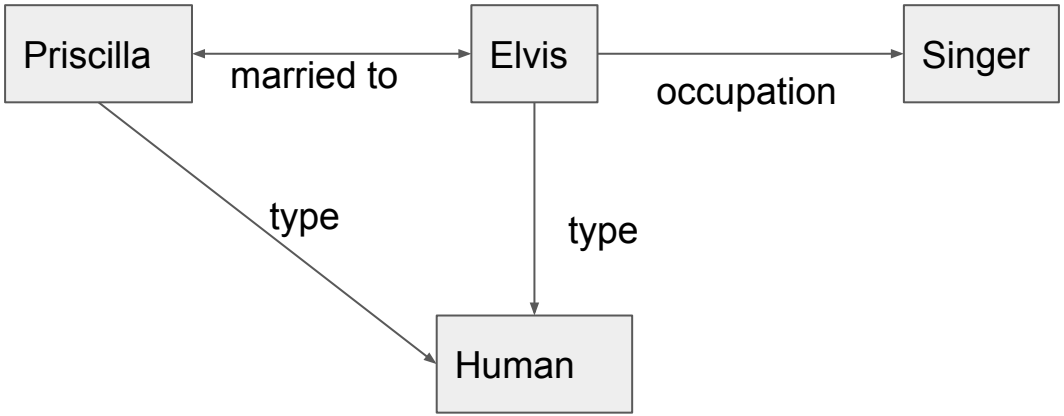
4

a reason-able knowledge base

Thomas Pellissier-Tanon, Gerhard Weikum, and Fabian Suchanek

http://yago-knowledge.org/

Knowledge base



Challenges:

- Data
- *Good* data (>95% accuracy)

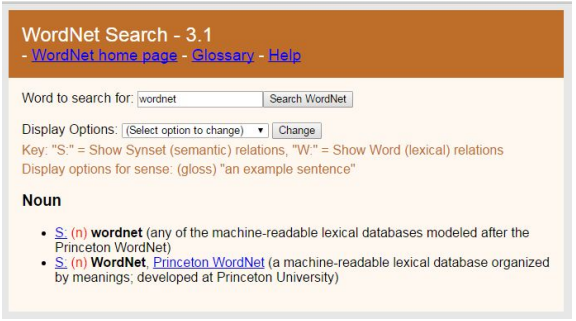
Usages:

- Search (Google…)
- Analytics (fraud…)
- Question answering

Yago, a large knowledge base

- First version in 2007
  Seoul Test of Time Award of WWW 2018
- Extracted from Wikipedia
  with a focus on precision
- Used by IBM Watson, DBpedia, ...
- 17M entities, 150M facts, 350,000 classes

Yago 3 (previous version)



10 **Wikipedia** languages + **Wordnet**

Great source of facts
150M facts

Great source of classes
350k classes

Extraction & combination & cleanup pipeline

95% accuracy

Wikipedia vs Wikidata

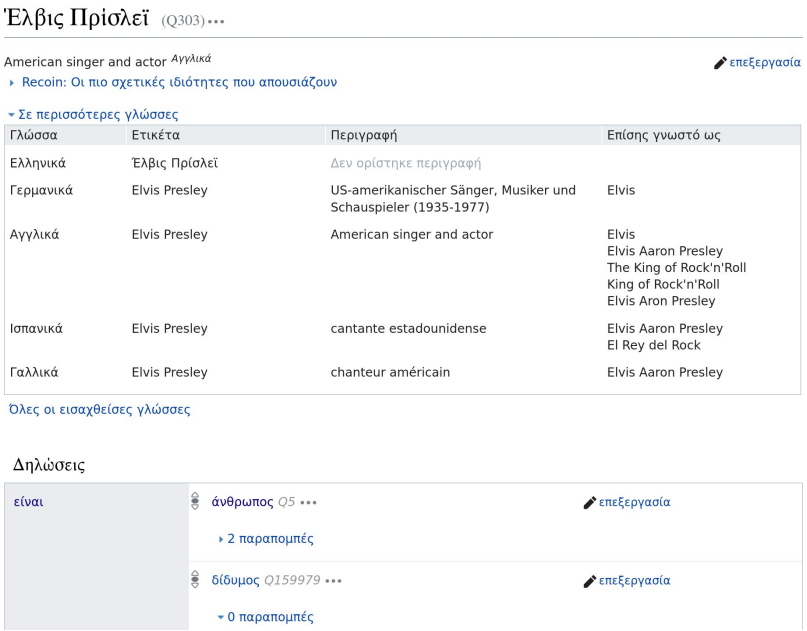
- 5M entities (English) >15M (all)
- Wikitext
- Large contributor community
- 300 languages
- Custom infoboxes
- Article titles (e.g. “Elvis Presley”)

- 70M entities
- Structured data
- Large contributor community
- 430+ languages
- 7k relations
- Identifiers (e.g. “Q303”)

Yago 3 cannot grow beyond Wikipedia
Why not use Wikidata?

Wikidata: Inconveniences for everyday use

1. Opaque identifiers



Έλβις Πρίσλεϊ (Q303) •••

American singer and actor Αγγλικά — επεξεργασία

Recoin: Οι πιο σχετικές ιδιότητες που απουσιάζουν

Σε περισσότερες γλώσσες

Γλώσσα	Ετικέτα	Περιγραφή	Επίσης γνωστό ως
Ελληνικά	Έλβις Πρίσλεϊ	Δεν ορίστηκε περιγραφή	
Γερμανικά	Elvis Presley	US-amerikanischer Sänger, Musiker und Schauspieler (1935-1977)	Elvis
Αγγλικά	Elvis Presley	American singer and actor	Elvis Elvis Aaron Presley The King of Rock'n'Roll King of Rock'n'Roll Elvis Aron Presley
Ισπανικά	Elvis Presley	cantante estadounidense	Elvis Aaron Presley El Rey del Rock
Γαλλικά	Elvis Presley	chanteur américain	Elvis Aaron Presley

Όλες οι εισαχθείσες γλώσσες

Δηλώσεις

είναι	άνθρωπος Q5 •••	επεξεργασία
	2 παραπομπές	
	δίδυμος Q159979 •••	επεξεργασία
	0 παραπομπές	

`wd:Q303` ≈ Elvis Presley

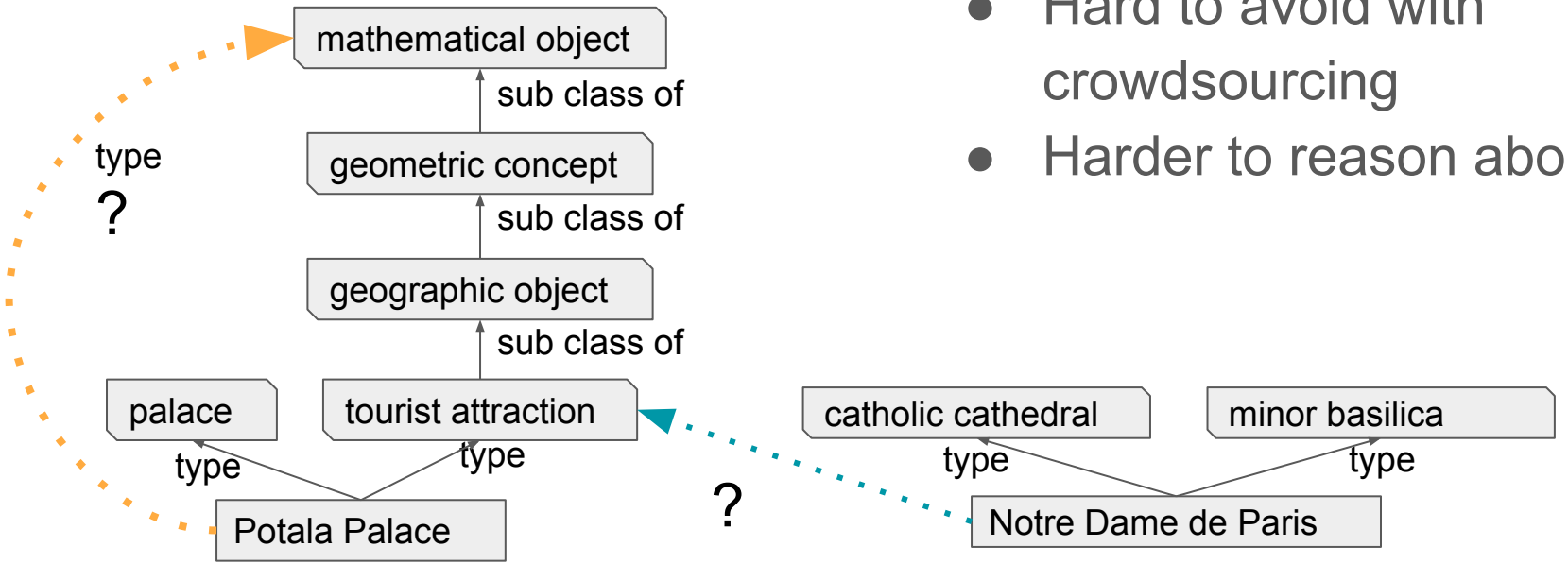
`wd:Q5` ≈ “human”

`wdt:P31` ≈ “type”

- Good with an internationalized UI
- Harder when playing with the data

Wikidata: Inconveniences for everyday use

1. Opaque identifiers
2. Taxonomy problems



- Good to represent all information
- Hard to avoid with crowdsourcing
- Harder to reason about

Wikidata: Inconveniences for everyday use

1. Opaque identifiers
2. Taxonomy problems
3. Constraint violation

Constraint violations:

- 5M functional property (“single value”)
- 1M domain (“‘type”)

- Needed to represent all information
- Hard to avoid with open contribution
- Harder to reason about

A solution: schema.org ?

- Large set of RDF classes (629) and properties (943)
- Actively maintained by a W3C user group
- Massive adoption for web pages annotations (>1M websites)

But no instances

Yago 4: Combining the best of both

Wikidata

- Millions of facts
- Multilingual labels/descriptions
- `rdf:type` relation

schema.org

- Classes (`schema:Person`)
- Properties (`schema:birthDate`)

+ manually defined consistency constraints

Yago 4: What it looks like

Yago 4 addresses Wikidata challenges

Yago 4:

1. ~~Opaque identifiers~~
2. Taxonomy problems
3. Constraint violation

- schema.org URIs
  e.g. `schema:Person`

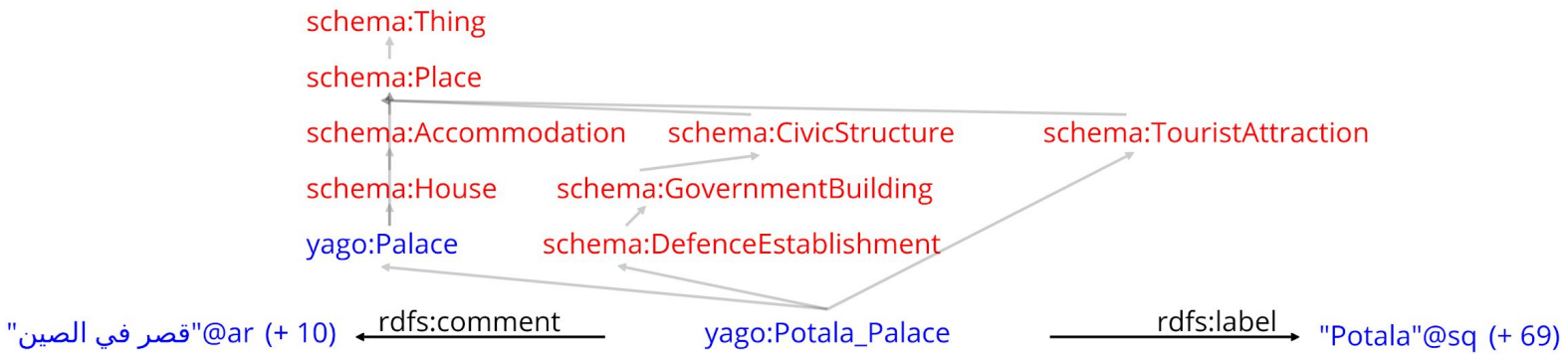
- URIs from Wikipedia articles names
  e.g. `yago:Elvis_Presley`

Yago 4 addresses Wikidata challenges

Yago 4:

1. ~~Opaque identifiers~~
2. ~~Taxonomy problems~~
3. Constraint violation

- Clean schema.org hierarchy
- Leaf classes from Wikidata (no meaning drift)



Yago 4 addresses Wikidata challenges

Yago 4:

~~1. Opaque identifiers~~
~~2. Taxonomy problems~~
~~3. Constraint violation~~

We enforce basic constraints

- Domain and range
- Functionality
- Disjointness

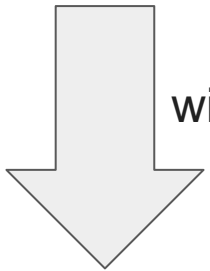
by removing the violating facts.

OWL-DL reasoning is possible

Yago 4: Reasoning example on >300M facts

“Is Elvis born in Memphis?”

```
yago:Elvis_Presley schema:birthPlace yago:Tupelo .
   schema:birthPlace a owl:FunctionalProperty .
```

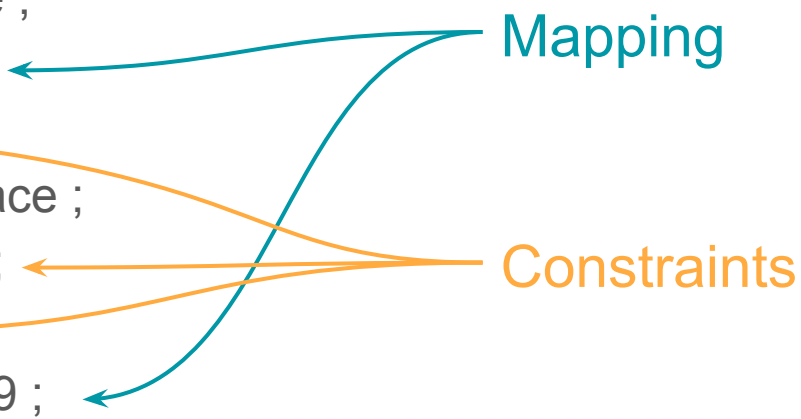


with unique name assumption

No

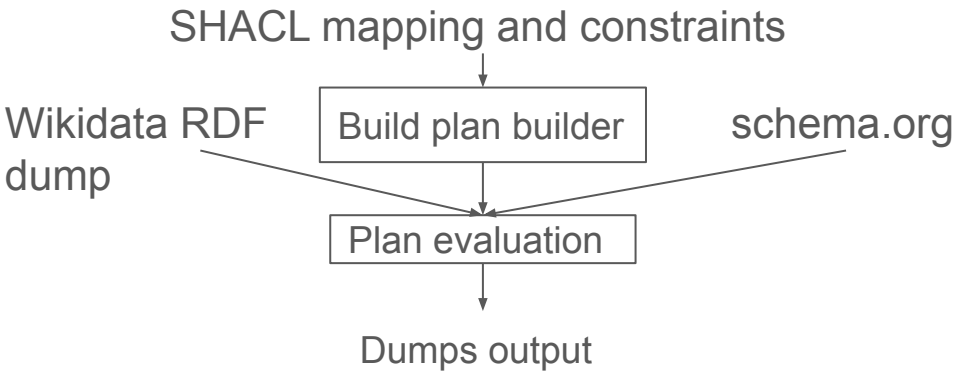
Yago 4: Schema enforcement and mapping: SHACL

```
schema:Person a sh:NodeShape ;
        ys:fromClass wd:Q215627 ;
        sh:property [
                sh:path schema:birthPlace ;
                sh:node schema:Place ;
                sh:maxCount 1 ;
                ys:fromProperty wdt:P19 ;
        ] .
```



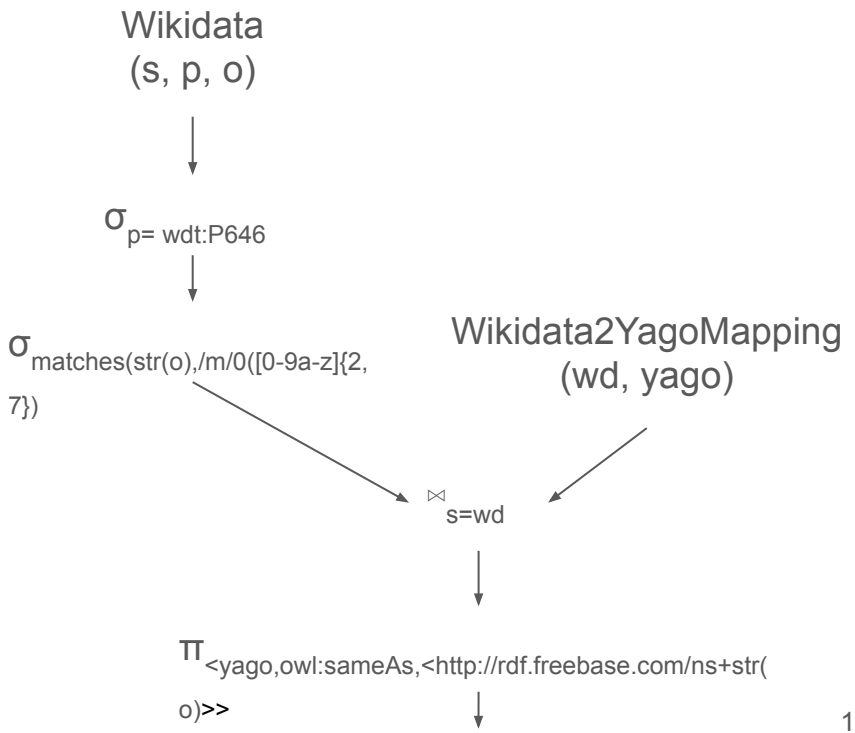
Yago 4: Architecture

- Build plan ≈ query plan
- Building Yago 4
  ≈ evaluating a query
- Does not strongly depend
  on Wikidata + schema.org



Example: Freebase sameAs

σ = filter
π = projection
⋈ = inner join



Yago 4 knowledge base: Stats

- 10k classes
- 120 relations
- 67M entities
- 340M facts
- 300M labels
- 1400M descriptions
- 132M facts removed by constraints

Two smaller flavors:

- All Wikipedias:
  15M entities 48M facts
- English Wikipedia:
  5M entities 15M facts

Yago 4: fact annotations

- RDF* annotations
- Based on Wikidata qualifiers
- only temporal for now

```
<< yago:Elvis_Presley schema:spouse yago:Priscilla_Presley >>
      schema:startDate “1967-05-01”^^xsd:date ;
      schema:endDate “1973-10-09”^^xsd:date .
```

Yago 4 knowledge base: How to use it?

- Web browsing
- Dump download
- SPARQL endpoint
- Open source build pipeline

All on

http://yago-knowledge.org

Yago 4: a Reason-able Knowledge Base

- **Simple schema** based on schema.org
- **OWL-DL compliant**, data cleanup based on constraints
- **340M facts**
- **Linked-open data 5☆**, SPARQL, URI dereferencing…

http://yago-knowledge.org