Building and Querying Semantic Layers for Web Archives

Pavlos Fafalios, Helge Holzmann, Vaibhav Kasturia, Wolfgang Nejdl

fafalios@l3s.de

L3S Research Center, University of Hannover, Germany

Introduction

- Web Archives
  - Valuable sources for research in many disciplines
    - *Digital humanities, historical sciences, journalism, sociology, …*
  - Allow inspecting how entities and events were reflected on the Web in different time periods

- We consider a broader notion of a web archive (not only web pages):
  - Web Archives (versioned web pages)
  - News Archives (non-versioned articles)
  - Social Media Archives (non-versioned, e.g., tweets)

- Accessing Web Archives
  - Limited query and exploration capabilities
  - Difficult to integrate information and identify interesting parts
  - Laborious to derive interesting (aggregated) information

Motivation

- 1 - Information exploration
  - **How to explore web archives in a more advanced and exploratory way?**
  - Find articles of a specific time period, discussing about a specific category of entities, or about entities sharing some characteristics

- 2 - Information integration
  - **How to explore web archives by also integrating information from existing knowledge bases (e.g., DBpedia)?**
  - How to integrate information coming form multiple (web) archives?

Motivation

- 3 - Information inference / knowledge discovery
  - **How to infer knowledge by exploiting the contents of a Web Archive?**
  - Identify important time periods related to one or more entities
  - Find out popular entities of a specific type in specific time periods

- 4 - Robustness in information change
  - **How to explore web archives by automatically taking into account the change of entities over time?**
  - Find documents without worrying about their correct reference

Motivation

- 5 - Multilinguality
  - **How to explore documents about entities independently of the document language** (and thus of the language of the entity mentions)?

- 6 - Interoperability
  - **How to facilitate exploration of web archives by other systems and tools?**
  - Expose information about web archives in the Web, in a standard and machine understandable format
  - Identify interesting parts for further analysis easily and fast

Existing Approaches

- **Exploring** Web Archives
  - Search services provided by Internet Archive (Wayback Machine), Memento (Time Travel), Archive-It, Portuguese Web Archive
  - Research works: [Holzmann and Anand, 2016], [Kanhabua et al., 2016], [Vo et al., 2016], [Jackson et al., 2016], [Singh et al., 2016]

- **Profiling** Web Archives
  - Improve effectiveness of query routing strategies in distributed archive search [AlSum et al., 2014], [Alam et al., 2015], [Bornand et al., 2016], [Alam et al., 2016]

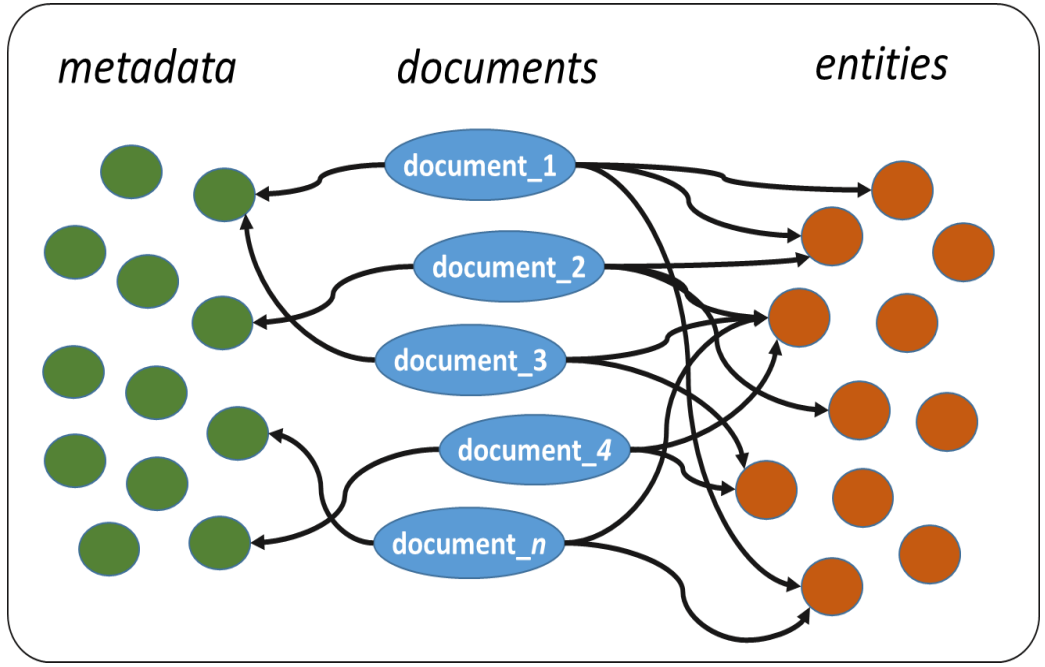
- **Analyzing** Web Archives
  - Frameworks for distributed analysis of Web Archives
  - ArchiveSpark [Holzmann et al., 2016], Warcbase [Lin et al., 2014]

Our approach: building and querying **Semantic Layers**

*for **profiling** and **exploring** web archives*

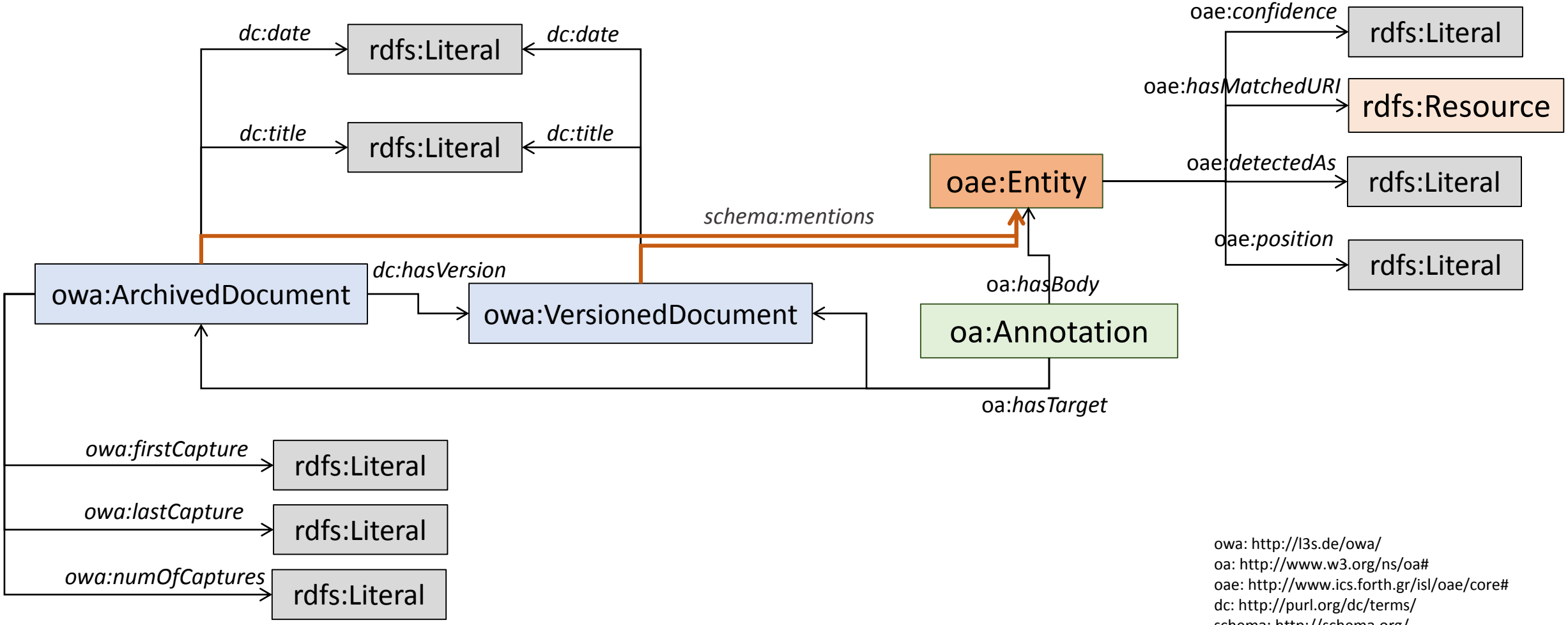
- Semantic Layer:
  - An **RDF** repository of **structured data** (RDF triples) about an archived collection of documents
- It allows:
  - Describing useful **metadata** information about the archived documents
  - **Annotating** the documents with semantic information, like entities, concepts and events mentioned in the documents
  - **Publishing** all this data on the Web as Linked Data
- Why?
  - Advanced, entity-centric query capabilities (using SPARQL)
  - Real-time data integration
  - Directly accessible and exploitable by other systems and tools
- Next step: development of user-friendly services on top of semantic layers

Semantic Layers



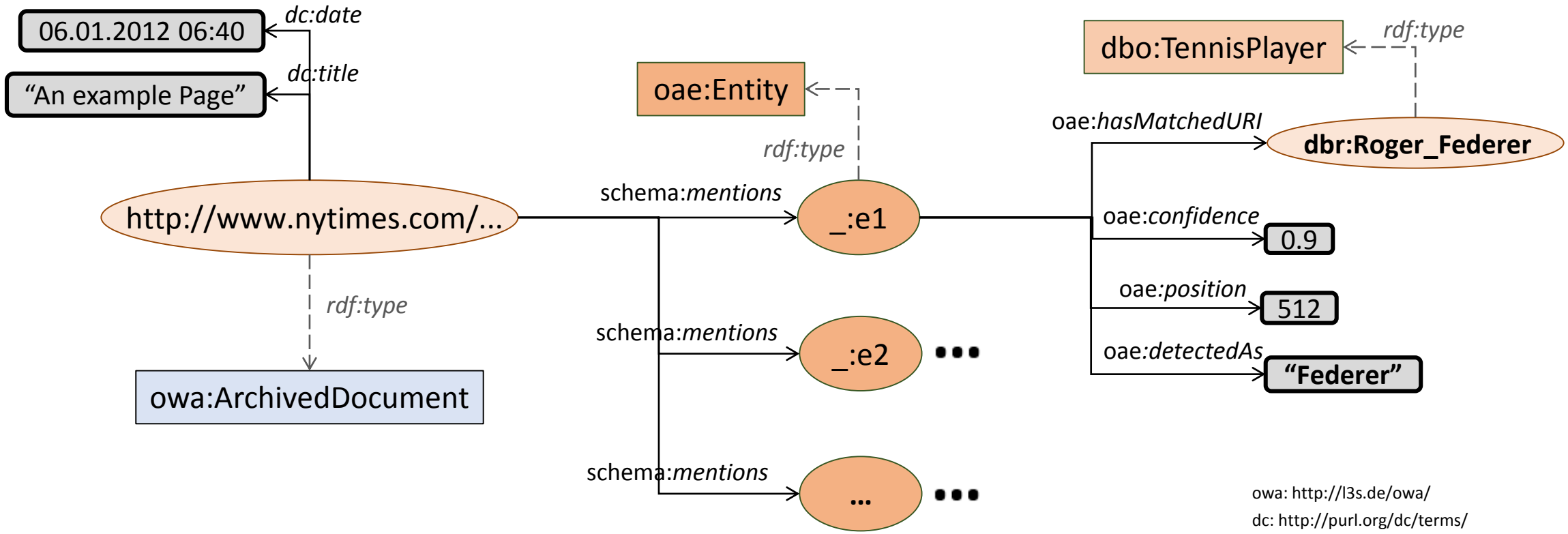
- Building Semantic Layers
  - RDF/S **data model**: “Open Web Archive”
  - Construction **process**
  - Open source **framework**: “ArchiveSpark2Triples”
- Case Studies and Query Capabilities
  - Semantic Layer over a **Web Archive** (versioned)
  - Semantic Layer over a **News Archive** (non-versioned)
  - Semantic Layer over a **Social Media Archive**
- Evaluation
- Problems and Limitations

RDF/S data model: *Open Web Archive* http://l3s.de/owa/



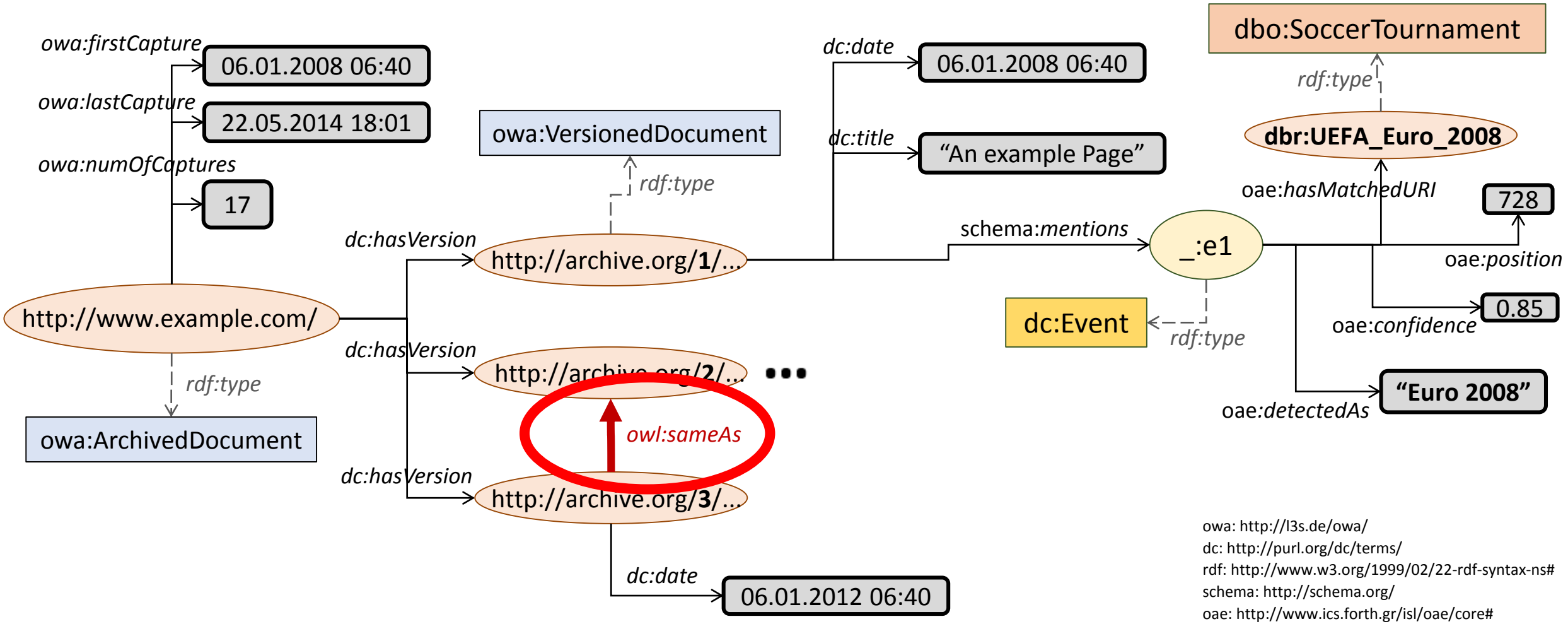
owa: http://l3s.de/owa/
oa: http://www.w3.org/ns/oa#
oae: http://www.ics.forth.gr/isl/oae/core#
dc: http://purl.org/dc/terms/
schema: http://schema.org/
rdfs: http://www.w3.org/2000/01/rdf-schema#

Open Web Archive – Example of Non-versioned Web Page



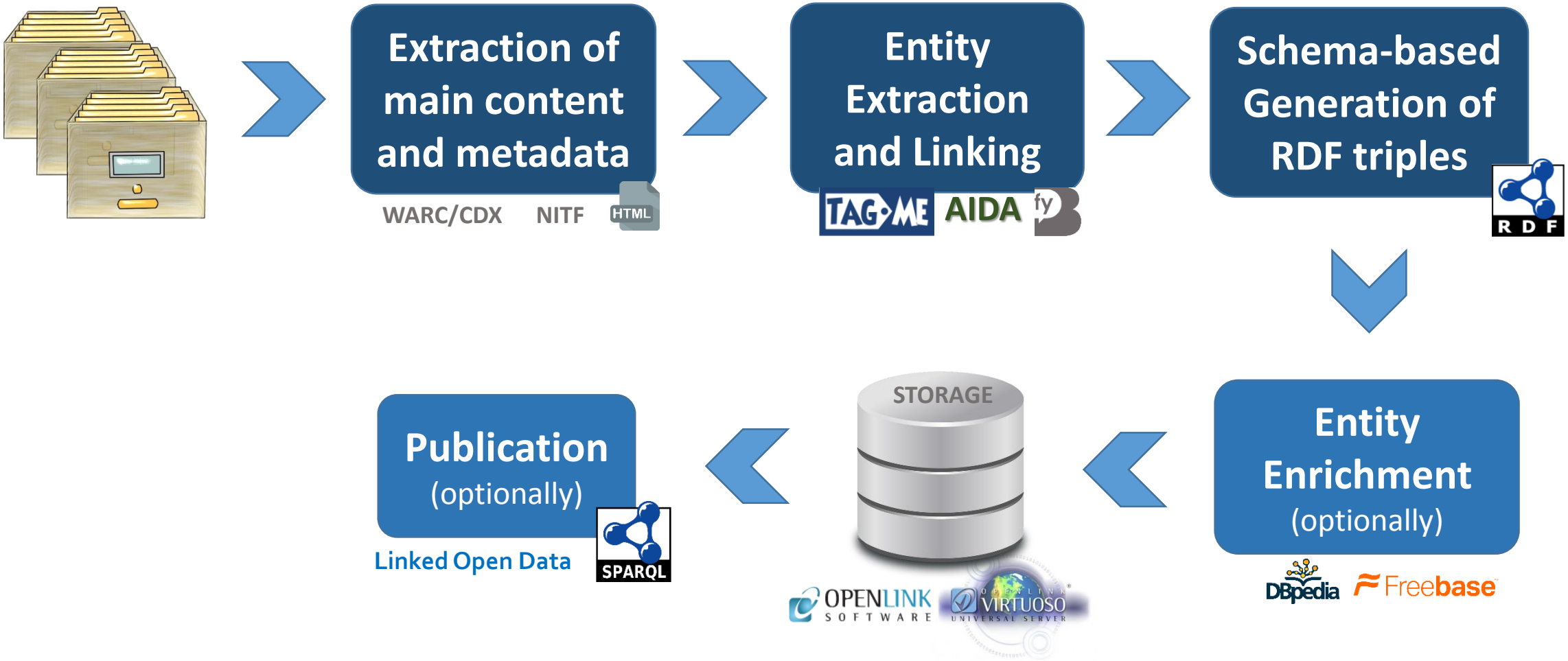
owa: http://l3s.de/owa/
dc: http://purl.org/dc/terms/
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
schema: http://schema.org/
oae: http://www.ics.forth.gr/isl/oae/core#

Open Web Archive – Example of Versioned Web Page



owa: http://l3s.de/owa/
dc: http://purl.org/dc/terms/
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#
schema: http://schema.org/
oae: http://www.ics.forth.gr/isl/oae/core#
owl:http://www.w3.org/2002/07/owl#

The construction process



Apache Spark framework: ***ArchiveSpark2Triples***

https://github.com/helgeho/ArchiveSpark2Triples

- Based on **ArchiveSpark** framework https://github.com/helgeho/ArchiveSpark
  - Programming framework for efficiently analyzing large web archives stored in the standard **WARC/CDX** format
  - Unified data model storing records in an hierarchical way
  - Very fast filtering, grouping and sorting based on metadata
  - Support of external modules, called **enrich functions**
- **ArchiveSpark2Triples**
  - Extension that automates the construction of a semantic layer
  - Output: Notation3 (N3) files
  - Customizable assignment of URLs and vocabularies to use → Extendable!
  - Extraction of entities using **Yahoo FEL** entity linking tool
    - *Enrich function available under **FEL4ArchiveSpark*** https://github.com/helgeho/FEL4ArchiveSpark

Apache Spark framework: ***ArchiveSpark2Triples***

https://github.com/helgeho/ArchiveSpark2Triples

- **Efficiency**
  - Very efficient for operations that only rely on metadata information (in CDX files)
    - All properties of archived documents, majority of properties of versioned documents
  - Actual contents are accessed only for applying **enrich functions** to the versioned documents that do **not** constitute duplicates of older versions
  - **Entity extraction** is the most expensive task
  - Actual time for the entire workflow depends on:
    - Dataset size and nature of data
    - Computing infrastructure and available resources
    - Indicatively: 24 hours for creating a semantic layer for a web archive of 9 million web pages (474.6 GB of compressed WARC and CDX files)
      - Hadoop cluster of 25 compute nodes, 268 CPU cores, 2,688 GB RAM, 110 executors in parallel most of the time

Case Studies

Available at: http://l3s.de/owa/semanticlayers/

- **Web Archive** (versioned)
  - **Occupy Movement** 2011/2012 collection (generously provided by **Archive-It**)
  - ≈ 9M captures of ≈ 3M URLs
  - **URLs for versions:** links to the collection's Wayback Machine provided by ArchiveIt
  - >10B triples, >1.3M **same-as** properties, 939,960 distinct entities
- **News Archive** (non-versioned)
  - New York Times Annotated Corpus
  - ≈ 1.5M articles published by NYT between 1987 and 2007
  - >195M triples, 856,283 distinct entities
- **Social Media Archive**
  - ≈ 1.4M tweets posted in 2016 by 469 twitter accounts of USA newspapers
  - Metadata: *creation date, username, favorite count, retweet count*
  - >19 million triples, 146,854 distinct entities

Case Studies – Query Capabilities

- Information Exploration and Integration
  - Articles of **summer 1989** mentioning **New York lawyers born in Brooklyn** (and for each lawyer show its **birth date** and a **description in French**)

```
SELECT DISTINCT ?article ?title ?date ?nylawyer ?birthDate ?abstr WHERE {
  SERVICE <http://dbpedia.org/sparql> {
    ?nylawyer dc:subject dbc:New_York_lawyers ;
              dbo:birthPlace dbr:Brooklyn .
    OPTIONAL {
       ?nylawyer dbo:birthDate ?birthDate ;
                 dbo:abstract ?abstr FILTER(lang(?abstr)='fr') } }

  ?article dc:date ?date FILTER(?date >= "1989-06-01"^^xsd:date &&
                                ?date <= "1989-08-31"^^xsd:date)
  ?article schema:mentions ?entity .
  ?entity oae:hasMatchedURI  ?nylawyer .
  ?article dc:title ?title } ORDER BY ?nylawyer
```

44 lawyers

Semantic Layer over NYT articles

Result: 184 articles

Case Studies – Query Capabilities

- Information Exploration and Integration
  - **Popular tweets** (with >50 re-tweets) posted during the **summer of 2016**, mentioning **basketball players** of the NBA team **Los Angeles Lakers**

Semantic Layer over tweets collection

```
SELECT DISTINCT ?tweet ?count ?date ?entityUri WHERE {
  SERVICE <http://dbpedia.org/sparql> {
     ?entityUri dc:subject dbc:Los_Angeles_Lakers_players }
  ?t a tw:Tweet ; dc:date ?date FILTER(?date>="2016-06-01"^^xsd:dateTime &&
                                       ?date<="2016-08-31"^^xsd:dateTime)
  ?t tw:retweetCount ?count FILTER (?count > 50) .
  ?t schema:text ?tweet ; schema:mentions ?entity .
  ?entity oae:hasMatchedURI ?entityUri }
```

7 players

Result: 14 tweets

Case Studies – Query Capabilities

- Information Inference / Knowledge Discovery
  - Most discussed **journalists** in **Occupy Movement** collection

Semantic Layer
over Occupy Movement

```
SELECT ?journ (COUNT(DISTINCT ?page) AS ?num) WHERE {
  SERVICE <http://dbpedia.org/sparql> {
    ?journ a yago:Journalist110224578 }
  ?page a owa:ArchivedDocument ; dc:hasVersion ?version .
  ?version schema:mentions ?entity .
  ?entity oae:hasMatchedURI  ?journ
} GROUP BY ?journ ORDER BY DESC(?num)
```

- Ralph Nader
- Chris Hedges
- Dylan Ratigan

Case Studies – Query Capabilities

- Information Inference / Knowledge Discovery
  - Number of articles per year mentioning **Nelson Mandela**

```
SELECT ?year (COUNT(DISTINCT ?article) AS ?num) WHERE {
  ?article dc:date ?date ; schema:mentions ?entity .
  ?entity oae:hasMatchedURI dbr:Nelson_Mandela
} GROUP BY (year(?date) AS ?year) order by ?year
```

Semantic Layer over NYT articles

1988	54
1989	80
1990	509
1991	193
1992	142
1993	157
1994	282
1995	172
1996	130

released from prison

become president
(South African multiracial general election)

Case Studies – Query Capabilities

- Information Inference / Knowledge Discovery
  - Most discussed **Drugs** in **1987 (left) and 1997 (right)**

Semantic Layer over NYT articles

```
SELECT DISTINCT ?drug (count(DISTINCT ?article) as ?numOfArticles) WHERE {
  SERVICE <http://dbpedia.org/sparql> { ?drug a dbo:Drug }
  ?article dc:date ?date FILTER(year(?date) = "1987") .
  ?article schema:mentions ?entity . ?entity oae:hasMatchedURI  ?drug .
} GROUP BY ?drug ORDER BY DESC(?numOfArticles)
```

Drug	Num of articles (1987)
http://dbpedia.org/resource/Cocaine	778
http://dbpedia.org/resource/Heroin	248
http://dbpedia.org/resource/Aspirin	63
http://dbpedia.org/resource/Zidovudine	53
http://dbpedia.org/resource/Furosemide	53

Drug	Num of articles (1997)
http://dbpedia.org/resource/Cocaine	462
http://dbpedia.org/resource/Heroin	275
http://dbpedia.org/resource/**Nicotine**	125
http://dbpedia.org/resource/**Fluoxetine**	61
http://dbpedia.org/resource/**Caffeine**	58

Case Studies – Query Capabilities

- Robustness and Multilinguality
  - Extracted entities are assigned unique URIs
    - Different mentions of an entity are assigned the same unique URI
      - e.g., name variants or names in different languages
  - For multilinguality, the entity linking system should support the identification of entities in different languages
  - Time-awareness and correct disambiguation of the entity linking system can affect the results!

Evaluation

- Objectives:
  - to show that for a bit more complex information needs, keyword-based search systems return poor results
    - Thus, calling for new, more advanced information seeking strategies!
  - to identify possible problems and limitations of our approach

- Setup
  - Archived collection: **NYT corpus**
  - 20 information needs of *exploratory nature*
    - each one requesting documents of a **specific time period**, related to some **entities of interest**
  - Each information need corresponds to **one SPARQL query** and **one free-text query**
  - Example of information need:
    - *"find articles of **June 2010** discussing about **African-American film producers**"*
    - Corresponding free text query: *"African-American film producers"* (we manually specify the data range to each system)

Evaluation

- Comparison:
  - SPARQL query on **Semantic Layer**
  - Free-text query on **Google News** (appending the string “ `site:nytimes.com`”)
  - Free-text query on **HistDiv** [Singh et al., 2016]
    - Time-aware and diversity-oriented approach
- Manual evaluation of all returned results
  - Considering only articles existing in all systems!

Evaluation – Results

Disambiguation error

	Information need	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
SPARQL	Num of results	27	34	37	16	11	14	**18**	8	**11**	15	**15**	12	13	16	15	12	15	13	16	15
	Num of **relevant** results	27	27	33	16	9	14	**2**	8	**1**	14	**1**	8	13	15	9	10	13	11	15	15
GOOGLE NEWS	Num of results	8	1	0	0	0	1	1	1	0	0	0	0	0	2	0	6	1	1	1	1
	Num of **relevant** results returned by SPARQL	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0
	Num of **relevant** results **not returned by SPARQL**	0	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**
HISTDIV	Num of results	0	3	1	0	0	0	0	4	0	0	0	0	0	0	0	25	2	0	0	0
	Num of **relevant** results returned by SPARQL	0	**2**	0	0	0	0	0	0	0	0	0	0	0	0	0	**3**	**1**	0	0	0
	Num of **relevant** results **not returned by SPARQL**	0	**1**	0	0	0	0	0	**3**	0	0	0	0	0	0	0	**3**	0	0	0	0

Information needs and full results available at: http://l3s.de/owa/semanticlayers/SemLayerEval.zip

Problems and Limitations

- False positive:
  - A returned document is not relevant due to disambiguation error
- False negative
  - A relevant document is not returned because:
    - An entity of interest was not recognized by the entity linking tool
    - Disambiguation error
    - Confidence score of extracted entity of interest below used threshold
- Temporal inconsistency
  - Change of entity properties
  - Completeness and freshness of used knowledge bases

Efficiency of Query Answering

- Query execution time depends on:
  - Efficiency of triplestore and server
  - Query itself
    - Use of costly operators (like FILTER, OPTIONAL and SERVICE)
- Indicatively:
  - ≈400 ms is the average execution time of the 20 queries used in our evaluation
    - Min: 56 ms, max: 2.4 sec
    - They all make use of SERVICE operator for querying DBpedia
    - Experiments on Openlink Virtuoso server installed in a modest personal computer (Intel Core i5, 8GB RAM)

Conclusions

- **Data model** and **framework** for constructing Semantic Layers for (web) archives
- Semantic Layers allow:
  - ❖ **Exploring** web archives in more advanced and exploratory ways (entity-centric)
  - ❖ **Integrating** information (at query-execution time) coming from other semantic layers and knowledge bases
  - ❖ **Inferring** new knowledge that is very laborious and time-consuming to derive otherwise (just with one query)
  - ❖ Coping with common problems like **temporal reference variants** and **multilinguality**
  - ❖ Making the contents of web archives **machine understandable**
- Vision:
  - Enrich the **LOD cloud** with semantic layers of web archives (e.g., *ArchiveIt* collections)

Future Work

- Development of **user-friendly interfaces** on top of Semantic Layers
  - Faceted Search and Exploration
  - Translation of free-text queries to SPARQL
- Ranking of SPARQL results
  - Since all results equally match the query
- Cope with temporal inconsistencies
  - Use entity URIs that lead to old DBpedia descriptions?

Thank you

Comments/Questions?

ALEXANDRIA Project (ERC Nr. 339233)