Dynamic Data Citation
Andreas Rauber
30.1.2017, Brussels
Background

**Business Intelligence / Data Science:**
- Machine Learning, Signal Processing, Data Warehousing, IR

**Complex data analytics projects**

**CRISP-DM:** Cross-Industry Standard Process for Data Mining

**Trust:** Traceability, Reproducibility

**ACM Statement on Algorithmic Transparency & Accountability**

- How can we document the processes?
- How can we identify the data that was used?
Motivation

Identifying the data used seems trivial:

1. Put data in data repository
2. Assign identifier (DOI, Ark, URI, …)
3. Make (and keep) it accessible
4. Refer to it in analysis document / dashboard / …

So where are the challenges?

- Dynamics
- Granularity
Identification of Dynamic Data

Identifiable datasets usually have to be static
- Fixed set of data, no changes:
  no corrections to errors, no new data being added

But: data is **dynamic**
- Adding new data, correcting errors, enhancing data quality, ...
- Changes sometimes highly dynamic, at irregular intervals

Current approaches
- Identifying entire data stream, without any versioning
- Using “accessed at” date
- “Artificial” versioning by identifying batches of data (e.g. annual), aggregating changes into releases (time-delayed!)

Would like to identify precisely the **data as it existed at any specific point in time**
Granularity of Subsets

What about the **granularity** of data to be identified?
- Massive collections of data in any repository
- Analysts use specific subsets of data
- Need to precisely identify the subset used

Current approaches
- Storing a copy of subset as used in study -> scalability
- Citing entire dataset, providing textual description of subset (methods section) -> imprecise (ambiguity)
- Storing list of record identifiers in subset -> scalability, not for arbitrary subsets (e.g. when not entire record selected)

Would like to be able to identify precisely the **subset of (dynamic) data used** in a process
Data Citation – Requirements

- **Allow analysts to easily identify the data used**
- **Dynamic data, for any type of data**
  - corrections, additions, ... for relational DBs, XML, files, ...
- **Arbitrary subsets of data (granularity)**
  - rows/columns, time sequences, ...
  - from single number to the entire set
- **Stable across technology changes**
  - e.g. migration to new database system
- **Machine-actionable**
  - not just machine-readable, definitely not just human-readable and interpretable
- **Scalable to very large / highly dynamic datasets**
  - But: should also work for small and/or static datasets!
Dynamic Data Citation

We have: Data + Means-of-access
Dynamic Data Citation

We have: Data + Means-of-access

Dynamic Data Citation:
Cite (dynamic) data dynamically via query!
Dynamic Data Citation

We have: Data + Means-of-access

Dynamic Data Citation:
Cite (dynamic) data dynamically via query!

Steps:

1. Data → versioned (history, with time-stamps)
Dynamic Data Citation

We have: Data + Means-of-access

Dynamic Data Citation:
Cite (dynamic) data dynamically via query!

Steps:

1. Data $\rightarrow$ versioned (history, with time-stamps)

Researcher creates working-set via some interface:

2. Access $\rightarrow$ store & assign persistent identifier to “QUERY”, plus
   - **Time-stamping** for re-execution against versioned DB
   - **Re-writing** for normalization, unique-sort, mapping to history
   - **Hashing** result-set: verifying identity/correctness
     (plus a few more things) leading to landing page

S. Pröll, A. Rauber. **Scalable Data Citation in Dynamic Large Databases: Model and Reference Implementation.** In IEEE Intl. Conf. on Big Data 2013 (IEEE BigData2013), 2013
http://www ifs tuwien ac at/ andi/publications/pdf/pro_ieeebigdata13 pdf
14 Recommendations grouped into 4 phases:
- Preparing data and query store
- Persistently identifying specific data sets
- Resolving PIDs
- Upon modifications to the data infrastructure

2-page flyer

Data Citation – Recommendations

Preparing Data & Query Store
- R1 – Data Versioning
- R2 – Timestamping
- R3 – Query Store

When Data should be persisted
- R4 – Query Uniqueness
- R5 – Stable Sorting
- R6 – Result Set Verification
- R7 – Query Timestamping
- R8 – Query PID
- R9 – Store Query
- R10 – Citation Text

When Resolving a PID
- R11 – Landing Page
- R12 – Machine Actionability

Upon Modifications to the Data Infrastructure
- R13 – Technology Migration
- R14 – Migration Verification
Dynamic Data Citation

- Analyst uses workbench to identify subset of data
- Upon executing selection („download“) user gets
  - Data (package, access API, ...)
  - PID (e.g. DOI) (Query is time-stamped and stored)
  - Hash value computed over the data for local storage
  - Recommended citation text (e.g. BibTeX, Endnote, text)
- PID resolves to landing page
  - Provides detailed metadata, link to parent data set, subset,...
  - Option to retrieve original data OR current version OR changes
- Upon activating PID associated with a data citation
  - Query is re-executed against time-stamped and versioned DB
  - Results as above are returned
- Query store aggregates data usage
Dynamic Data Citation

- Analyst/her uses workbench to identify subset of data
- Upon executing selection ("download") user gets:
  - Data (package, access API, ...)
  - PID (e.g. DOI) (Query is time-stamped and stored)
  - Hash value computed over the data for local storage
  - Recommended citation text (e.g. BibTeX)
- PID resolves to landing page
  - Provides detailed metadata, link to parent data set, subset, ...
  - Option to retrieve original data OR current version OR changes
- Upon activating PID associated with a data citation
  - Query is re-executed against time-stamped and versioned DB
  - Results as above are returned
- Query store aggregates data usage

Note: query string provides excellent provenance information on the data set!
Dynamic Data Citation

- Analyst uses workbench to identify subset of data
- Upon executing selection ("download") user gets
  - Data (package, access API, …)
  - PID (e.g. DOI) (Query is time-stamped and stored)
  - Hash value computed over the data for local storage
  - Recommended citation text (e.g. BibTeX)
- PID resolves to landing page
  - Provides detailed metadata, link to parent data set, subset,…
  - Option to retrieve original data OR current version OR changes
- Upon activating PID associated with a data citation
  - Query is re-executed against time-stamped and versioned DB
  - Results as above are returned
- Query store aggregates data usage

Note: query string provides excellent provenance information on the data set!

This is an important advantage over traditional approaches relying on, e.g. storing a list of identifiers/DB dump!!!
Dynamic Data Citation

- Analyst her uses workbench to identify subset of data
- Upon executing selection ("download") user gets
  - Data (package, access API, …)
  - PID (e.g. DOI)
  - Hash value computed over the data for local storage
  - Recommended citation text (e.g. BibTeX)
- PID resolves to landing page
  - Provides detailed metadata, link to parent data set, subset, …
  - Option to retrieve original data OR current version OR changes
- Upon activating PID associated with a data citation
  - Query is re-executed against time-stamped and versioned DB
  - Results as above are returned
- Query store aggregates data usage

Note: query string provides excellent provenance information on the data set!

This is an important advantage over traditional approaches relying on, e.g. storing a list of identifiers/DB dump!!!

Identify which parts of the data are used. If data changes, identify which queries (studies) are affected!

This is an important advantage over traditional approaches relying on, e.g. storing a list of identifiers/DB dump!!!
Pilots / Adopters

- Series of Webinars presenting implementations
  - Recordings, slides, supporting papers
  - Implementing of the RDA Data Citation Recommendations by the Climate Change Centre Austria (CCCA) for a repository of NetCDF files
  - Implementing the RDA Data Citation Recommendations for Long-Tail Research Data / CSV files
  - Implementing the RDA Data Citation Recommendations in the Distributed Infrastructure of the Virtual and Atomic Molecular Data Center (VAMDC)
  - Implementation of Dynamic Data Citation at the Vermont Monitoring Cooperative
  - Adoption of the RDA Data Citation of Evolving Data Recommendation to Electronic Health Records
Industry Partners

- Less willing to share insights into their DM practices
- IT Solutions company
  - Enterprise offering data management / transformation / data migration / ETL services
  - Relational Databases (Oracle, MySQL, Postgres): study on different versioning / historization approaches
  - Question: how to deal with schema evolution?
  - Likely will get permission to publish the study
- In many cases core building blocks already in place (versioning, query processing, ...)
- Straightforward mechanism to add auditability for source data used in analysis / processing
- Add query re-writing, storing queries, interface adaptations
- **Effort** required: it depends – pilots: 5-8 PM
Summary - Advantages

- Precisely identify any arbitrary subset of data
- Principles applicable to all types of data
- Straightforward to implement in most settings
- Optimizations for high-volume / very dynamic data possible
- Transparent for the analyst / data scientist
- Reduces documentation effort for analysts / data scientist
- Reduces data management complexity for data centre
- Increases traceability of results, trust
Thank you!

Thanks!

https://rd-alliance.org/working-groups/data-citation-wg.html