}
  }}
| defaultMappings =
  {{PropertyMapping | templateProperty = nom | ontologyProperty = foaf:name }}
  ...
  {{PropertyMapping | templateProperty = devise | ontologyProperty = motto }}
  ...
}}
```

<Page URI > a dbr:Pope.

<Page URI > a dbr:ChristianBishop.

<Page URI > a dbr:Cardinal.

<Page URI > a dbr:Cleric.

# PropertyMapping Example

```
Mapping fr:Infobox Prélat catholique{ { ConditionalMapping
| cases =
...
{{Condition | templateProperty = titre | operator = equals | value = Évêque
| mapping = {{TemplateMapping | mapToClass = ChristianBishop }}
}}
...
{{Condition | operator = otherwise
| mapping = {{TemplateMapping | mapToClass = Cleric }}
}}
| defaultMappings =
{{PropertyMapping | templateProperty = nom | ontologyProperty = foaf:name }}
...
{{PropertyMapping | templateProperty = devise | ontologyProperty = motto }}
...
}}
```

<Page URI > foaf:name <nom>.

<Page URI > dbo:motto <devise>.



# IntermediateNodeMapping

```
{ { PropertyMapping  
  | templateProperty = devise | ontologyProperty = motto } }
```

<Page URI > dbo:motto <devise >.

## { { **IntermediateNodeMapping**

```
| nodeClass = PersonFunction  
| correspondingProperty = occupation  
| mappings =
```

<**Fresh IRI**> a dbr:PersonFunction.

```
{ { PropertyMapping  
  | templateProperty = ministère 1 | ontologyProperty = title } }
```

<Page URI > dbo:occupation <**Fresh IRI**>.

<**Fresh IRI**> dbo:title < ministère 1 >.

[...]

```
{ { PropertyMapping  
  | templateProperty = successeur 1 | ontologyProperty = successor  
  } }
```

<**Fresh IRI**> dbo:successor <successeur 1 >.

```
}}
```

# Tuple Generating Dependencies (TGDs)

- Basic (full) tgds
  - $\forall \vec{x} (\psi(\vec{x}) \rightarrow CQ(\vec{x}))$
  - $\forall x_1 \forall x_2 (Cleric(x_1) \wedge WorksFor(x_1, x_2) \rightarrow Church(x_2))$
- $CQ(\vec{x})$  is a conjunctive query with free variables  $\vec{x}$ .
- $\psi$  is a FO formula (often also just a CQ).
- Enforcing (aka “chasing”) tgds:
  - For each assignment  $\vec{a}$  for  $\vec{x}$  satisfying  $\psi$  on the source instance, yield the facts instantiating  $CQ(\vec{a})$ .

# The Infobox Schema **W**

- $W_i$  (Page\_**U**RI, Infobox\_**T**ype, Infobox\_**I**D)
- $W_d$  (Infobox\_**I**D, **P**roperty\_Name, Property\_**V**alue)
- Infobox constraints:
  - $UT \rightarrow I$
  - $IP \rightarrow V$
  - $W_d.I \subseteq W_i.I$

# Mapping rules amenable to full TGDs

Mapping fr: Infobox **Prélat catholique** { { ConditionalMapping

$$\forall U \forall I \left( W_i(U, \text{"Prélat catholique"}, I) \wedge W_d(I, \text{"Èvêque"}, X) \rightarrow dbr: \text{ChristianBishop}(U) \right)$$

```
{ { Condition | templateProperty = titre | operator = equals | value = Èvêque  
  | mapping = { { TemplateMapping | mapToClass = ChristianBishop } }  
}
```

$$\forall U \forall I \left( W_i(U, \text{"Prélat catholique"}, I) \wedge W_d(I, \text{"Cardinal"}, X) \rightarrow dbr: \text{Cardinal}(U) \right)$$

```
{ { Condition | templateProperty = titre | operator = equals | value = Cardinal  
  | mapping = { { TemplateMapping | mapToClass = Cardinal } }  
}
```

...

```
{ { Condition | operator = otherwise  
  | mapping = { { TemplateMapping | mapToClass = Cleric } }  
}
```

$$\forall U \forall I \forall X \left( W_i(U, \text{Prélat catholique}, I) \wedge \neg W_d(I, \text{"Pape"}, X) \wedge \neg W_d(I, \text{"Èvêque"}, X) \wedge \neg W_d(I, \text{"Cardinal"}, X) \wedge \dots \rightarrow dbr: \text{Cleric}(U) \right)$$

# PropertyMapping Example

```
Mapping fr:Infobox Prélat catholique{ { ConditionalMapping
| cases =
...
{{ Condition | operator = otherwise
      | mapping = {{ TemplateMapping | mapToClass = Cleric }}
}}

| defaultMappings =
{{ PropertyMapping | templateProperty = nom | ontologyProperty = foaf:name }}
...

```

$$\forall U \forall I (W_i(U, \text{"Prélat catholique"}, I) \wedge W_d(I, \text{"nom"}, X) \rightarrow foaf:name(U, X))$$

```

{{ PropertyMapping | templateProperty = devise | ontologyProperty = motto }}

```

$$\forall U \forall I (W_i(U, \text{"Prélat catholique"}, I) \wedge W_d(I, \text{"devise"}, X) \rightarrow dbp:motto(U, X))$$

```

}}
```

# TGDs with existentially quant. variables

- $\forall \vec{x} (\psi(\vec{x}) \rightarrow \exists \vec{y} CQ(\vec{x}, \vec{y}))$
  
- $\forall x (Cleric(x) \rightarrow \exists z PraysFor(x, z))$
  
- Chase with non-full TGDs:
  - For each assignment  $\vec{a}$  for  $\vec{x}$  satisfying  $\psi$ ,  
 construct an assignment  $\vec{b}$  of **fresh distinct values** for  $\vec{y}$   
 (often special **labeled null** values);
  - Yield the atoms in  $CQ(\vec{a}, \vec{b})$

# IntermediateNodeMapping via (non full) TGDs

$$\forall U \forall I (W_i(U, \text{"Prélat catholique"}, I) \wedge W_d(I, \text{"ministère 1"}, X)) \\ \rightarrow \exists Z \text{ dbo:title}(Z, X)$$
$$\forall U \forall I (W_i(U, \text{"Prélat catholique"}, I) \wedge W_d(I, \text{"prédécesseur1"}, X)) \\ \rightarrow \exists Z \text{ dbo:predecessor}(Z, X)$$

Chase does not enforce that the same value instantiates  $Z$  in both cases.

# Nested TGDs

- $\forall \vec{x}_1 \left( \psi_1(\vec{x}_1) \rightarrow \exists \vec{y}_1 \left( CQ_1(\vec{x}_1, \vec{y}_1) \wedge \forall \vec{x}_2 \left( \psi_2(\vec{x}_1, \vec{x}_2) \rightarrow \exists \vec{y}_2 (CQ_2(\vec{x}_1, \vec{x}_2, \vec{y}_1, \vec{y}_2) \wedge \dots) \right) \right) \right) \right)$
  
- $\forall x_1 \left( Pastor(x_1) \rightarrow \exists y \left( PraysFor(x_1, y) \wedge \forall x_2 (ParishMember(x_2, x_1) \rightarrow PraysFor(x_2, y)) \right) \right)$
  
- Chase with nested TGDs vs. standard TGDs:
  - Preconditions of nested implications can access the values bound in the preconditions of ancestor implications: e.g.,  $ParishMember(x_2, x_1)$
  - Fresh values generated in the ancestor implications can be reused in the conclusions of nested implications, e.g.:  $PraysFor(x_2, y)$



# IntermediateNodeMapping via Nested TGDs

```
{ { PropertyMapping  
  | templateProperty = devise | ontologyProperty = motto } }
```

```
{ { IntermediateNodeMapping  
  | nodeClass = PersonFunction  
  | correspondingProperty = occupation  
  | mappings =
```

```
    { {  
       $\forall U \forall I (W_i(U, \text{"Prélat catholique"}, I) \rightarrow$   
         $\exists Z (dbr: PersonFunction(Z) \wedge dbo: occupation(U, Z)$   
           $\wedge (W_d(I, \text{"ministère 1"}, X) \rightarrow dbo: title(Z, X))$   
          .....  
           $\wedge (W_d(I, \text{"prédécesseur 1"}, X) \rightarrow dbo: predecessor(Z, X))$   
           $\wedge (W_d(I, \text{"successeur 1"}, X) \rightarrow dbo: successor(Z, X)))$   
    } }
```

```
}}}
```

# DBpedia Mappings: syntactic restrictions

- Only unary interpreted predicates in conditions (e.g., "if name = full\_name" not possible)
- NO nested conditions
  - At most two atoms on the left-hand side of implications (test condition + atomic query of a property value)
- IntermediateNodeMappings are not nested currently.
- Fixed structure of generated triples:
  - Subject: page URI or an intermediate node
  - Predicate: defined by the property mapping
  - Object: infobox property value

# Dbpedia mappings: Expressive features

- Limited forms of negation:
  - only the first satisfied condition triggers,
  - the final „otherwise“ branch
- Generation of new IRIs (existentially quantified variables in the encoding via tgds)
- Special functions:  
(“contains”, “exists”, “not exists” in conditions),  
string & geocoordinate manipulation,  
arithmetic operations for property generation

# DBpedia update translation

# Semantics of updates

- Input: Dbpedia update  $u = (u^-, u^+)$
- $u^-$  atoms to be deleted
- $u^+$  atoms to be inserted
- Semantics: a set of possible Wiki updates  $(e_i^-, e_i^+)$  that

## ***Consistency-oblivious semantics***

- Comply with the Infobox schema
- Minimal (e.g., w.r.t. the subset relation)
- Together with the mappings and the TBox, derive a DBpedia instance which contain  $u^+$  and does not contain  $u^-$

## ***Consistency-aware semantics***

- Updated instance is consistent with the TBox

# Existence of solutions (consistency-oblivious)

## For a fixed mapping:

Given an insert  $u^+$ , is its consistency-oblivious semantics empty?

Standard NP completeness proof (in DB context Abiteboul&Duschka 98)

Underlying data source (Database): **colored** graph.

View (ontology): non-colored graph, colors, and **invalid coloring marker** set to false: require a valid coloring

$(\exists C_1 \exists C_2 \exists A \exists B$

$Edge(A, B) \wedge Color(A, C_1) \wedge Color(B, C_2) \wedge C_1 = C_2) \rightarrow Invalid$

No binary predicates (e.g.  $C_1 = C_2$ ) in the conditions of DBpedia mappings!

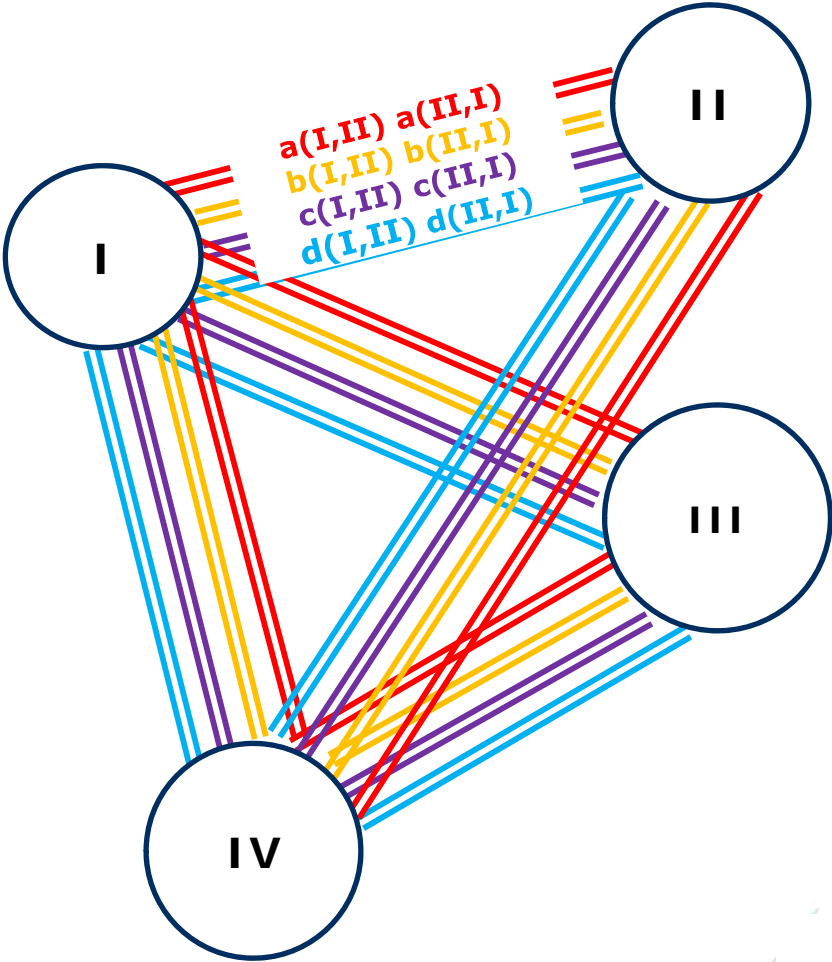
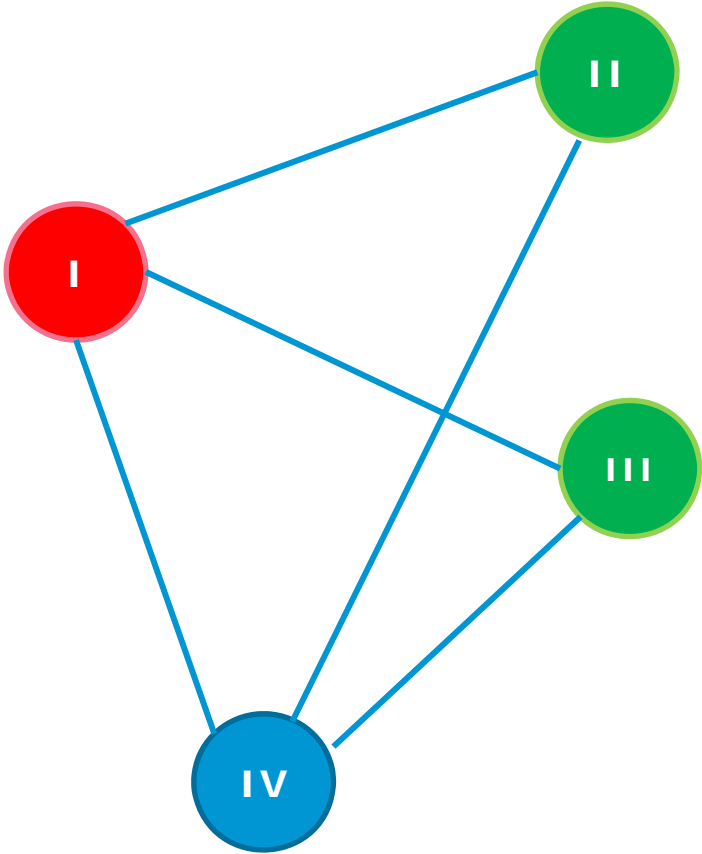
# Source consistency of the ABox: Reduction from 3COL

- Graph of degree  $\leq 4$  (intractable 3COL)
  - Proper ABox shape: IRIs connected by symmetric labeled properties **a**, **b**, **c**, **d**
- $\Rightarrow$  8 property assertions to represent a single graph edge in the materialized ABox.



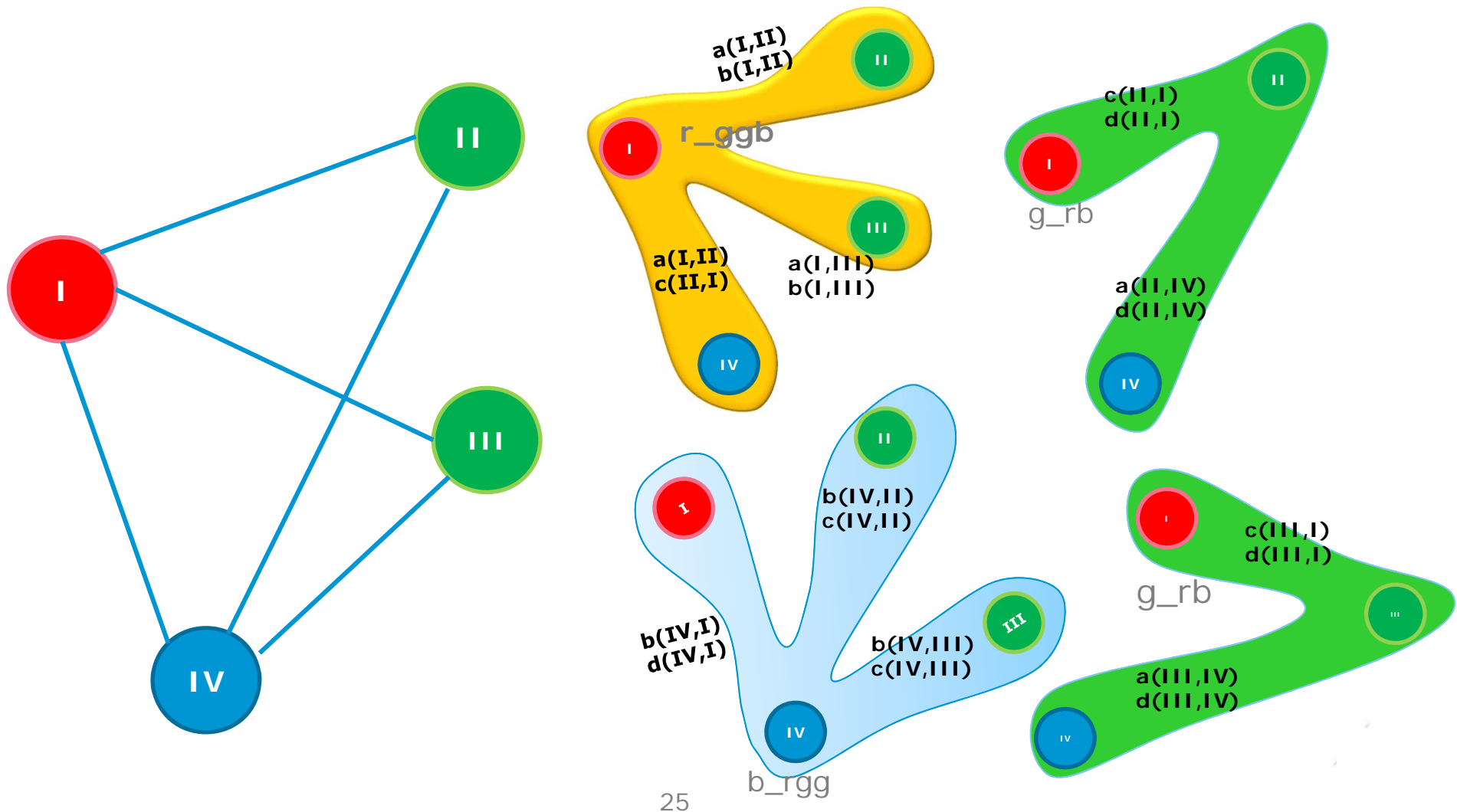
- TBox:  $\mathbf{a} \sqsubseteq \mathbf{a}^-$ ,  $\mathbf{b} \sqsubseteq \mathbf{b}^-$ ,  $\mathbf{c} \sqsubseteq \mathbf{c}^-$ ,  $\mathbf{d} \sqsubseteq \mathbf{d}^-$

# Colored graph and its representation as a Dbpeida ontology





# Infoboxes: 1-Neighborhoods of nodes



# Mapping conditions: ensure consistent color assignment across Infoboxes

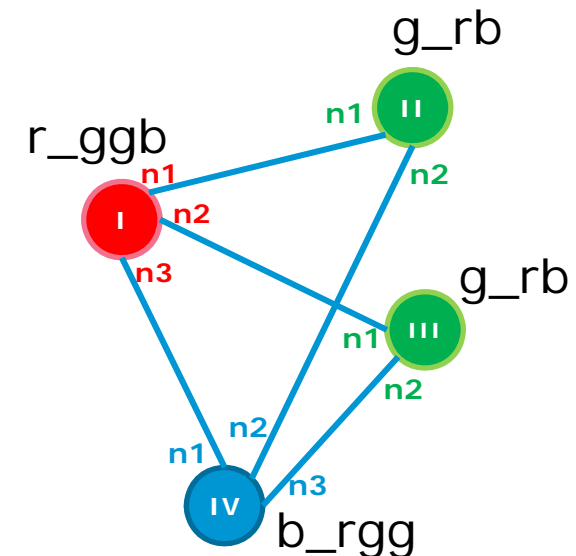
if neighbourhood type = „**r\_ggb**“

<this> dbo:a <neighbor n1>.  
 <this> dbo:b <neighbor n1>.  
 <this> dbo:a <neighbor n2>.  
 <this> dbo:b <neighbor n2>.  
 <this> dbo:a <neighbor n3>.  
 <this> dbo:c <neighbor n3>.

	r	g	b
r		ab	ac
g	cd		ad
b	bd	bc	

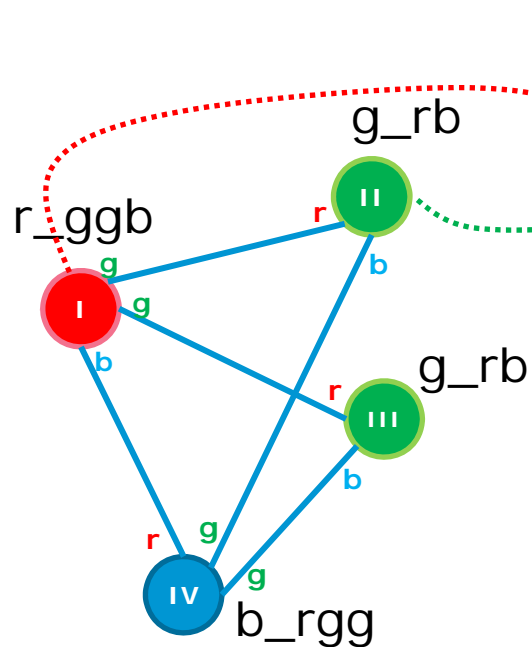
if neighbourhood type = „**b\_rgg**“

<this> dbo:b <neighbor n1>.  
 <this> dbo:d <neighbor n1>.  
 <this> dbo:b <neighbor n2>.  
 <this> dbo:c <neighbor n2>.  
 <this> dbo:b <neighbor n3>.  
 <this> dbo:c <neighbor n3>.

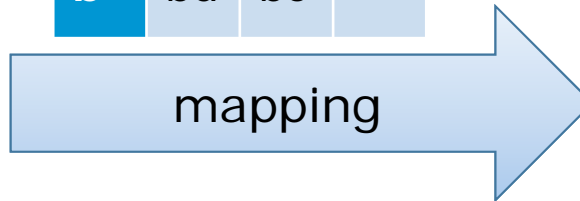


# 3COL of max. degree 4 graphs via Source consistency of the ABox

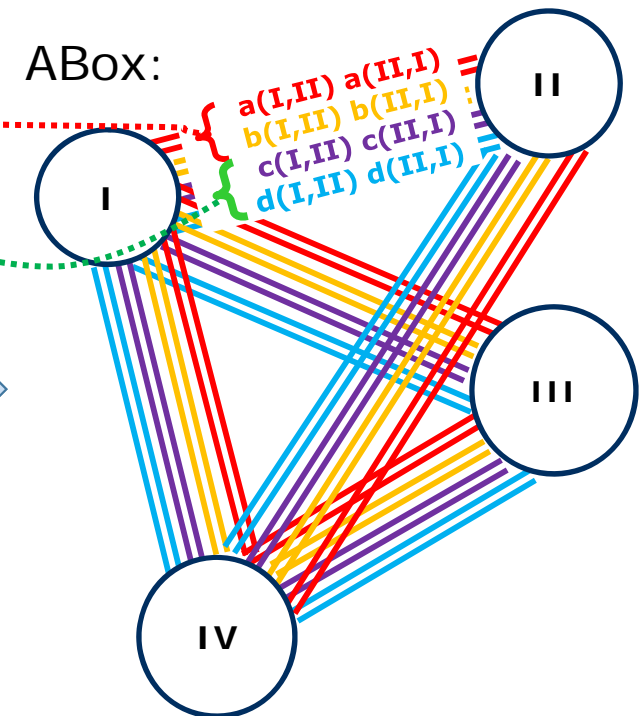
Wiki Infoboxes:



	r	g	b
r		ab	ac
g	cd		ad
b	bd	bc	



TBox:  $a \sqsubseteq a^-$ ,  $b \sqsubseteq b^-$ ,  
 $c \sqsubseteq c^-$ ,  $d \sqsubseteq d^-$



3COL iff consistent coverage by 1-neighborhoods (Infoboxes) exist  
 iff ABox has regular shape.

# Worst-case vs. actual mappings

- DBpedia and many Linked Open Data sources [Glimm,Hogan,Krötzsch,Polleres LDOW'12] is economic in terms of the TBox features used:
  - Especially with respect to inconsistency derivation:
  - Few disjoint classes / properties.
  - Functionality only for data properties.
- Computational intractability is less of a practical issue than infeasibility for verification by curators (Wikipedia maintainers)!

# Our Approach

- Consider each subject (Wiki page) separately
- Find the most likely translation of properties based on evidence, applying update resolution policies
- Construct sound re-usable **patterns of update translation**
- Use patterns to implement batch updates

# Update resolution policies

```
{{ PropertyMapping | templateProperty = state1 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = state2 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = state3 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = state4 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = state5 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = state6 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = state7 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = state8 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = state9 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = district | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = district1 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = district2 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = district3 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = district4 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = district5 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = district6 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = district7 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = district8 | ontologyProperty = region }}
{{ PropertyMapping | templateProperty = district9 | ontologyProperty = region }}
```

# Update resolution policies

## *Infobox frequency first*

- Prefer most frequently used properties
- Based on precomputed statistics across all infoboxes

```
{{ PropertyMapping | templateProperty = state7 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = state8 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = state9 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = district | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = district1 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = district2 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = district3 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = district4 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = district5 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = district6 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = district7 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = district8 | ontologyProperty = region }}  
{{ PropertyMapping | templateProperty = district9 | ontologyProperty = region }}
```

# Update resolution policies

## *Infobox frequency first*

- Prefer most frequently used properties
- Based on precomputed statistics across all infoboxes

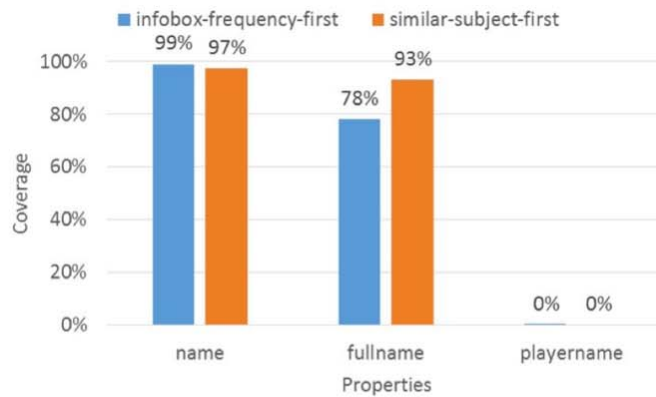
```
{ { PropertyMapping | templateProperty = state / | ontologyProperty = region } }
```

## *Similar subject first*

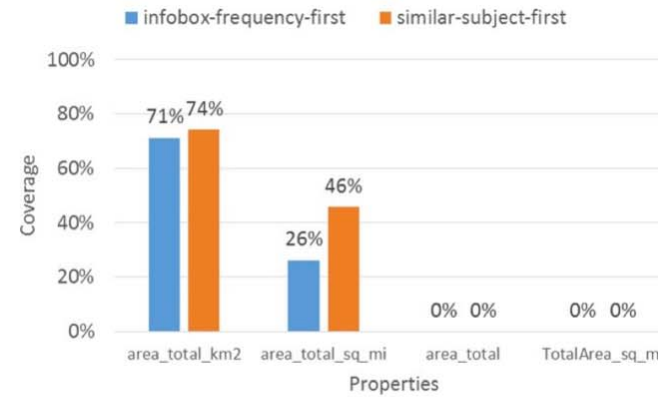
- Taking only similar subjects into account: same infobox type and having same properties as being inserted.
- “How are other football players are represented in the infobox?”



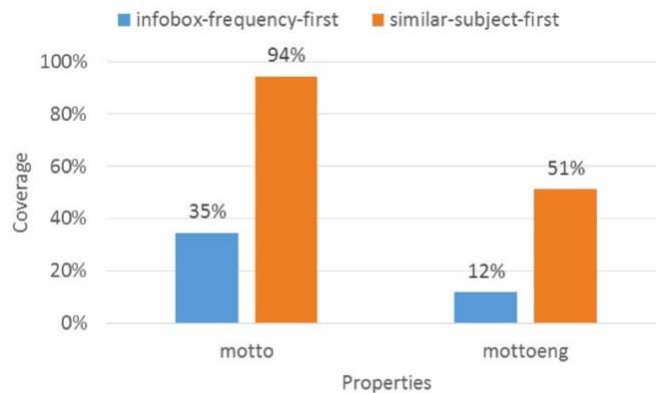
# Examples of resolution policies



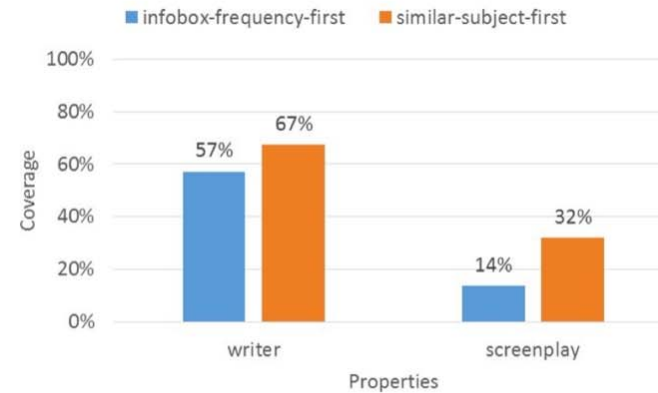
(a) Infobox Football biography



(b) Infobox Settlement



(c) Infobox Universities



(d) Infobox Film

# Update resolution patterns

- GOAL: input update  $(u^-, u^+)$  for a single subject
- CONTEXT: relevant pre-existent facts that need to be removed to avoid clashes
- ACTION: single wiki update  $(e^-, e^+)$

IRIs and data values replaced by variables, to enable reuse across Wiki pages in bulk updates.

# Summary and conclusions

- DBpedia mappings: fixed-depth nested tgds with negation and unary interpreted predicates in preconditions.
- Existence of solutions (DBpedia update feasibility) is intractable even not taking consistency w.r.t. TBox into account.
- Evidence-based update resolution policies.
- Introduce update resolution patterns as a means to formulate **reusable examples of update translation** in OBDM.