LOD is all about evolution

Querying and Managing evolving Linked Open Data

Javier D. Fernández

Drift-a-LOD'17

11TH SEPTEMBER 2017

Special thanks to Axel Polleres for his input

About me:

Valladolid

Pablo de la Fuente
Miguel A. Martínez-Prieto

Santiago

Claudio Gutiérrez

Madrid

Óscar Corcho

Rome

Maurizio Lenzerini

Vienna

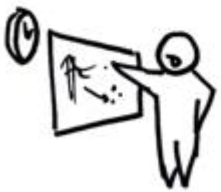
Axel Polleres

- since 2015 @WU, Inst. for Information Business
  - Research interest: Semantic Web, Open Data, Big (Semantic) Data Management, Databases, Data Compression, Privacy and Security

- https://www.wu.ac.at/en/infobiz/team/fernandez/

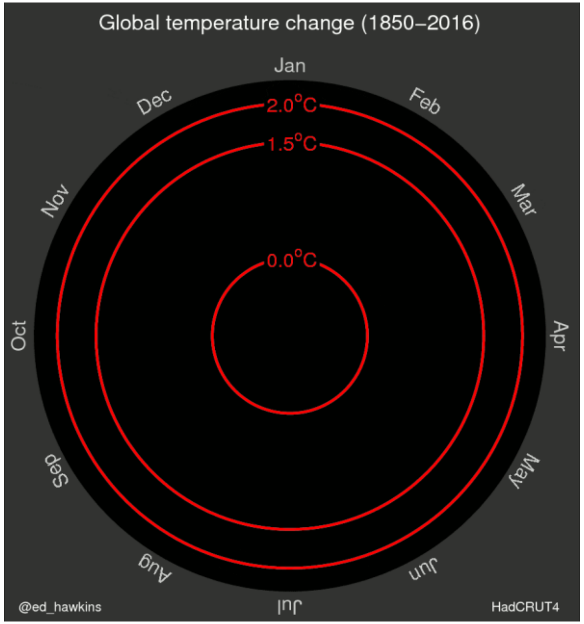
General agenda

- **Monitoring Evolution and Archiving**
- ***Archiving the Web of Data***
- **Representing and querying evolving semantic data**
- **Open Data evolution**

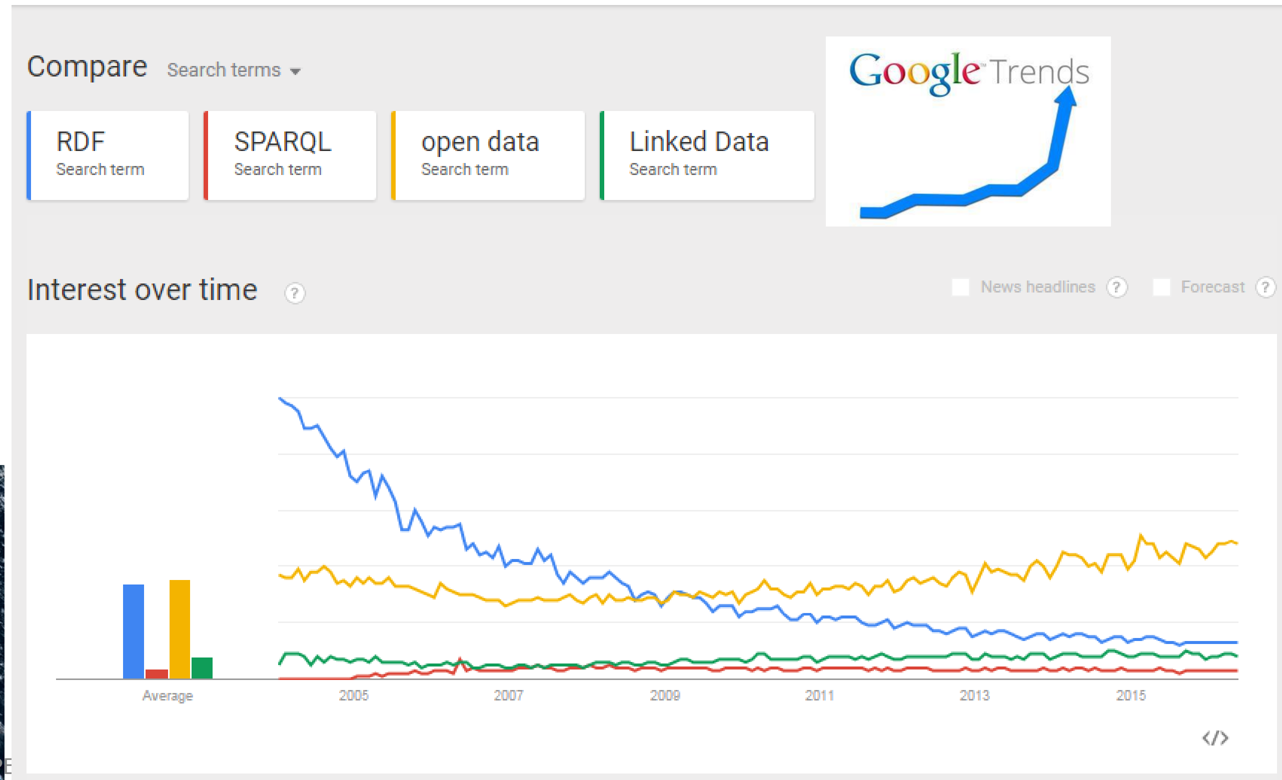


images: zurb.com

Why evolution matters (Creationists: please ignore this slide…)

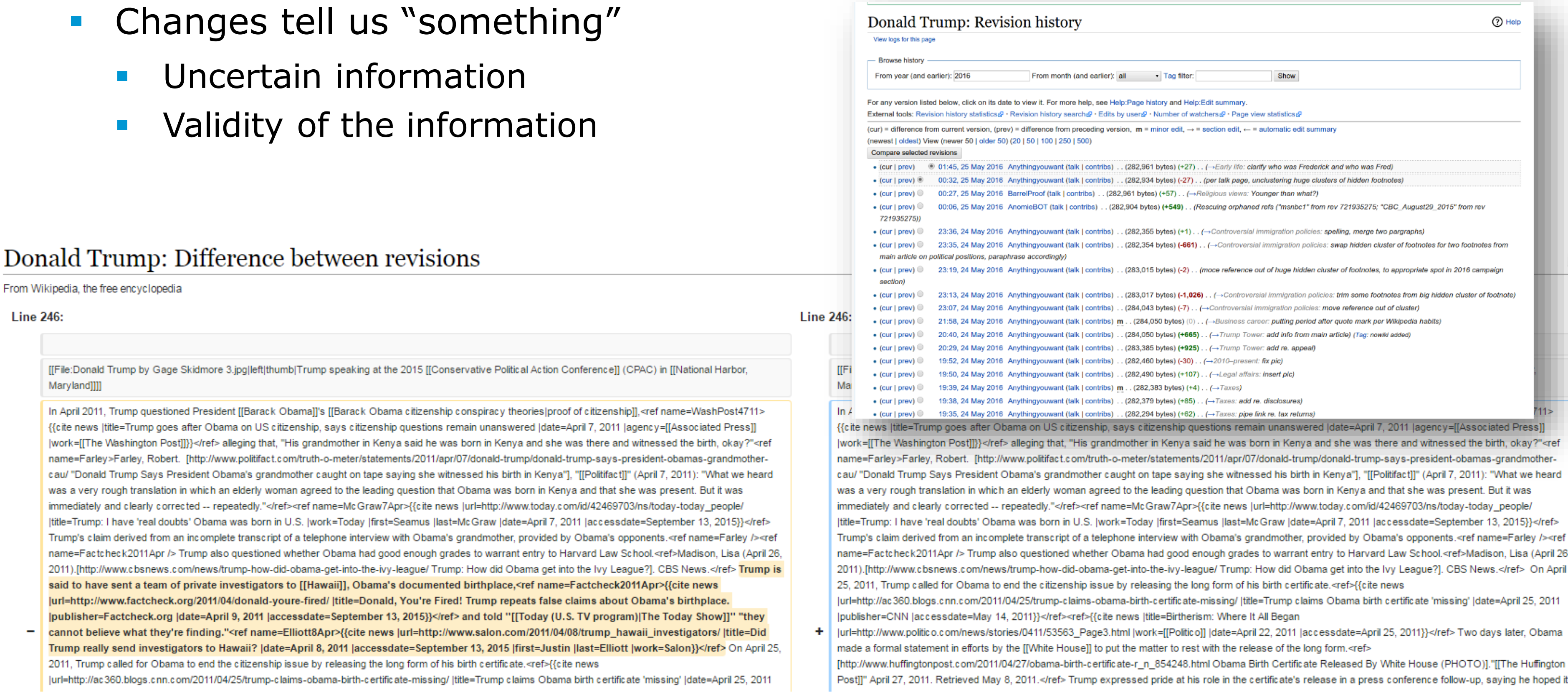


- Monitoring evolution is relevant



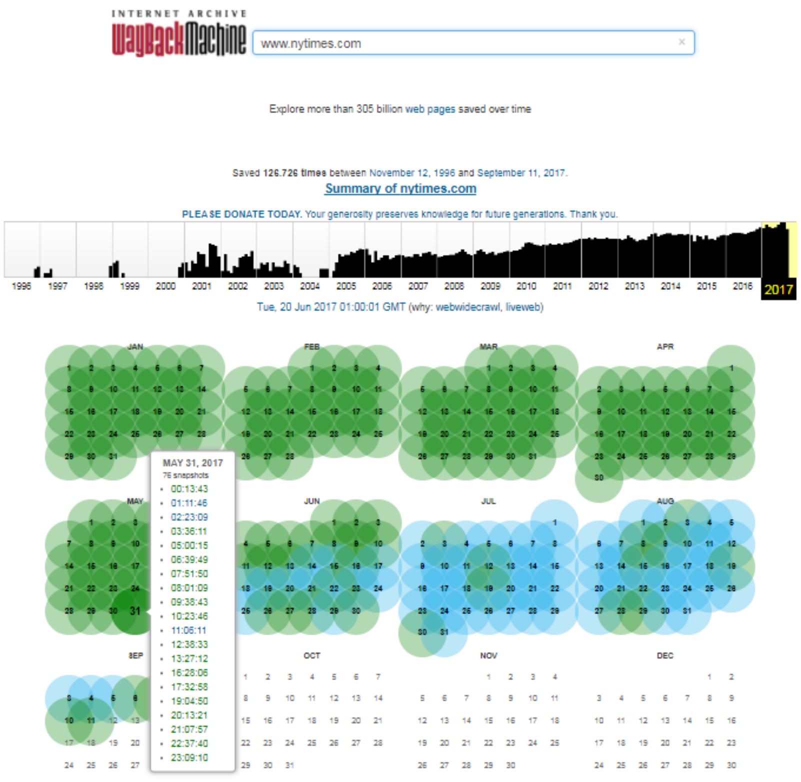
Evolution matters

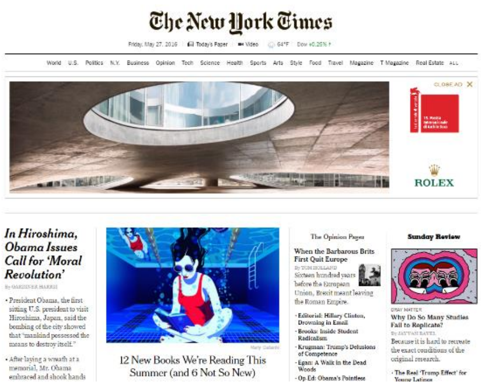
- Changes tell us "something"
  - Uncertain information
  - Validity of the information



Preservation matters

- Web archives: Common Crawl, Internet Memory, Internet Archive, …



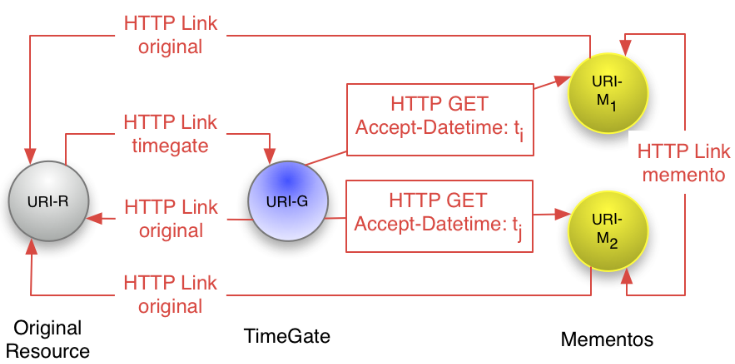


Time-based access matters

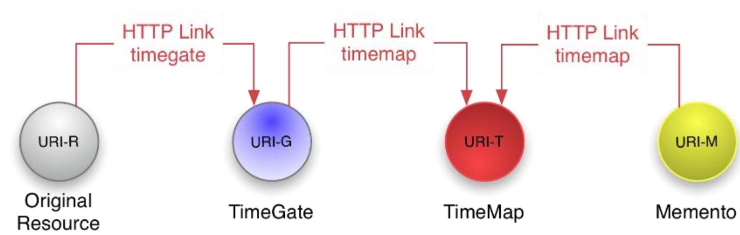
- The Memento protocol

  RFC 7089

Follow your nose
(HTTP content negotiation with datetime)



Batch discovery
(list of URIs of Mementos of the Original Resource)



But…

Challenges (Web archives)

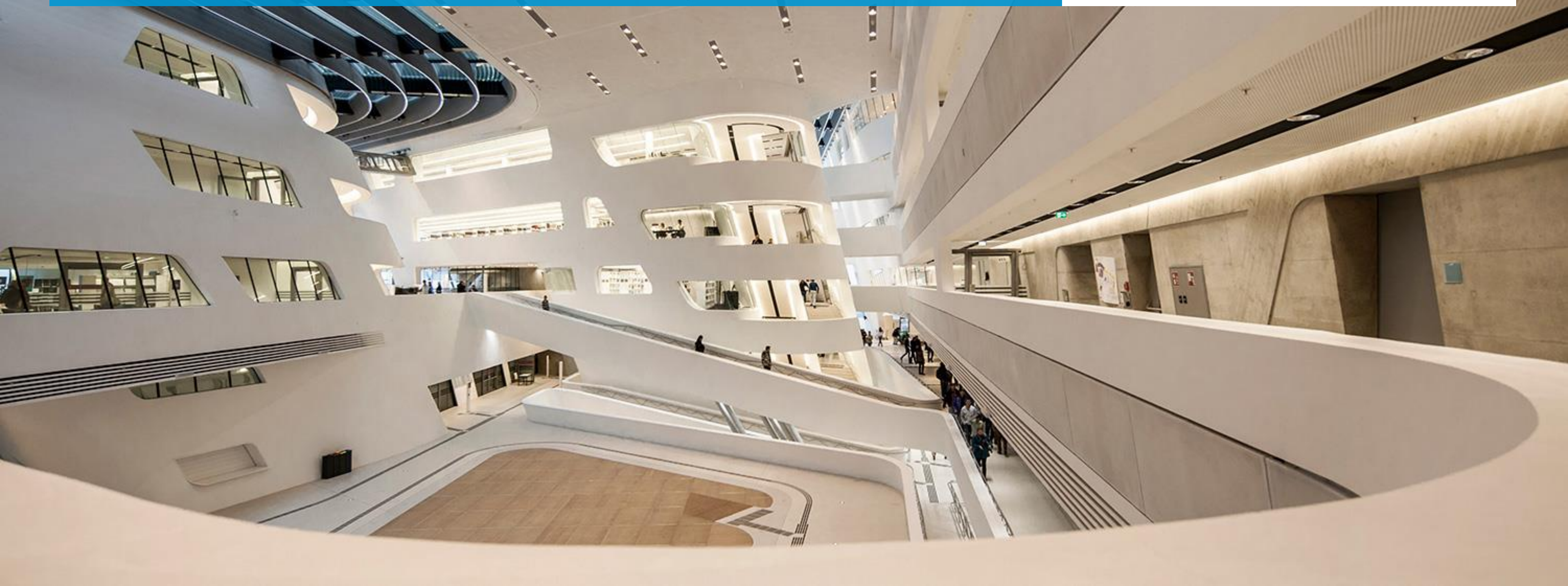
- **Poor granularity** (“some” snapshots)
- **Aggregated data**, only, rather than raw data access
  - (e.g. in Google trends)
- What is the right query language?
  - basic retrieval features (get version at timestamp t)
  - when did a certain information disappear?
  - when was it changed?
  - structured queries?
- Scalability problems

Is it easier/better for RDF/Linked Data?

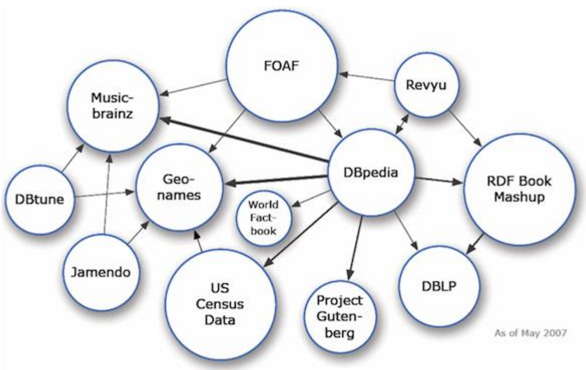
Arching the Web of Data

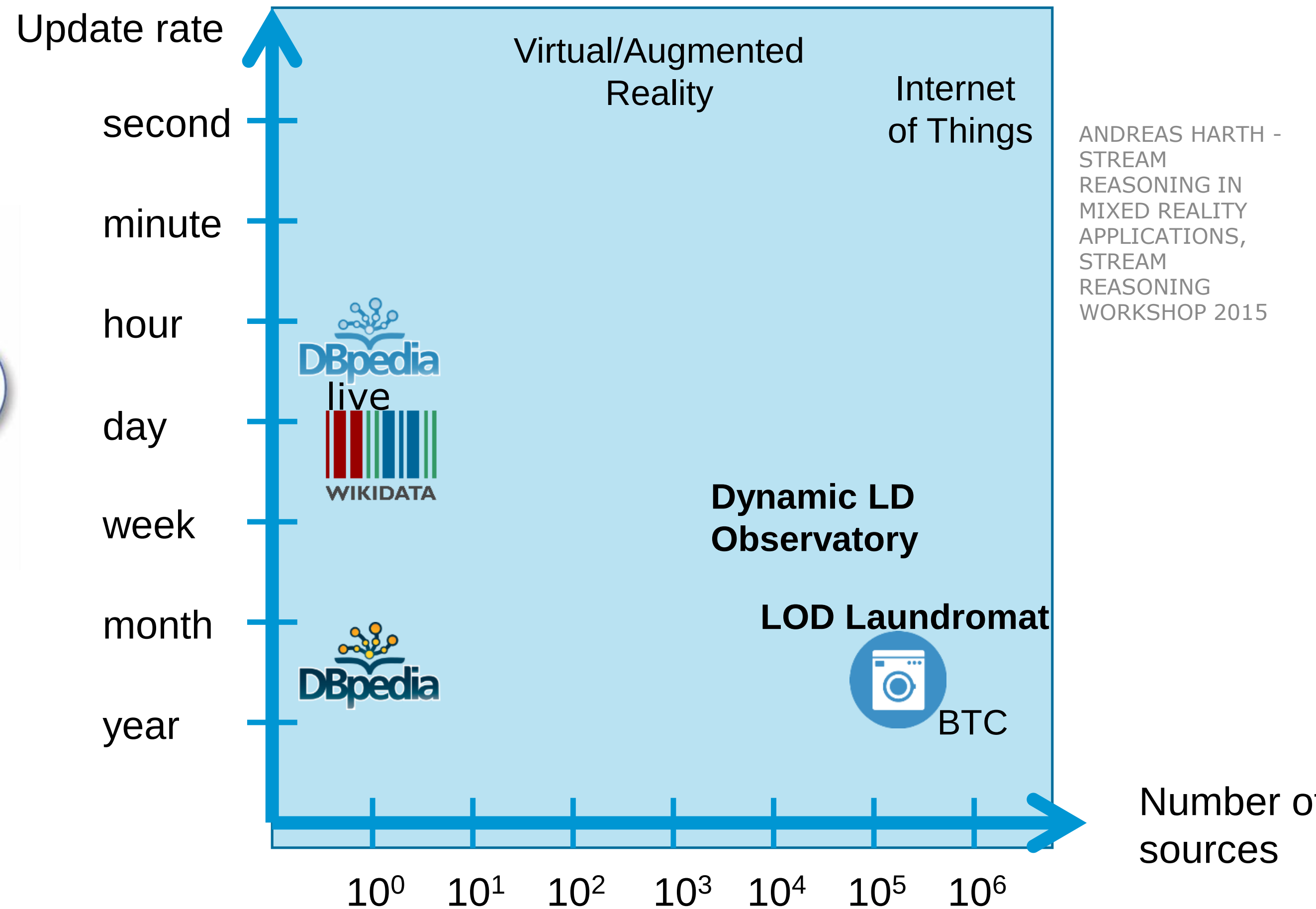
WIRTSCHAFTS UNIVERSITÄT WIEN VIENNA UNIVERSITY OF ECONOMICS AND BUSINESS

COMPLEXITY SCIENCE HUB VIENNA



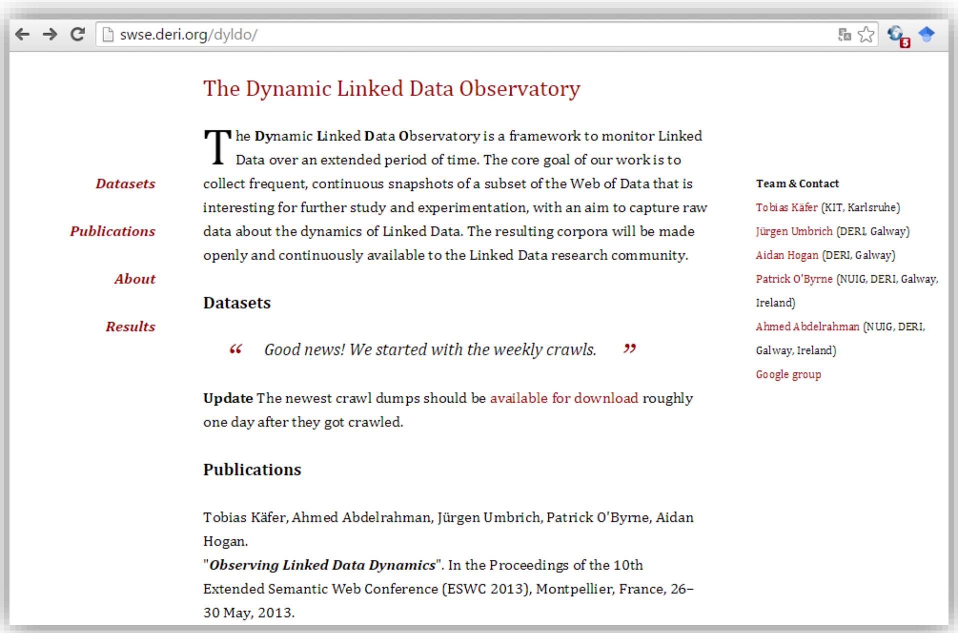
Linked Data is evolving

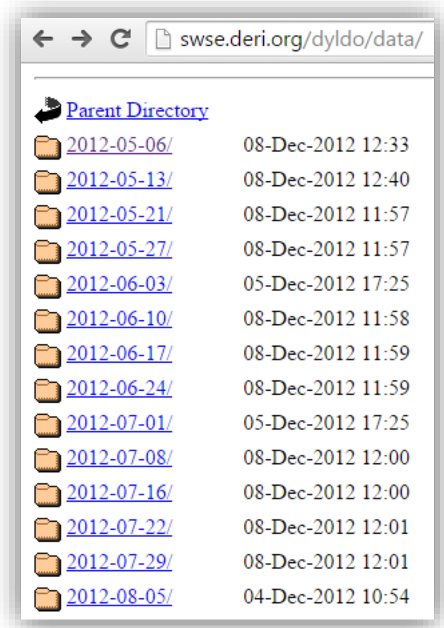




ANDREAS HARTH - STREAM REASONING IN MIXED REALITY APPLICATIONS, STREAM REASONING WORKSHOP 2015

One of the first (and last?) LOD archives: The Dynamic Linked Data Observatory (evolving Linked Data since 2012)

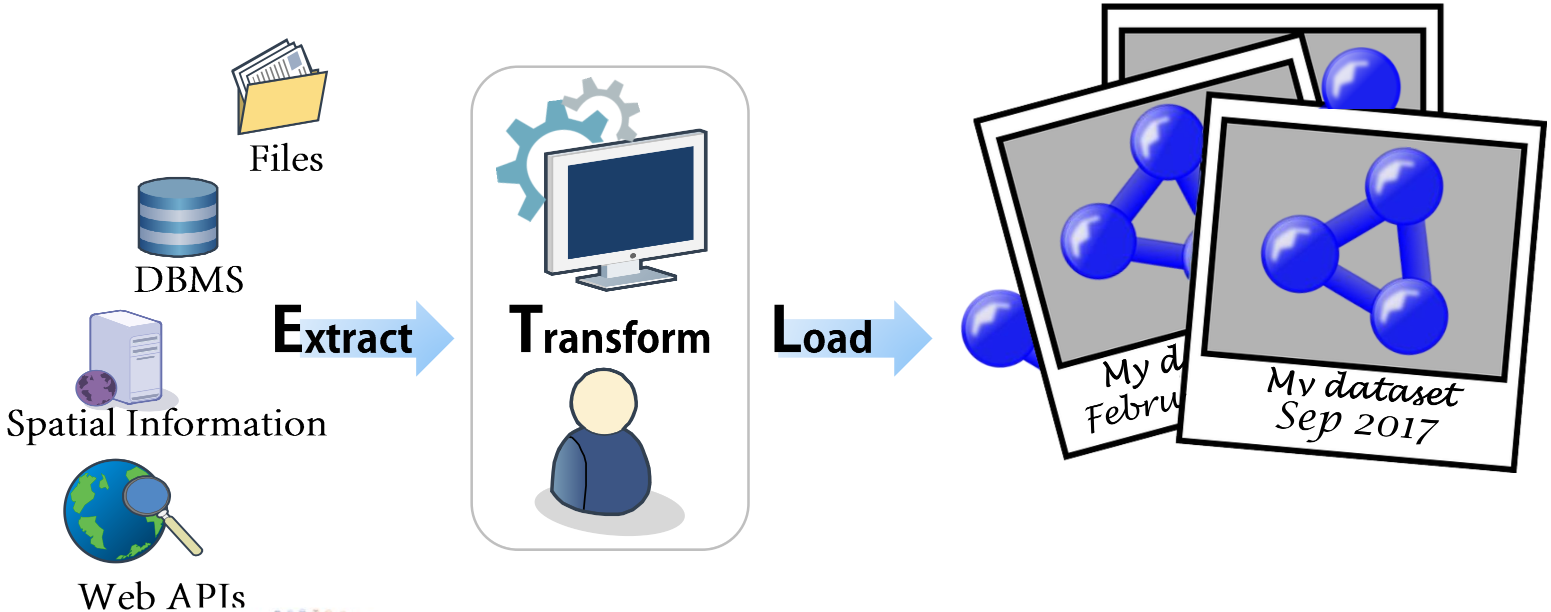


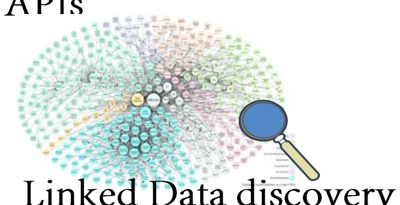


Weekly dumps of crawl snapshots...

Granularity?
Queries?
Crawl failures?

Linked Data Archives:
The missing link in the RDF evolution

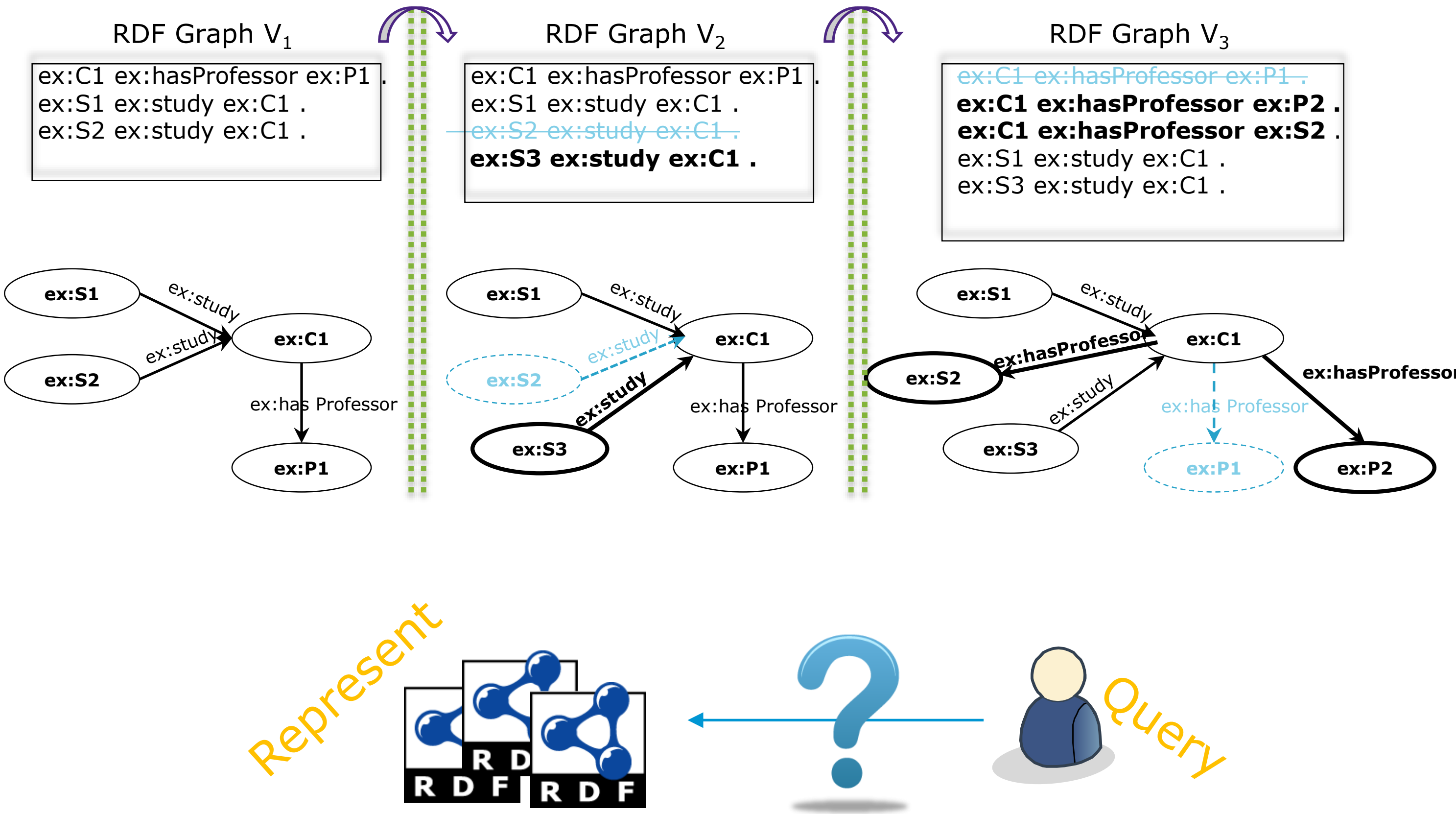




Most semantic Web/Linked Data tools are focused on this "**static** view" **but do not consider versioning/evolution**

Sindice, SWSE, Swoogle, LOD Cache, LOD-Laundromat… so far, no versions!

RDF Archiving. Example



Research challenges on evolving structured interlinked data

- How can we **represent** archives of continuously evolving linked datasets?
- How can we **minimize** the redundant information of archives? (e.g. duplicates in snapshots)
- How can we improve **completeness** of archiving?
- How can emerging **retrieval** demands in archiving be satisfied?
  - e.g. time-traversing and traceability? Avoiding bottlenecks?
- How can certain **time-specific queries** over archives be answered?
  - Can we re-use existing technologies (e.g. SPARQL or temporal extensions)?
  - What is the right query language for such queries?
  - e.g. knowing if a dataset has changed, and how, in a certain time period?

…in the last few years:

DIACHRON
Managing the Evolution and Preservation of the Data Web (FP7)

PRELIDA
Preserving Linked Data (FP7)

Research projects

The Dynamic Linked Data Observatory

DBpedia WaybackMachine

Archives

V-RDFCSA

DYDRA

Apache Marmotta

Stardog

LEDS

memento

#LD Linked Data Fragments

Tools

HOBBIT

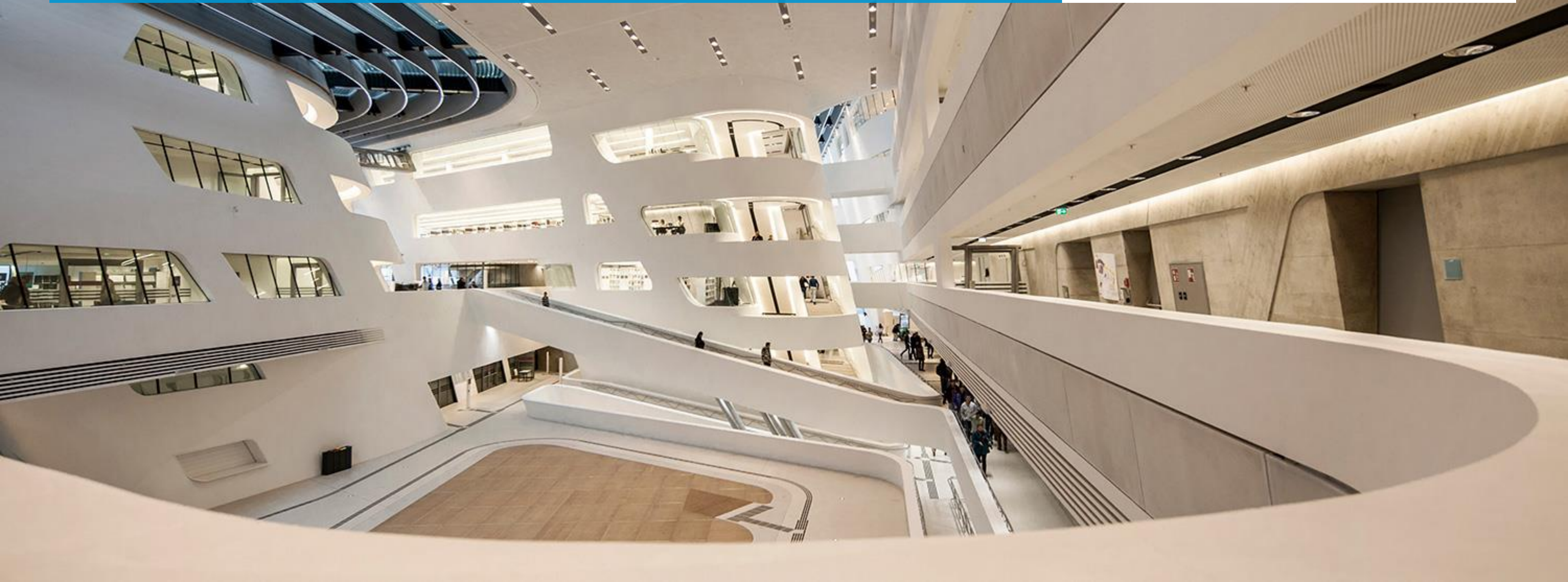
BEAR

Benchmarking

Representing and querying evolving semantic data

WU WIRTSCHAFTS UNIVERSITÄT WIEN VIENNA UNIVERSITY OF ECONOMICS AND BUSINESS

COMPLEXITY SCIENCE HUB VIENNA

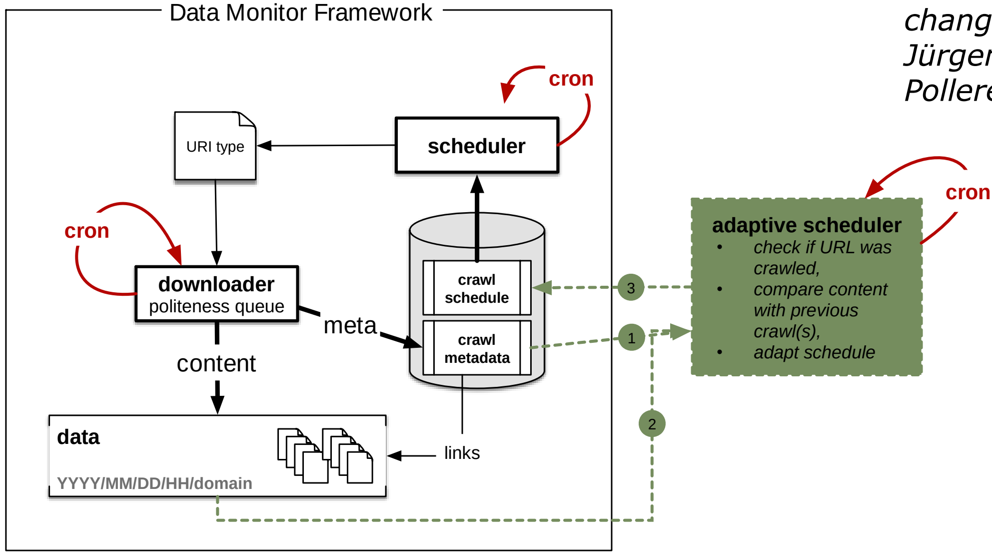


The cold-start problem

How we can get archives of RDF data

Pull changes (crawl) vs. Push changes (notify)

- Some services that publish or are mapped to RDF change regularly, but we don't know the frequency upfront!



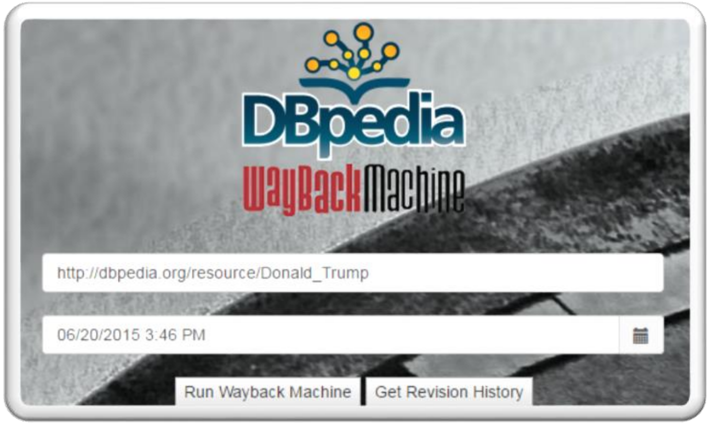
Towards capturing and preserving changes on the Web of Data 50-65 Jürgen Umbrich, Nina Mrzelj, Axel Polleres. DIACHRON WS 2015

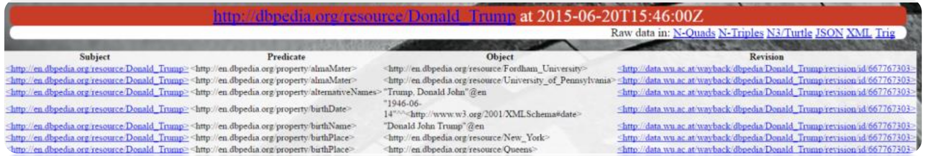
- Some services mapped to RDF announce/archive their changes already, so they already keep an archive…

DBpedia Wayback Machine

- Retrieve historical versions of a DBpedia resource
  - What was the version of "Donald Trump" on dd/mm/yyyy?
- Re-apply DBpedia mappings on the Wikipedia revision history

http://data.wu.ac.at/wayback

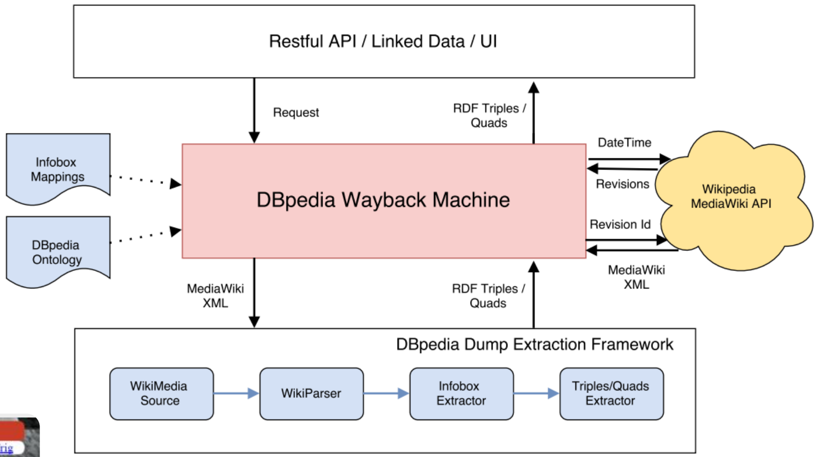




http://dbpedia.org/resource/Donald_Trump at 2015-06-20T15:46:00Z

Raw data in: N-Quads N-Triples N3/Turtle JSON XML Trig

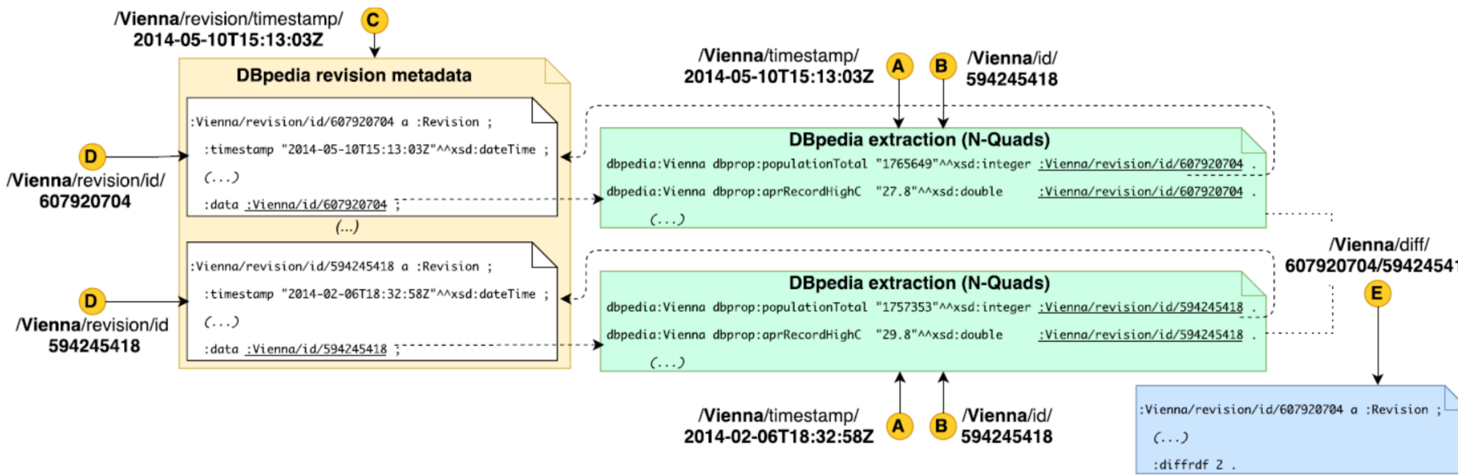
Subject	Predicate	Object	Revision
<http://en.dbpedia.org/resource/Donald_Trump>	<http://en.dbpedia.org/property/almaMater>	<http://en.dbpedia.org/resource/Fordham_University>	<http://data.wu.ac.at/wayback/dbpedia/Donald_Trump/revision/id/667767303>
<http://en.dbpedia.org/resource/Donald_Trump>	<http://en.dbpedia.org/property/almaMater>	<http://en.dbpedia.org/resource/University_of_Pennsylvania>	<http://data.wu.ac.at/wayback/dbpedia/Donald_Trump/revision/id/667767303>
<http://en.dbpedia.org/resource/Donald_Trump>	<http://en.dbpedia.org/property/alternativeNames>	"Trump, Donald John"@en	<http://data.wu.ac.at/wayback/dbpedia/Donald_Trump/revision/id/667767303>
<http://en.dbpedia.org/resource/Donald_Trump>	<http://en.dbpedia.org/property/birthDate>	"1946-06-14"^^<http://www.w3.org/2001/XMLSchema#date>	<http://data.wu.ac.at/wayback/dbpedia/Donald_Trump/revision/id/667767303>
<http://en.dbpedia.org/resource/Donald_Trump>	<http://en.dbpedia.org/property/birthName>	"Donald John Trump"@en	<http://data.wu.ac.at/wayback/dbpedia/Donald_Trump/revision/id/667767303>
<http://en.dbpedia.org/resource/Donald_Trump>	<http://en.dbpedia.org/property/birthPlace>	<http://en.dbpedia.org/resource/New_York>	<http://data.wu.ac.at/wayback/dbpedia/Donald_Trump/revision/id/667767303>
<http://en.dbpedia.org/resource/Donald_Trump>	<http://en.dbpedia.org/property/birthPlace>	<http://en.dbpedia.org/resource/Queens>	<http://data.wu.ac.at/wayback/dbpedia/Donald_Trump/revision/id/667767303>



On-demand "archive"

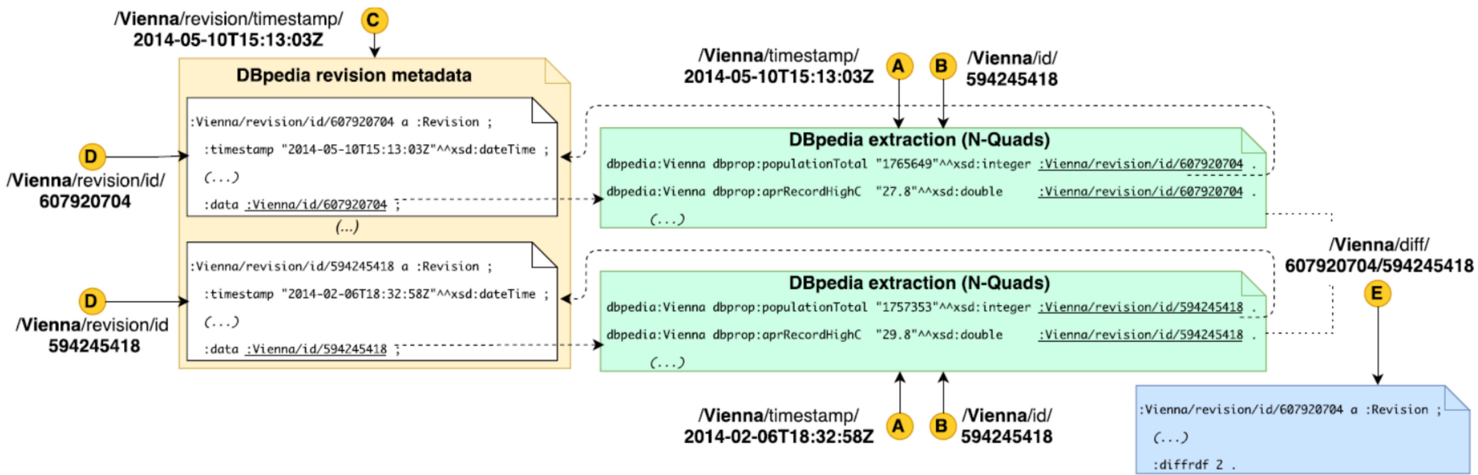
DBpedia Wayback Machine

- How can one represent revisions while respecting DBpedia?
  - a) quads → <dbpediaSubject> <pred> <obj> <Revision> .
  - b) proprietary triples → <ownSubject/Revision> <pred> <obj> .
- Operations?
  - Get revisions meta-data for one resource (by revisionID or timestamp)
  - Get "materialised" versions of a resource (by revisionID or timestamp)
  - Get difference between two revisions

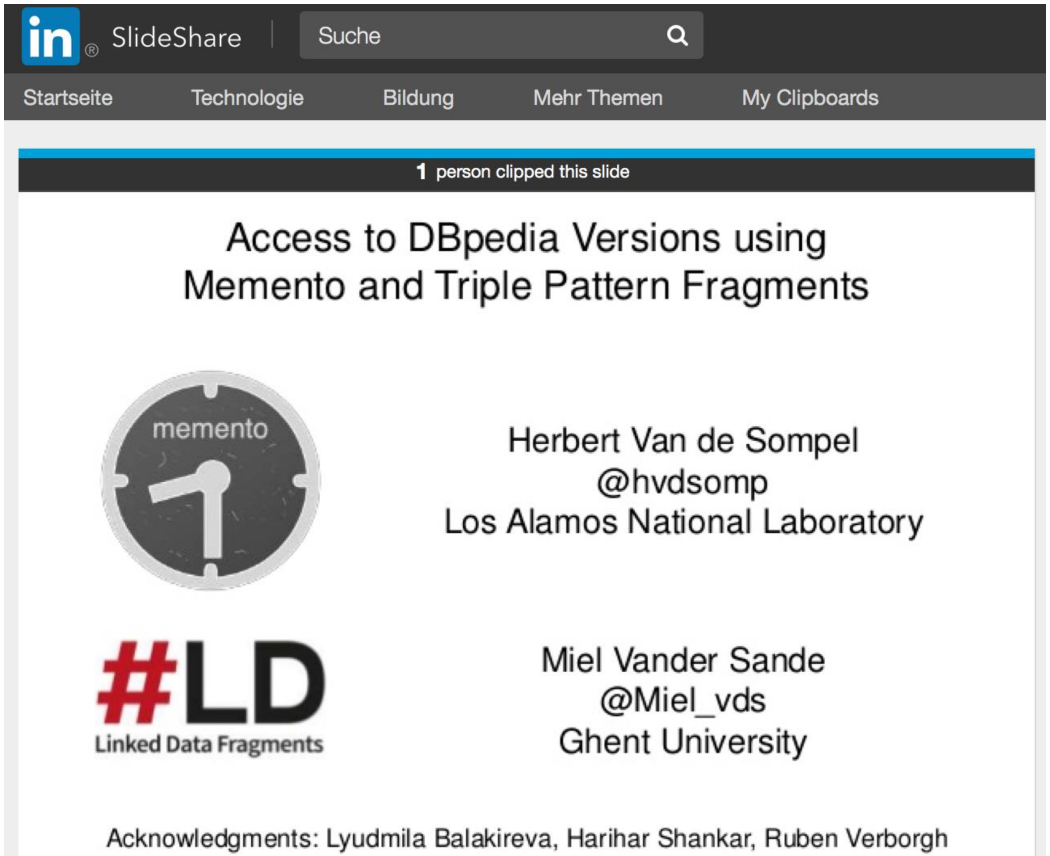


DBpedia Wayback Machine

- More complex operations/queries? Open challenge
  - a) On-demand? Query rewriting, similar to RDB2RDF
  - b) Batch: Fetch the desired information, then store and query it



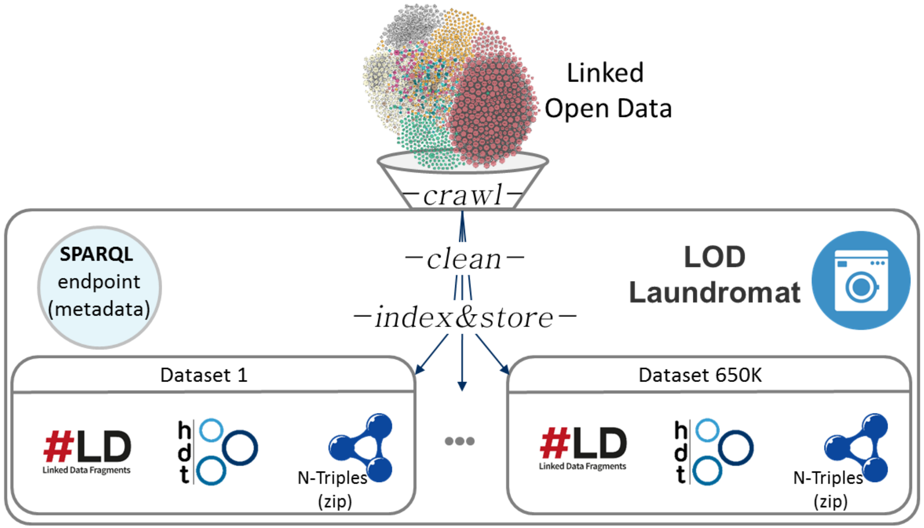
We are (obviously) not the only ones looking into this…



However:
Only one version per "irregular" dbpedia dump

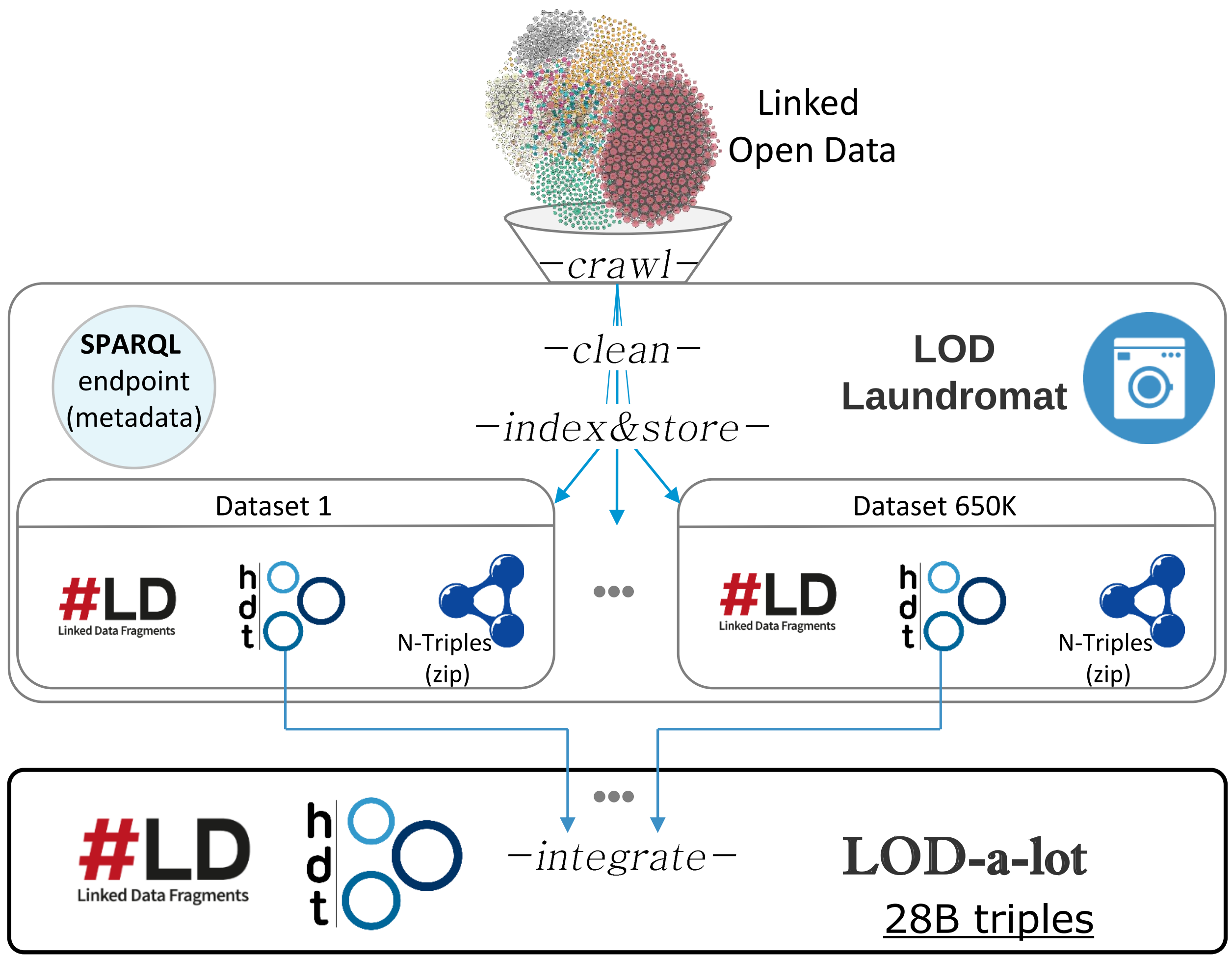
LOD Laundromat

- Lodlaundromat.org: a central repository of LD



- Problems?
  - Still you need to access/query 650K datasets
  - Of course the solution is not complete, but "a good approximation"

LOD-a-lot: LOD-a-lot: Low cost archiving of LOD



Kudos Javier D. Fernandez, Wouter Beek, Miguel A. Martínez-Prieto, Mario Arias, Ruben Verbogh

LOD-a-lot (some numbers)

#Triples	#Subjects	#Predicates	#Objects	#Common SO	#Literals
28,362,198,927	3,214,347,198	1,168,932	3,178,409,386	1,298,808,567	1,302,285,394

- Disk size:
  - HDT: 304 GB
  - HDT-FoQ (additional indexes): 133 GB

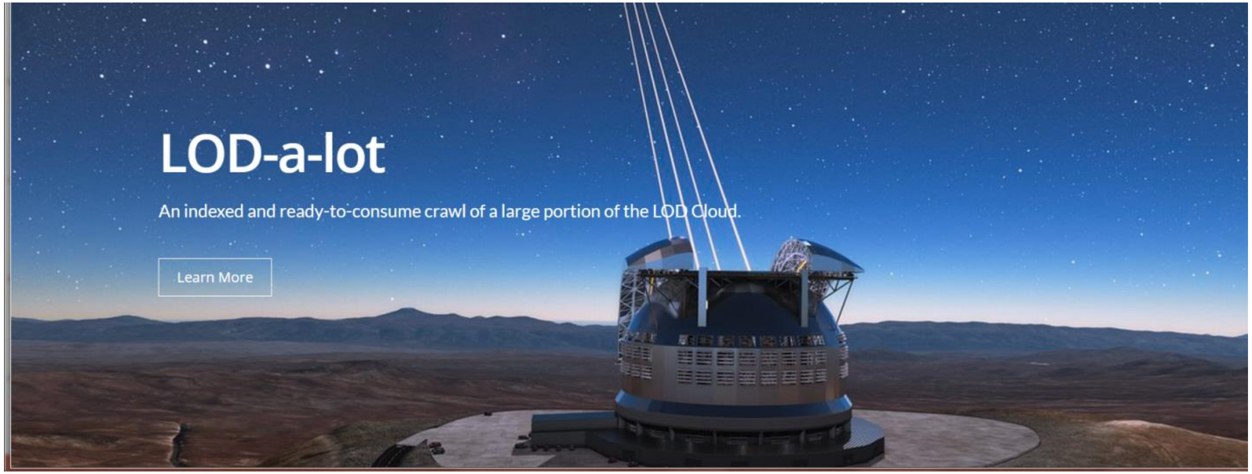
- Memory footprint (to query):
  - 15.7 GB of RAM (3% of the size)
  - 144 seconds loading time
    - 8 cores (2.6 GHz), RAM 32 GB, SATA HDD on Ubuntu 14.04.5 LTS

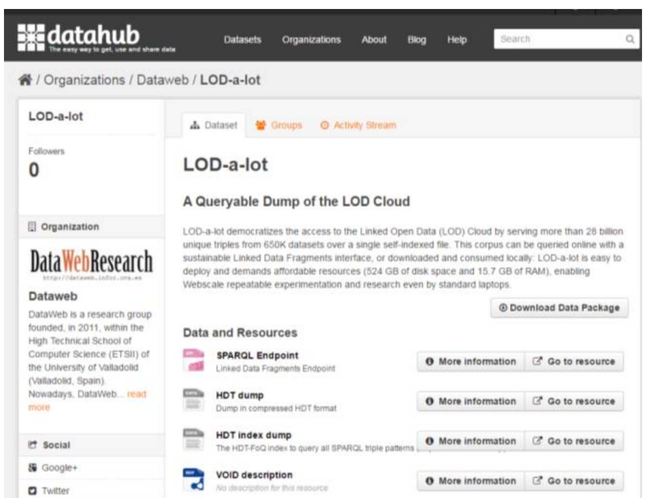
- LDF page resolution in milliseconds.

(LOD-a-lot creation took 64 h & 170GB RAM. HDT-FoQ took 8 h & 250GB RAM)

LOD-a-lot

http://purl.org/HDT/lod-a-lot





https://datahub.io/dataset/lod-a-lot

LOD-a-lot (some use cases)

- Archiving
  - We plan to have quarterly releases
- Query resolution at Web scale
- Evaluation and Benchmarking
  - No excuse ☺
- RDF metrics and analytics

Don't miss
Tuesday 16:05 - 17:35 !

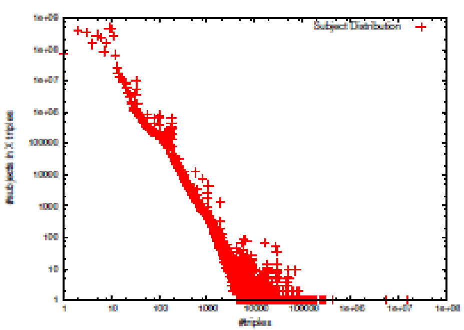
Room 6+7

Session 2.6 Data Science

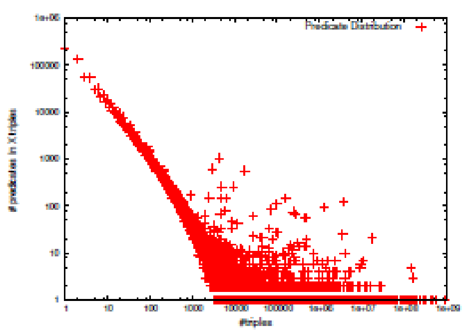
Querying and Visualization

Laura Hollink

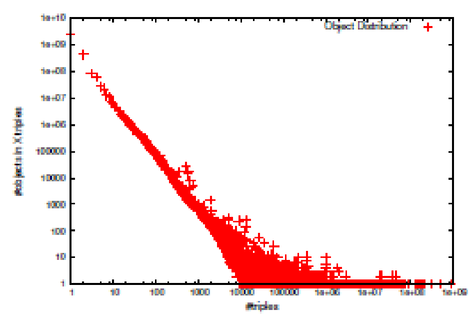
*LOD-a-lot: A Single-File Enabler for Data Science (Wouter Beek, Javier D. Fernández and Ruben Verborgh) **



subjects



predicates



objects

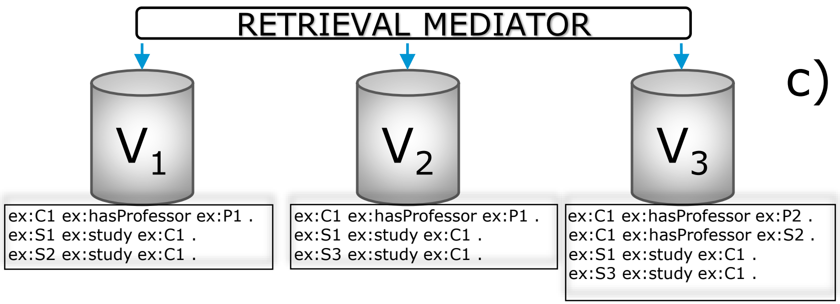
ACKs LOD-a-lot

The archiving problem

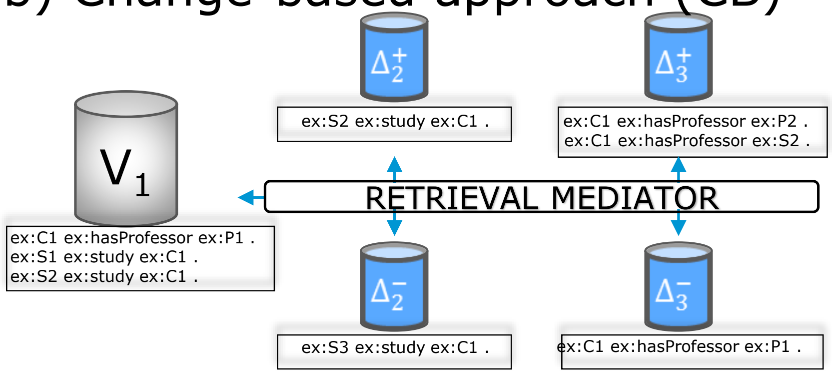
Now, how can we efficiently archive and perform time-based retrieval queries of a dataset?

RDF Archiving. Archiving policies

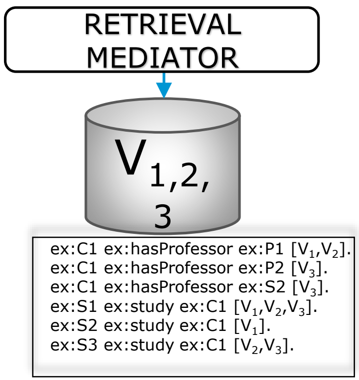
a) Independent Copies/Snapshots (IC)



b) Change-based approach (CB)



c) Timestamp-based approach (TB)



d) Hybrid approaches

BEAR

https://aic.ai.wu.ac.at/qadlod/bear.html

OVERVIEW BEAR-A BEAR-B BEAR-C BENCHMARK RESULTS PUBLICATIONS CONTACT QADLOD PROJECT

BEAR

BEnchmark of RDF ARchives

1. Overview

There is an emerging demand on efficiently archiving and (temporal) querying different versions of evolving semantic Web data. As novel archiving systems are starting to address this challenge, foundations/standards for benchmarking RDF archives are needed to **evaluate its storage space efficiency and the performance of different retrieval operations**.

To this end, we have developed a BEnchmark of RDF ARchives (BEAR), a test suite composed of three real-world datasets together with queries with varying complexity, covering a broad range of archiving use cases.

The DATA:

BEAR comprises three main datasets, namely BEAR-A, BEAR-B, and BEAR-C, each having different characteristics.

Dynamic Linked Data — BEAR-A

BEAR-A is composed of 58 weekly snapshots from the Dynamic Linked Data Observatory. BEAR-A provides triple pattern queries to test atomic queries such as Materialization, Diff, Version, etc.

DBpedia Live — BEAR-B

The BEAR-B dataset has been compiled from DBpedia Live changesets over the course of three months and contains the 100 most volatile resources along with their updates and real-world triple pattern queries from user logs.

Open Data portals — BEAR-C

BEAR-C used the Open Data Portal Watch project to take the datasets descriptions of the European Open Data portal for 32 weeks. With the help of Open Data experts, we created 10 complex queries that retrieve different information from datasets and files.

BEAR:
Benchmarking the Efficiency of RDF Archives

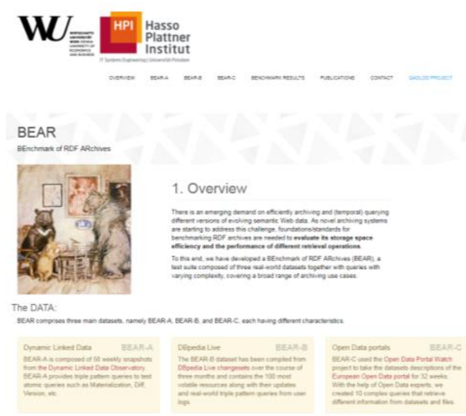
- Blueprint on benchmarking archives of semantic data
  - How can one define the corpus?
  - How can one design benchmark queries? Which queries?
- BEAR: concrete basic benchmark
  - Data: Crawl from Linked Data Observatory
  - Basic queries: Materialize, get Version…
  - Initial evaluation on archiving policies

https://aic.ai.wu.ac.at/qadlod/bear.html

BEAR:
Benchmarking the Efficiency of RDF Archives

- **Blueprint on benchmarking archives of semantic data**
  - **How can one define the corpus?**
  - **How can one design benchmark queries? Which queries?**

- BEAR: concrete basic benchmark
  - Data: Crawl from Linked Data Observatory
  - Basic queries: Materialize, get Version…
  - Initial evaluation on archiving policies



https://aic.ai.wu.ac.at/qadlod/bear.html

Benchmarking: **Define the corpus**

- Number of versions / size

Definition 1 (RDF Archive). *A version-annotated triple is an RDF triple (s,p,o) with a label $i \in \mathcal{N}$ representing the version in which this triple holds, denoted by the notation $(s,p,o):[i]$. An RDF archive graph $\mathcal{A}$ is a set of version-annotated triples.*

- Data dynamicity
  - Version change ratio
  - Version data growth

$$\delta_{i,j} = \frac{|\Delta^+_{i,j} \cup \Delta^-_{i,j}|}{|V_i \cup V_j|}$$

$$growth(V_i, V_j) = \frac{|V_j|}{|V_i|}.$$

- Data static core

$$\mathcal{C}_\mathcal{A} = \{(s,p,o) | \forall i \in \mathcal{N}, (s,p,o):[i] \in \mathcal{A}\}$$

- Total triples (version-oblivious)

$$\mathcal{O}_\mathcal{A} = \{(s,p,o) | \exists i \in \mathcal{N}, (s,p,o):[i] \in \mathcal{A}\}$$

- Others
  - RDF vocabulary
    - Per version / evolution

Benchmarking: **Define the queries**

- Structured query languages managing time.
  - Temporal databases (T-Quel, TSQL2)
    - Overlapping, meeting, before, equal, during, finish
  - RDF/Linked Data
    - SPARQL extensions
      - T-SPARQL, SPARQL-ST
      - AnQL
      - DIACHRON Query Language
        - SPARQL with specific constructors such as DATASET (similar to a named graph), VERSION, or CHANGES

Benchmarking: **Define the queries**

- Design of benchmark queries
  - Archive-driven Cardinality + Selectivity (disregard versions)
  - Version-driven Cardinality + Selectivity + dynamicity

$$dyn(Q, V_i, V_j) = \frac{|(\Omega_i \setminus \Omega_j) \cup (\Omega_j \setminus \Omega_i)|}{|\Omega_i \cup \Omega_j|}$$

  - Basic temporal retrieval features of queries
    - Mat (Q, V_i): version materialization
    - Diff (Q, V_i,V_j): delta materialization
    - Version(Q): results of Q annotated with the version
    - Join(Q1,V_i, Q2,V_j)
    - Change(Q): Returns versions in which Diff(Q, V_i, V_{i-1}) $!=\emptyset$

Benchmarking: **Define the queries**

- Instantiation of archive queries in AnQL [1]

  - *Antoine Zimmermann, Nuno Lopes, Axel Polleres, and Umberto Straccia.* A general framework for representing, reasoning and querying with annotated Semantic Web data*. Journal of Web Semantics (JWS), 12:72--95, March 2012.*

  - ***Mat(Q,V)***
  - *Diff(Q,V1,V2)*
  - Ver(Q)
  - join(Q1,vi,Q2,vj)
  - Change(Q)

```
SELECT * WHERE { Q :[v] }
```

Benchmarking: **Define the queries**

- Instantiation of archive queries in AnQL [1]

  - *Antoine Zimmermann, Nuno Lopes, Axel Polleres, and Umberto Straccia.* A general framework for representing, reasoning and querying with annotated Semantic Web data*. Journal of Web Semantics (JWS), 12:72--95, March 2012.*

  - *Mat(Q,V)*
  - ***Diff(Q,V1,V2)***
  - Ver(Q)
  - join(Q1,vi,Q2,vj)
  - Change(Q)

```
SELECT * WHERE {
{ { {Q :[v1]} MINUS {Q :[v2]} } BIND (v1 AS ?V )
}
UNION
{ { {Q :[v2] } MINUS {Q :[v1]}} BIND (v2 AS ?V )
}
```

Benchmarking: **Define the queries**

- Instantiation of archive queries in AnQL [1]

  - *Antoine Zimmermann, Nuno Lopes, Axel Polleres, and Umberto Straccia.* A general framework for representing, reasoning and querying with annotated Semantic Web data*. Journal of Web Semantics (JWS), 12:72--95, March 2012.*

  - *Mat(Q,V)*
  - *Diff(Q,V1,V2)*
  - **Ver(Q)**
  - join(Q1,vi,Q2,vj)
  - Change(Q)

```
SELECT * WHERE { Q :?V }
```

Benchmarking: **Define the queries**

- Instantiation of archive queries in AnQL [1]

  - *Antoine Zimmermann, Nuno Lopes, Axel Polleres, and Umberto Straccia.* A general framework for representing, reasoning and querying with annotated Semantic Web data*. Journal of Web Semantics (JWS), 12:72--95, March 2012.*

  - *Mat(Q,V)*
  - *Diff(Q,V1,V2)*
  - Ver(Q)
  - **join(Q1,v1,Q2,v2)**
  - Change(Q)

```
SELECT * WHERE { {Q :[v1]} {Q :[v2]} }
```

Benchmarking: **Define the queries**

- Instantiation of archive queries in AnQL [1]

  - *Antoine Zimmermann, Nuno Lopes, Axel Polleres, and Um* general framework for representing, reasoning and query annotated Semantic Web data. *Journal of Web Semantics 95, March 2012.*

  - *Mat(Q,V)*
  - *Diff(Q,V1,V2)*
  - Ver(Q)
  - join(Q1,vi,Q2,vj)
  - **Change(Q)**

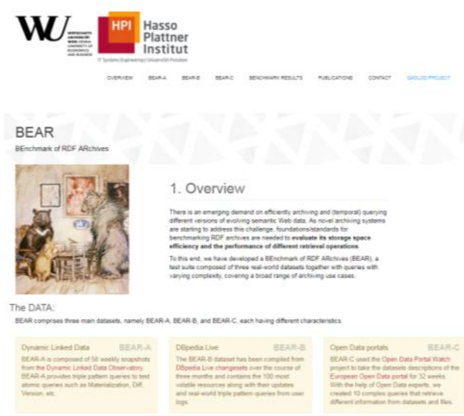
Open question remains:
What is the right query syntax for archive queries?

```
SELECT ?V1 ?V2 WHERE
{ {{Q :?V1 } MINUS {Q :?V2}} UNION
  {{Q :?V2 } MINUS {Q :?V1}}
  FILTER( abs(?V1-?V2) = 1 ) }
```

BEAR: Benchmarking the Efficiency of RDF Archiving

- blueprint on benchmarking archives of semantic data
  - How can one define the corpus?
  - How can one design benchmark queries? Which queries?

- **BEAR: concrete basic benchmark**
  - **Data: Crawl from Linked Data Observatory**
  - **Basic queries: Materialize, get Version…**
  - **Initial evaluation of archiving policies**



https://aic.ai.wu.ac.at/qadlod/bear.html

BEAR: Benchmarking the Efficiency of RDF Archiving

- Queries and systems
  - We implemented and evaluate archiving systems on Jena-TDB and HDT, based on IC, CB and TB policies.
    - Serve as an initial baseline to compare archiving systems
    - More info: https://aic.ai.wu.ac.at/qadlod/bear.html

BEAR datasets

Dynamic Linked Data — BEAR-A

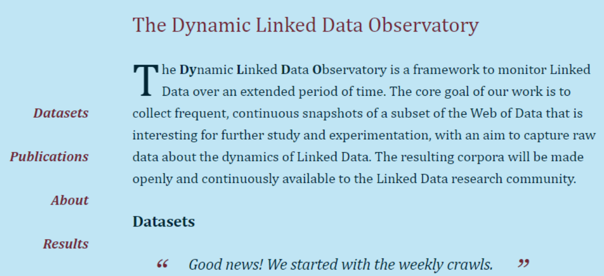
BEAR-A is composed of 58 weekly snapshots from the Dynamic Linked Data Observatory. BEAR-A provides triple pattern queries to test atomic queries such as Materialization, Diff, Version, etc.

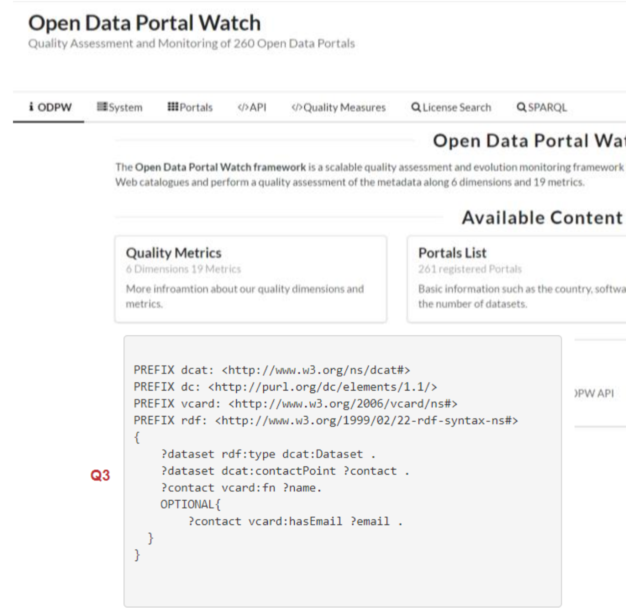
DBpedia Live — BEAR-B

The BEAR-B dataset has been compiled from DBpedia Live changesets over the course of three months and contains the 100 most volatile resources along with their updates and real-world triple pattern queries from user logs.

Open Data portals — BEAR-C

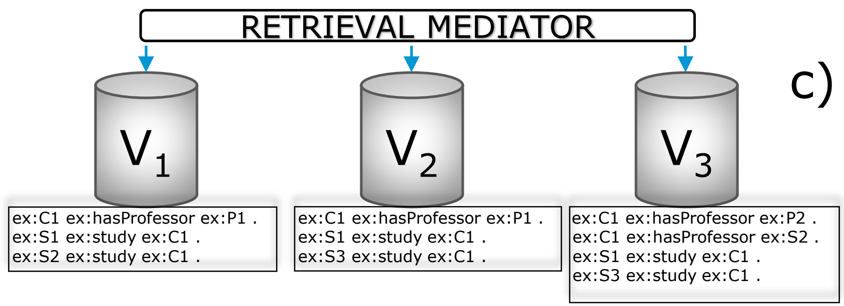
BEAR-C used the Open Data Portal Watch project to take the datasets descriptions of the European Open Data portal for 32 weeks. With the help of Open Data experts, we created 10 complex queries that retrieve different information from datasets and files.



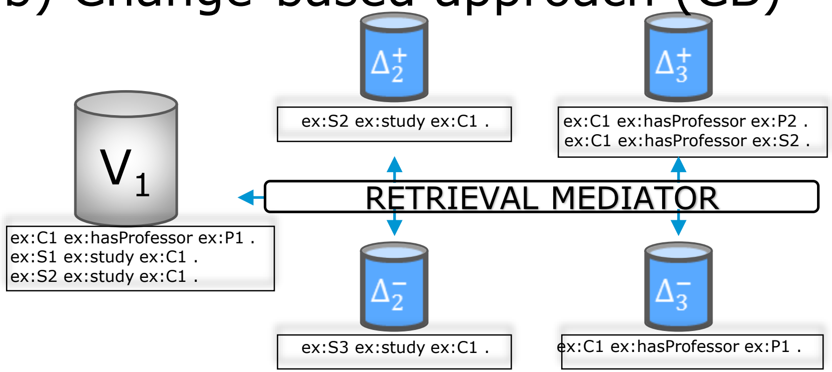


RDF Archiving. Archiving policies

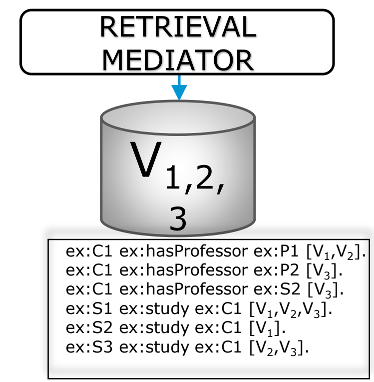
a) Independent Copies/Snapshots (IC)



b) Change-based approach (CB)



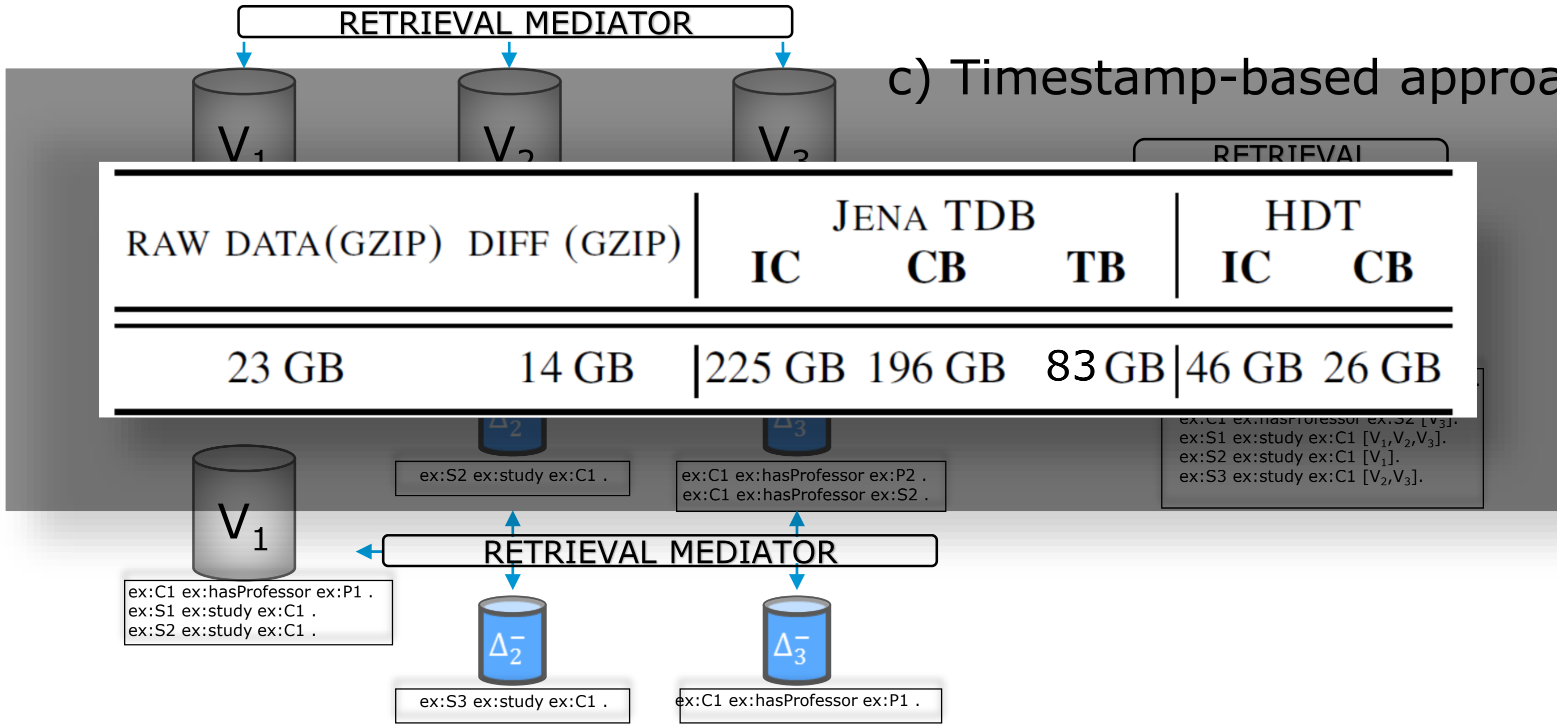
c) Timestamp-based approach (TB)



RDF Archiving. Archiving policies

a) Independent Copies/Snapshots (IC)

c) Timestamp-based approach (TB)



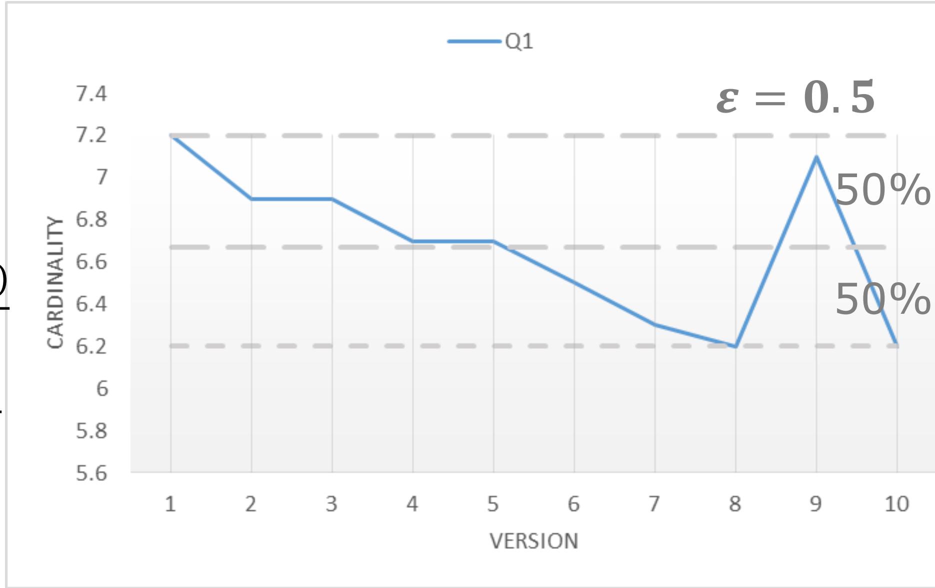
RAW DATA(GZIP)	DIFF (GZIP)	JENA TDB IC	JENA TDB CB	JENA TDB TB	HDT IC	HDT CB
23 GB	14 GB	225 GB	196 GB	83 GB	46 GB	26 GB

Time-based access. Queries

- **Triple Pattern queries**
- **Queries with "similar" number of results in all versions.**
  - **ε-stable query**

$$max_{\forall i \in N}\ CARD(Q,V_i) \leq (1+\varepsilon)\frac{\sum_{\forall i \in N} CARD(Q,V_i)}{|N|}$$

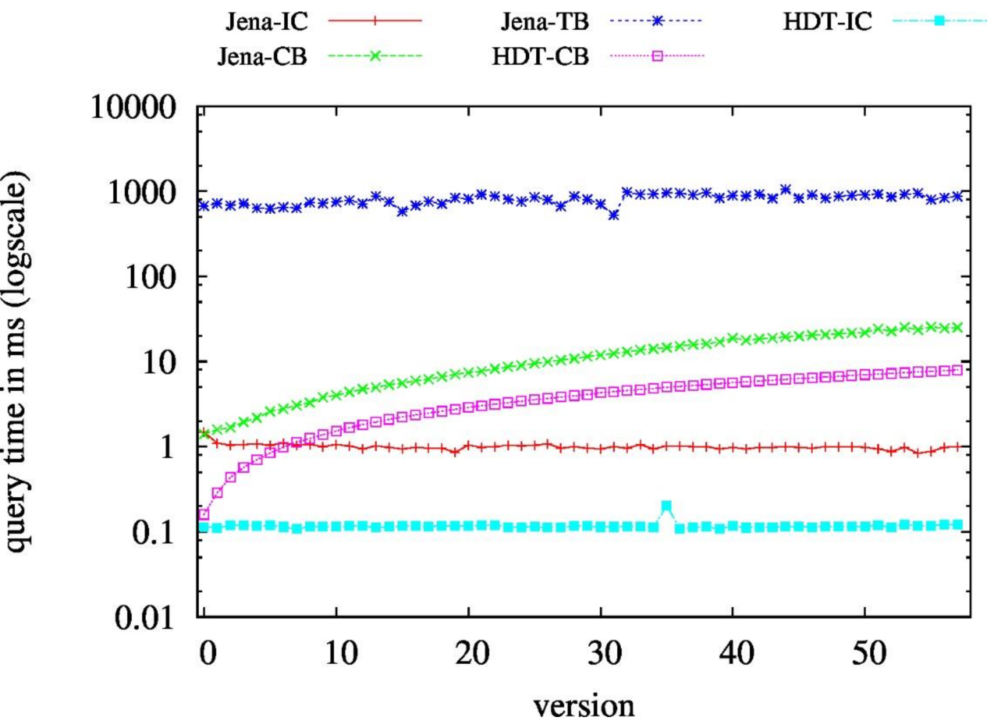
$$min_{\forall i \in N}\ CARD(Q,V_i) \geq (1+\varepsilon)\frac{\sum_{\forall i \in N} CARD(Q,V_i)}{|N|}$$

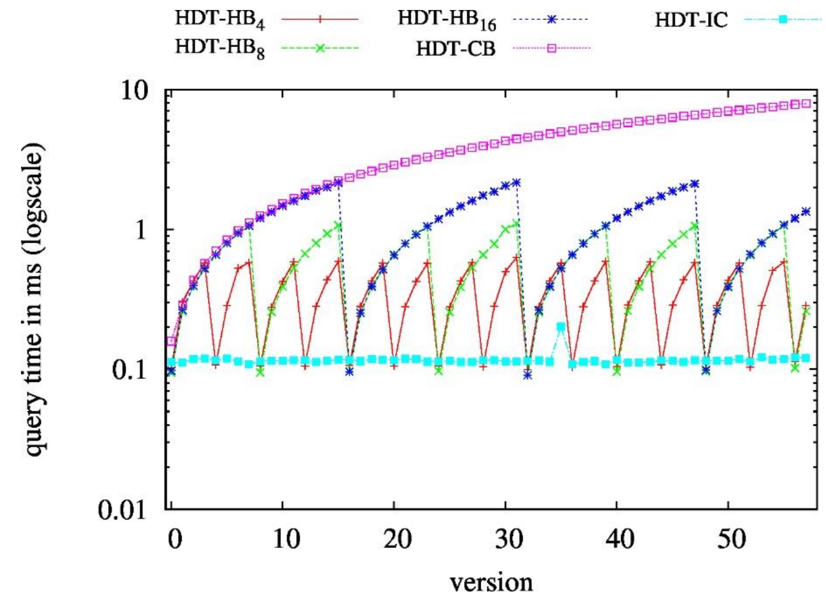


Time-based access. Queries

Materialize (s,?,? ; version)

IC	CB	HB_4	HB8	HB_{16}
48 GB	28 GB	34 GB	31 GB	29 GB



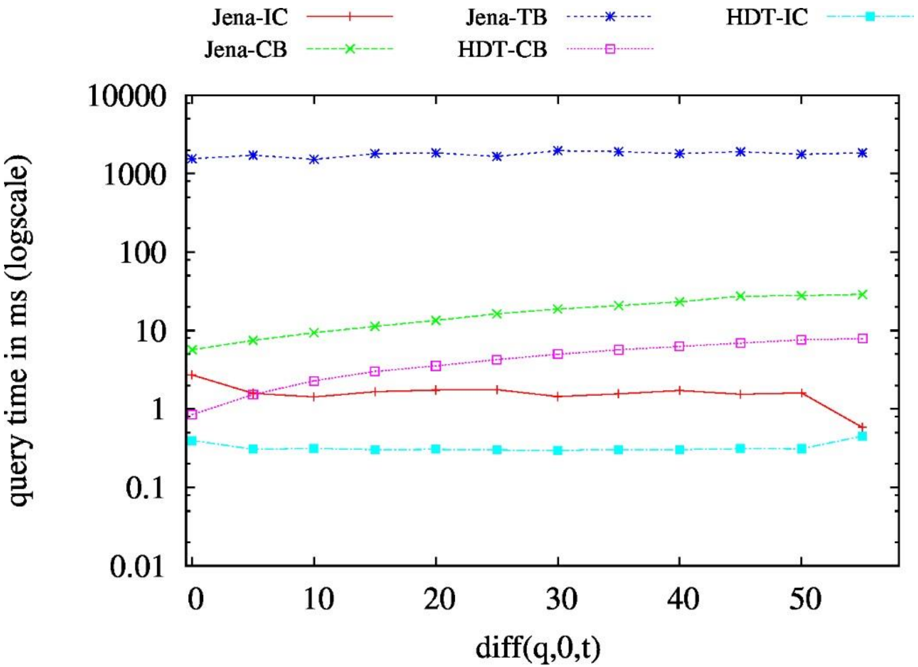


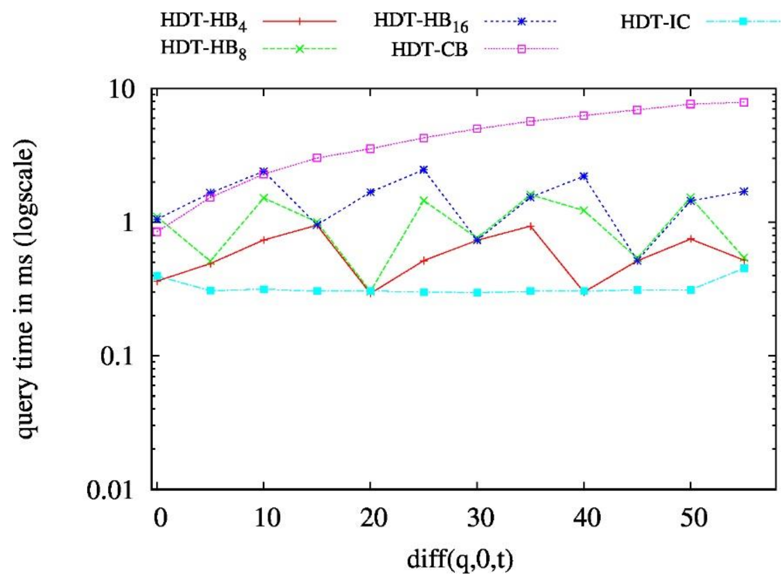
Hybrid approach

Time-based access. Queries

diff(?,?,o ; version0 ; version t)

IC	CB	HB_4	HB8	HB_{16}
48 GB	28 GB	34 GB	31 GB	29 GB

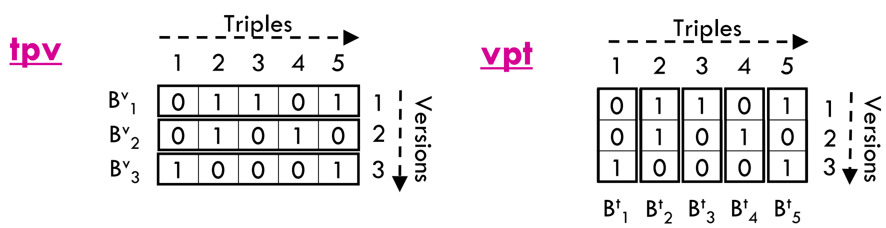


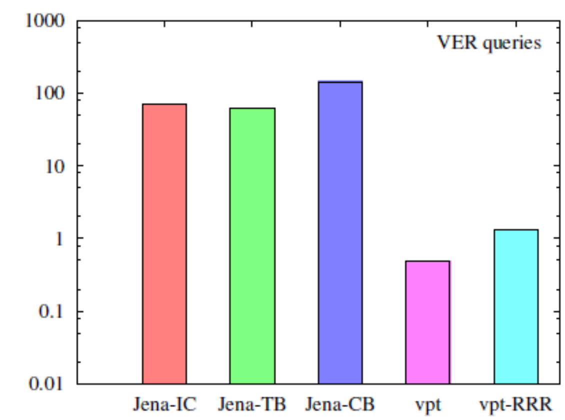


Hybrid approach

Self-Indexing RDF Archives: v-RDFCSA

- RDFCSA: RDF index based on a Compressed Suffix Array
- v-RDFCSA[2] is designed as a lightweight TB approach
- Version information encoding
  - Any triple can be identified by the position of its subject within SA
  - Let be N the number of different versions and n the set of version-oblivious triples
  - Two alternative encoding strategies
    - tpv: N bitsequences, each position i encodes if the triple i appears in the version
    - vpt: n bitsequences, each position i encodes if the version i includes the triple





Performs more than **one order of magnitude** faster than Jena-TDB

[2] Ana Cerdeira-Pena, Antonio Fariña, Javier D. Fernández, and Miguel A. Martínez-Prieto. Self-Indexing RDF Archives. Data Compression Conference (DCC), 2016.

Open Data evolution

WIRTSCHAFTS UNIVERSITÄT WIEN VIENNA UNIVERSITY OF ECONOMICS AND BUSINESS

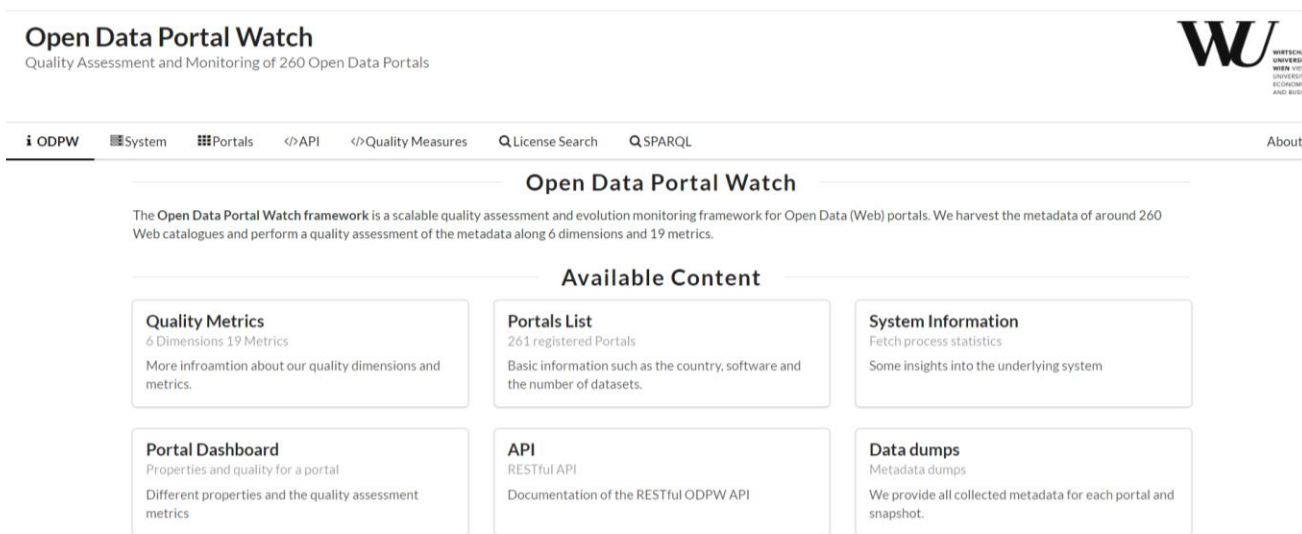
COMPLEXITY SCIENCE HUB VIENNA

How is Open Data evolving?

OPEN DATA PORTAL WATCH
... a first step.

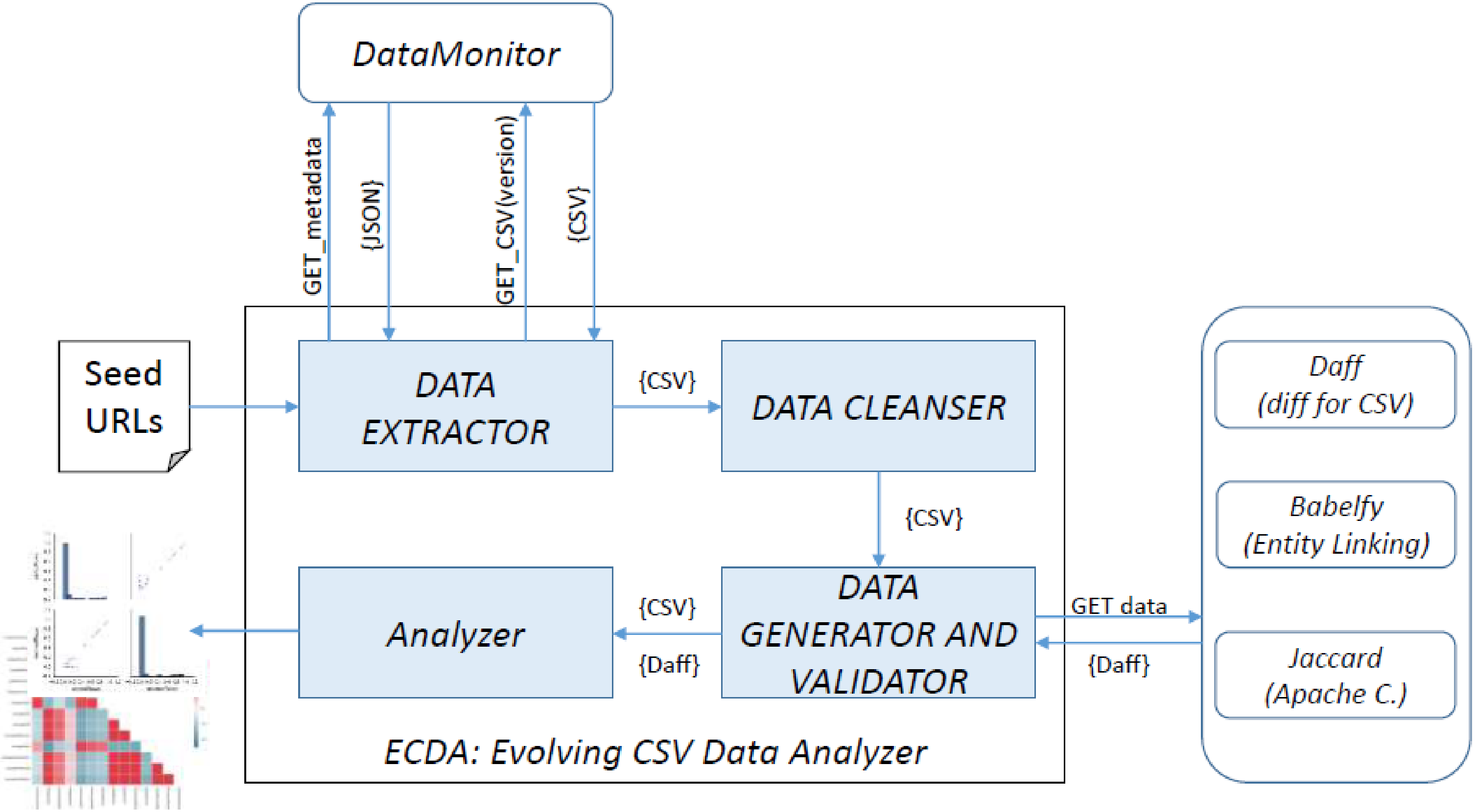
- Periodically monitoring a list of Open Data Portals
  - 90 CKAN powered Open Data Portals
- Quality assessment
- Evolution tracking
  - Meta data
  - Data

http://data.wu.ac.at/portalwatch/



Jürgen Umbrich
Sebastian Neumaier

ECDA: Evolving CSV Data Analyzer



ECDA: Evolving CSV Data Analyzer

- Analysis of 726 datasets
  - Mean of 18 versions per file
  - Mean of 430 rows and 5.4 columns
- Increasing nature (x1.85 number of rows)
  - Small value modifications (0.85 jaccard)
  - Mostly string types (80% of 8-25 characters)

Data domain	Number of versions
www.landesdatenbank.nrw.de	493
ourairports.com	83
data.gov.au	23
Other domains	127

Type of changes	Number of sets
STATIC: No changes in number of rows or columns	10 (1.4%)
INCREASING: No deletions of rows or columns, just additions.	71 (9.8%)
DECREASING: No deletions of rows or columns, just deletions.	2 (0.3%)
SAME SCHEMA: Only changes (additions or deletions) of rows	117 (16.12%)
SAME ROWS: Only changes (additions or deletions) of columns	9 (1.24%)
IRREGULAR: Changes (additions or deletions) of rows and columns	600 (82.64%)

Variable	Information
Number of versions	mean= 17.83, std=6.18
Original rows	mean= 431.23, std=1596.23
Original columns	mean= 5.40, std=8.48
Added rows	mean= 1.85, std=21.37
Deleted rows	mean= 0.79, std=2.02
Modified rows	mean= 0.09, std=0.80
Added columns	mean= 0.23, std=0.28
Deleted columns	mean= 0.22, std=0.29
Modified columns	mean= 0, std=0
Changed cells	mean= 1663.96, std=29564.60

ECDA: Evolving CSV Data Analyzer

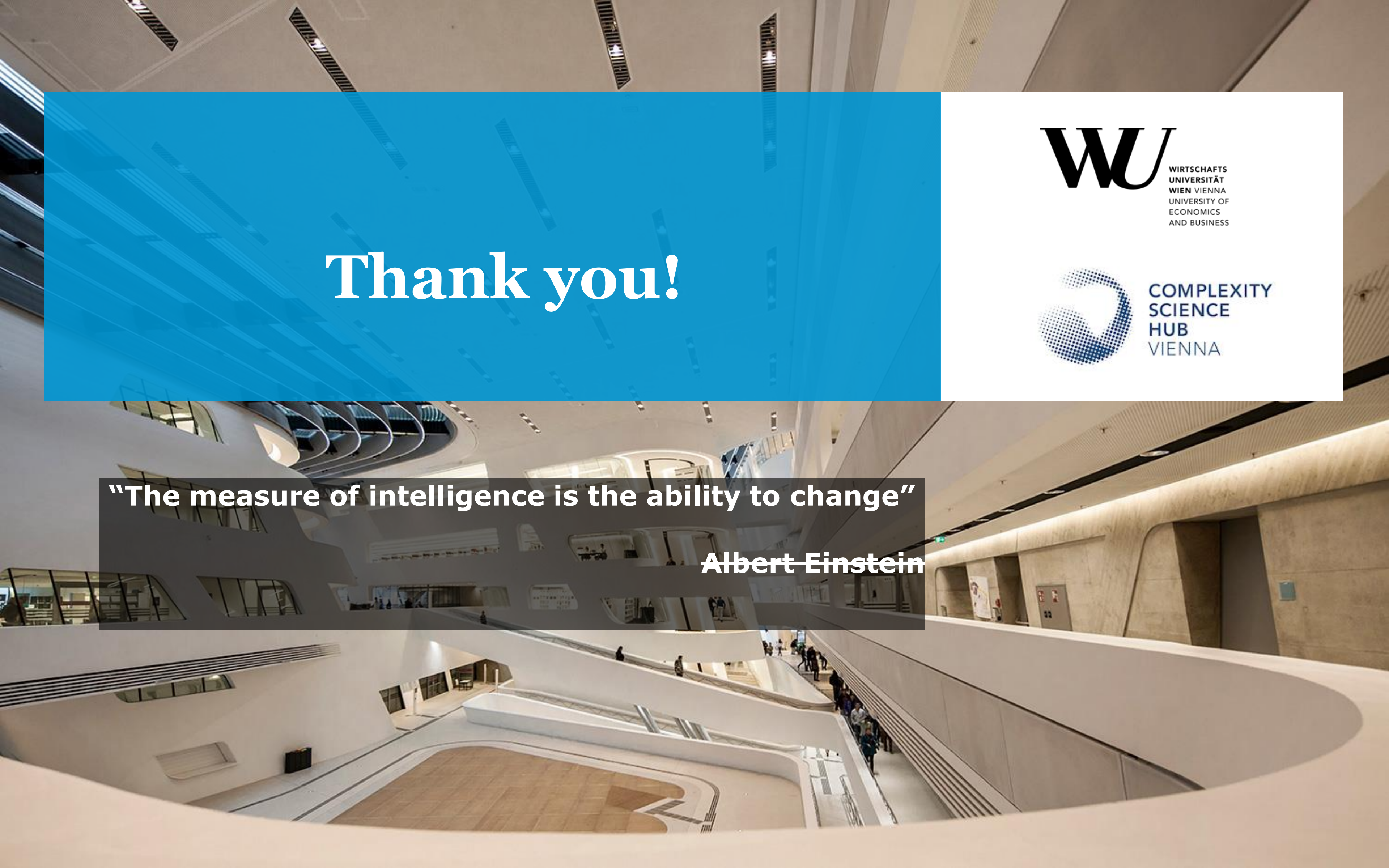
- Analysis of 726 datasets
  - Entities (Babelfy)
    - On average there are around 0.07 entities per cell
    - Entities are static in the header (a mean of 3)
    - 1/3 of the entities change across time
    - Number of entities slightly decrease in time

Variable	Information
Number of cells	mean= 2597.09, std=9890.53
Original entities	mean= 180.86, std=184.41
Actual entities	mean= 149.40, std=202.90
New entities	mean= 86.65, std=170.97
Entities maintained	mean= 62.75, std=131.23
Original header entities	mean= 2.68, std=11.09
Actual header entities	mean= 2.74, std=11.14
Header entities maintained	mean= 2.64, std=11.07

Finally, many open questions remain still!

- Archiving and querying evolving semantic Web data

Objective	Research Question
Representation of archives	❖ **minimize** the redundant information ❖ respect the original modeling and **provenance** information (e.g. LOD-a-lot)
Query language	❖ design **a query language** satisfying these requirements for evolving interlinked data ❖ our BEAR operations are meant to be an **extensible** starting point
Indexing	❖ index archives at **large scale** ❖ keep up with **evolution rate** (streaming vs. archiving) to process the queries efficiently
Analysis/Optimization	❖ use **evolution patterns** to **optimize** representations and queries ❖ Querying archives of **structures and non-structured** sources? E.g. Open Data!
Application	❖ **LOD-a-lot** is a good examples but **modularity can be improved** ❖ **Any low-cost** but functional archiving at LOD scale can be a major milestone for the community



Thank you!

WIRTSCHAFTS UNIVERSITÄT WIEN VIENNA UNIVERSITY OF ECONOMICS AND BUSINESS

COMPLEXITY SCIENCE HUB VIENNA

"The measure of intelligence is the ability to change"

~~Albert Einstein~~