Enabling Precise Identification and Citabilityof Dynamic Data
–
Recommendations of the RDA Working Group on Data Citation

Andreas Rauber

Vienna University of Technology
Favoritenstr. 9-11/188
1040 Vienna, Austria
rauber@ifs.tuwien.ac.at
http://ww.ifs.tuwien.ac.at/~andi

ifs FACULTY OF INFORMATICS

Outline

- Why should we want to cite data?
- What are the challenges in data identification and citation?
- How should we do it, according to the RDA WG?
- Who is doing it so far, and how?
- Summary

Why to cite data?

- Data is the basis for almost everything
  - eScience, digital humanities,
  - Industry 4.0
  - Driving policies, society, …
- Why should we cite data?
  - Prevent scientific misconduct (“extrinsic”) ?

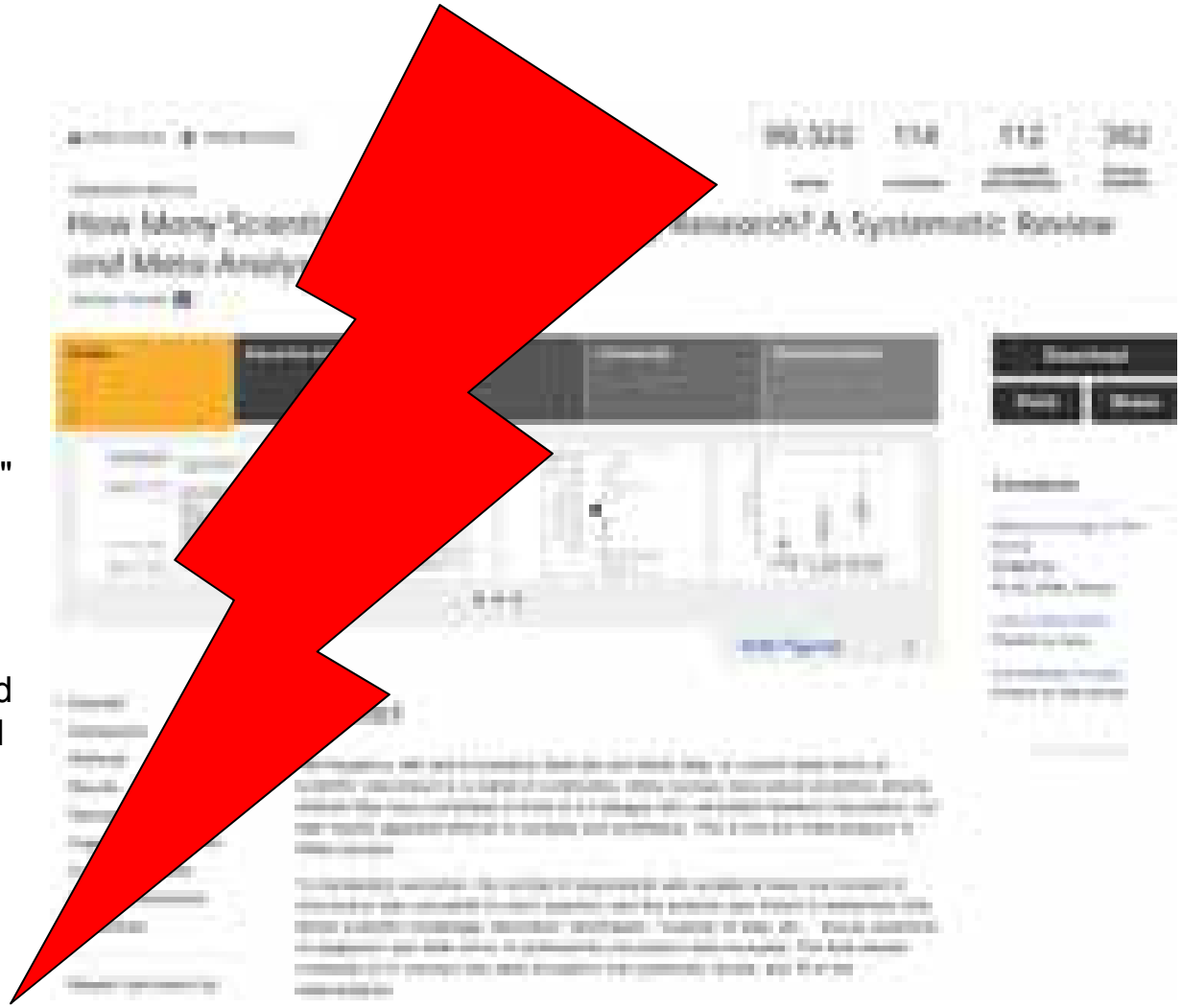
Prevent Scientific Misconduct

- On average: ~ 2% of scientists admitted they had "fabricated" (made up), "falsified" or "altered" data to "improve the outcome" at least once.
- 34% admitted to other questionable research practices including "failing to present data that contradict one's own previous research" and "dropping observations or data points from analyses based on a gut feeling that they were inaccurate."
- 14% knew someone who had fabricated, falsified or altered data.
- Up to 72% knew someone who had committed other questionable research practices.

Source: http://www.plosone.org

Prevent Scientific Misconduct

- On average: ~ 2% of scientists admitted they had "fabricated" (made up), "falsified" or "altered" data to "improve the outcome" at least once.
- 34% admitted to other questionable research practices including "failing to present data that contradict one's own previous research" and "dropping observations or data points from analyses based on a gut feeling that they were inaccurate."
- 14% knew someone who had fabricated, falsified or altered data.
- Up to 72% knew someone who had committed other questionable research practices.



Source: http://www.plosone.org

Why to cite data?

- Data is the basis for almost everything
  - eScience, digital humanities,
  - Industry 4.0
  - Driving policies, society, …
- Why should we cite data?
  - Prevent scientific misconduct (“extrinsic”) ?
  - Give credit (“altruistic”) ?

Giving credit

- Prime motivator for sharing data
- Shared data gets cited more frequently
- Citations are the currency of science

Giving credit

- Prime motivator for sharing da
- Shared data gets cited more
- Citations are the currency

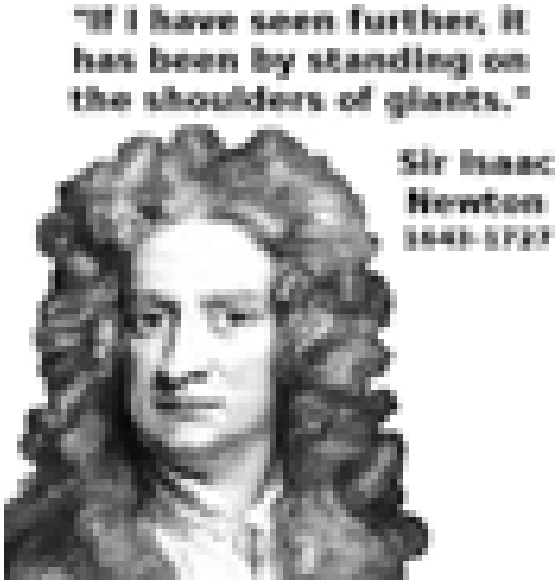
Why to cite data?

- Data is the basis for almost everything
  - eScience, digital humanities,
  - Industry 4.0
  - Driving policies, society, …
- Why should we cite data?
  - Prevent Scientific misconduct (“extrinsic”) ?
  - Give credit (“altruistic”) ?
  - Show solid basis (“egoistic”) ?

Citing to give credit

Why do we cite papers? (“related work”)

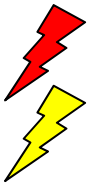
- Fundamental basis for own work – foundation!
- No need to prove - it’s been done!
- Speed-up the process, efficiency
- Basis for discourse, scientific work, …



Why to cite data?

- Data is the basis for almost everything
  - eScience, digital humanities,
  - Industry 4.0
  - Driving policies, society, …

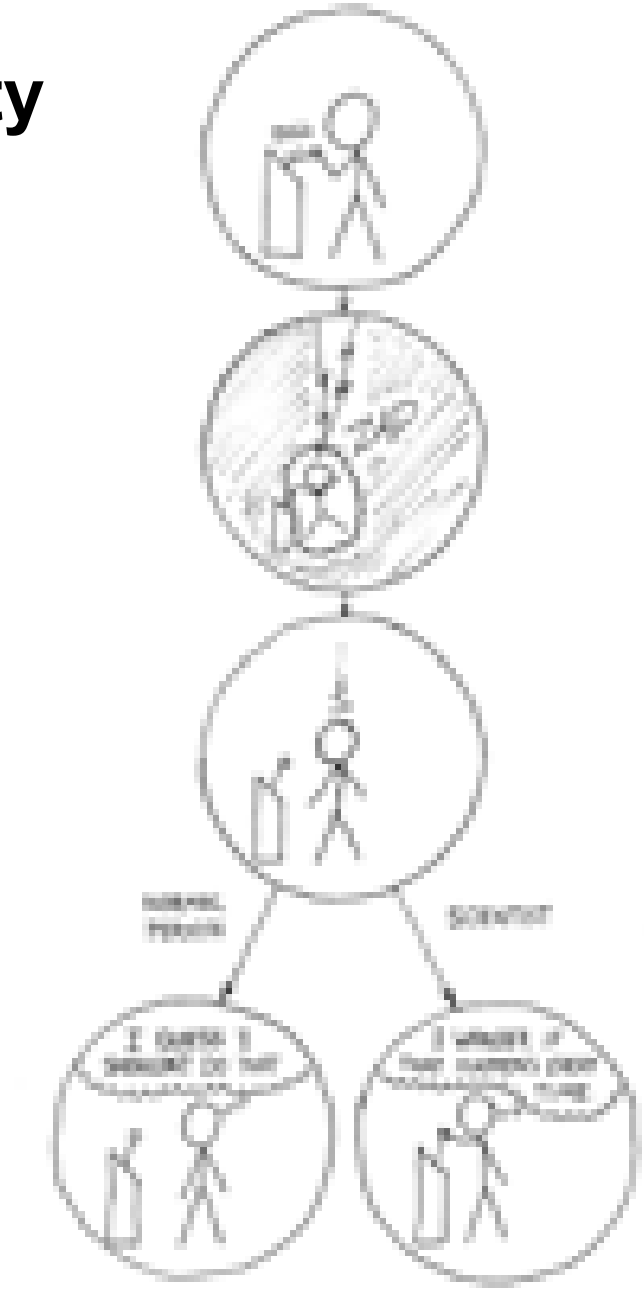
- Why should we cite data?
  - Prevent Scientific misconduct (“extrinsic”) ?
  - Give credit (“altruistic”) ?
  - Show solid basis (“egoistic”) ?
  - Enable reproducibility, re-use (extrinsic + altruistic + egoistic) ?



Reproducibility

- Reproducibility is core to the scientific method
- Focus not on misconduct – but on complexity and the will to produce good work

- Should be easy
  - Get the code, compile, run, …
  - Why is it difficult?



https://xkcd.com/242/

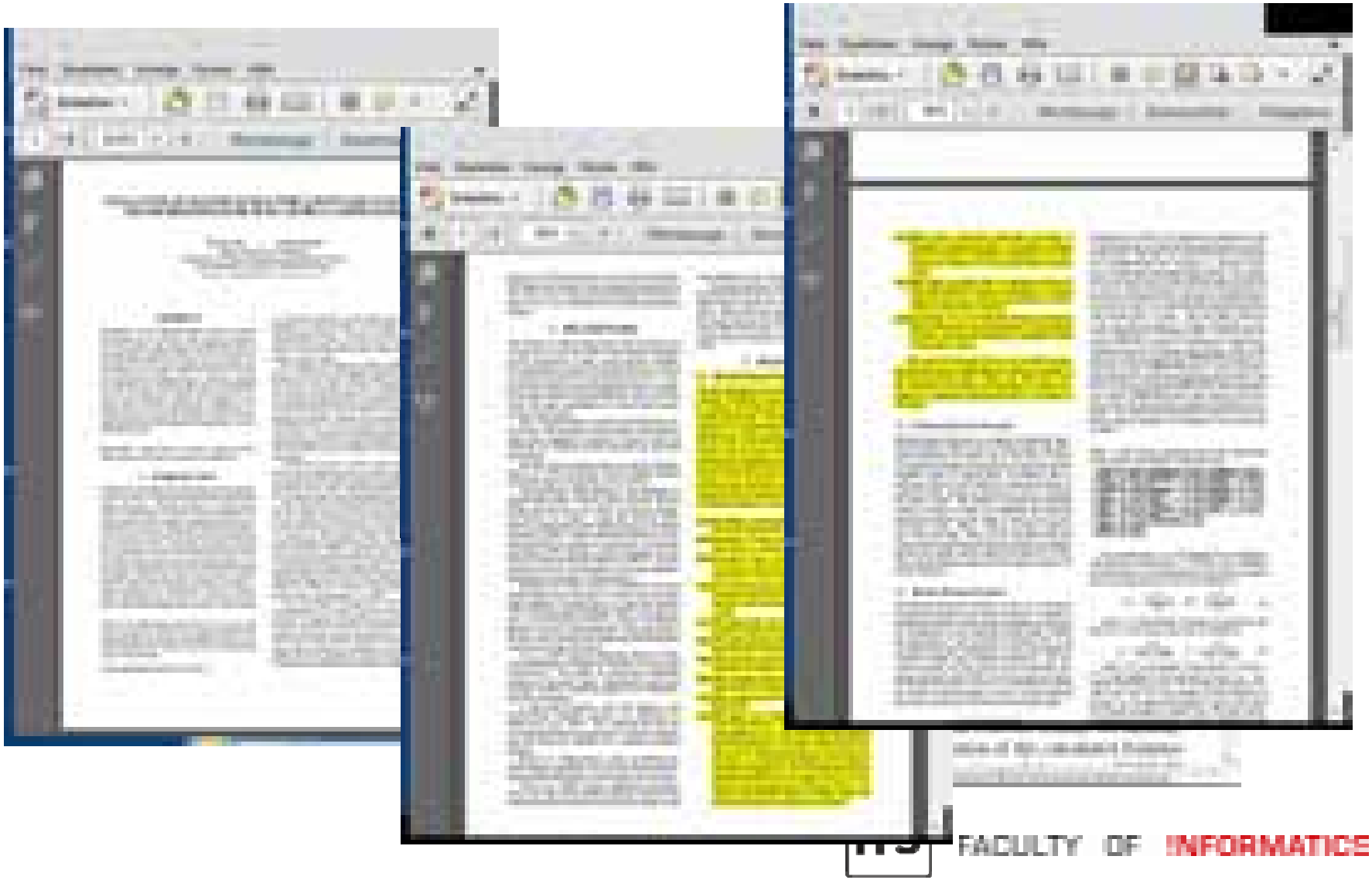
Challenges in Reproducibility

http://www.plosone.org/article/info%3Adoi%2F10.1371%2Fjournal.pone.0038234

ifs FACULTY OF INFORMATICS

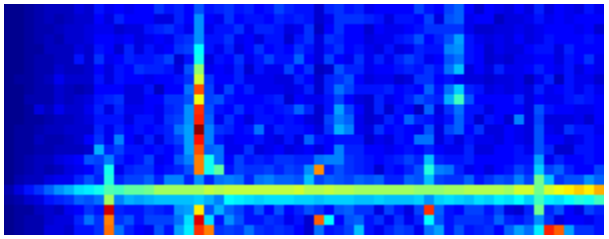
Challenges in Reproducibility

- Excursion: Scientific Processes

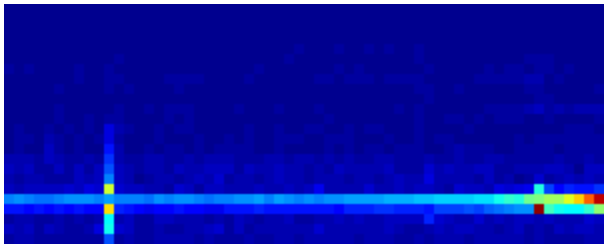


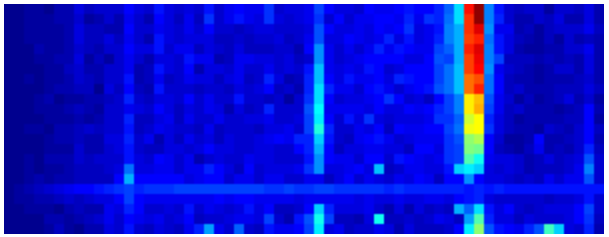
Challenges in Reproducibility

- Excursion: scientific processes

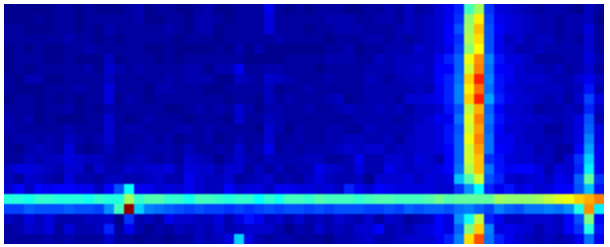


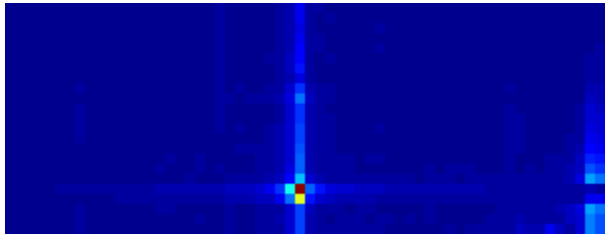
set1_freq440Hz_Am11.0Hz



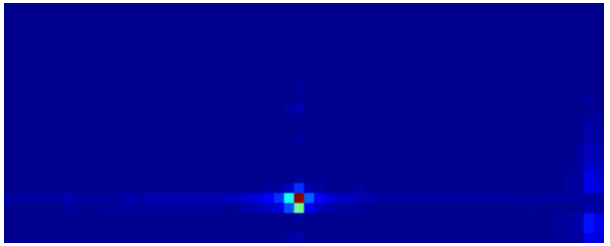


set1_freq440Hz_Am12.0Hz





set1_freq440Hz_Am05.5Hz

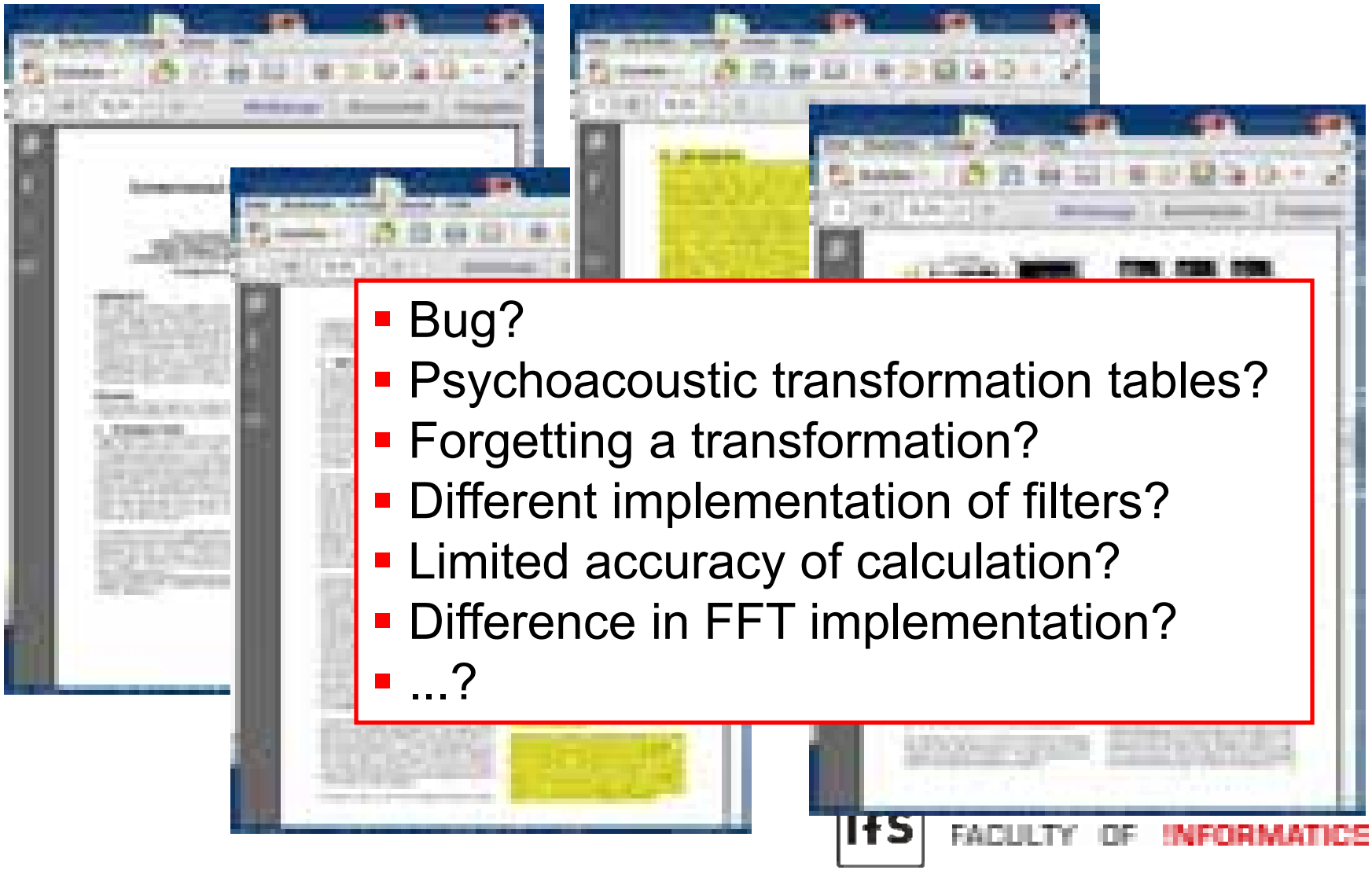


Java

Matlab

Challenges in Reproducibility

- Excursion: Scientific Processes

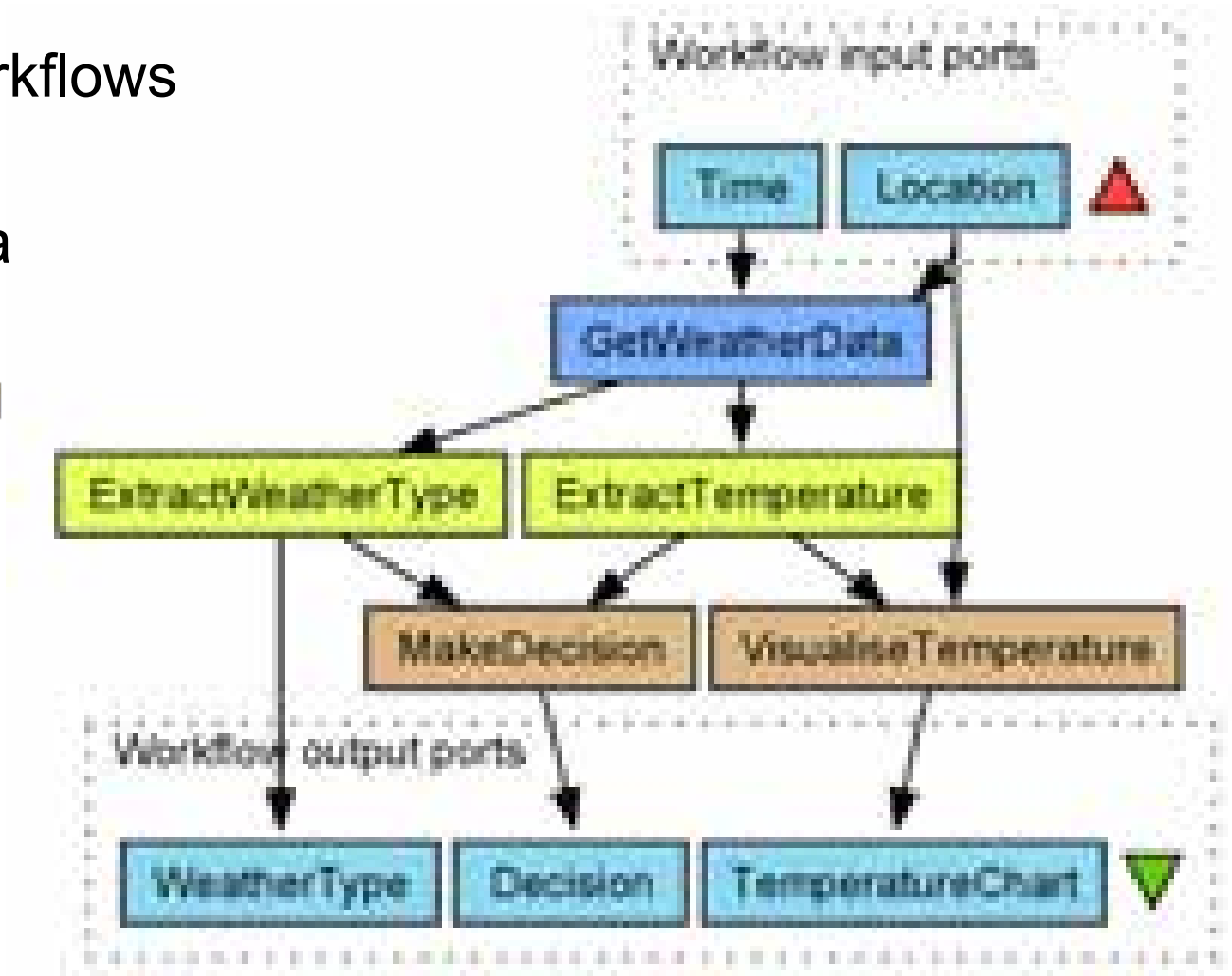


- Bug?
- Psychoacoustic transformation tables?
- Forgetting a transformation?
- Different implementation of filters?
- Limited accuracy of calculation?
- Difference in FFT implementation?
- ...?

Challenges in Reproducibility

- Workflows

Taverna

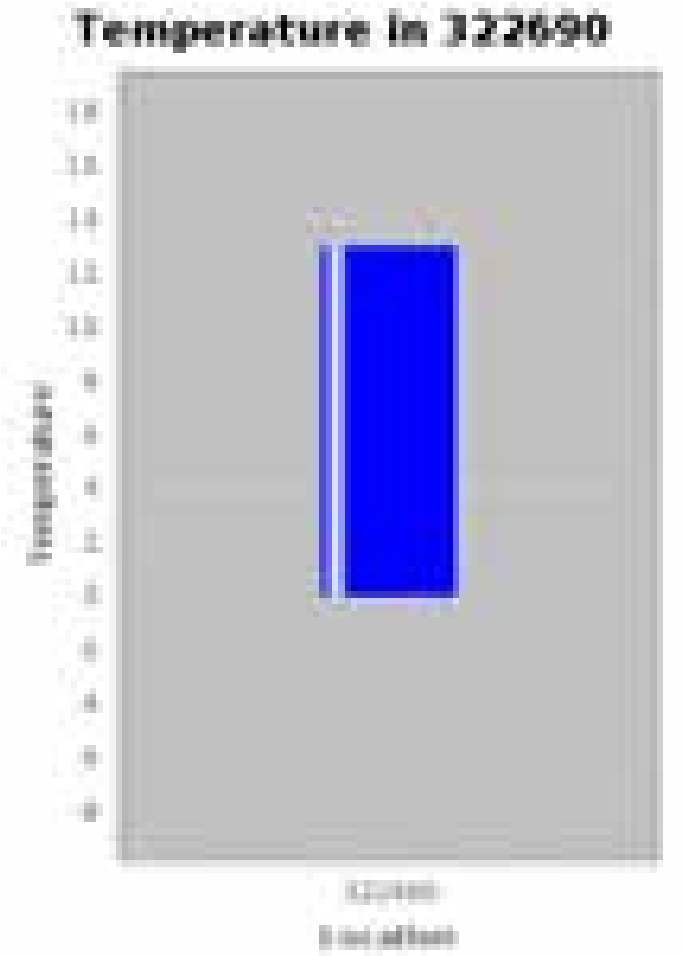


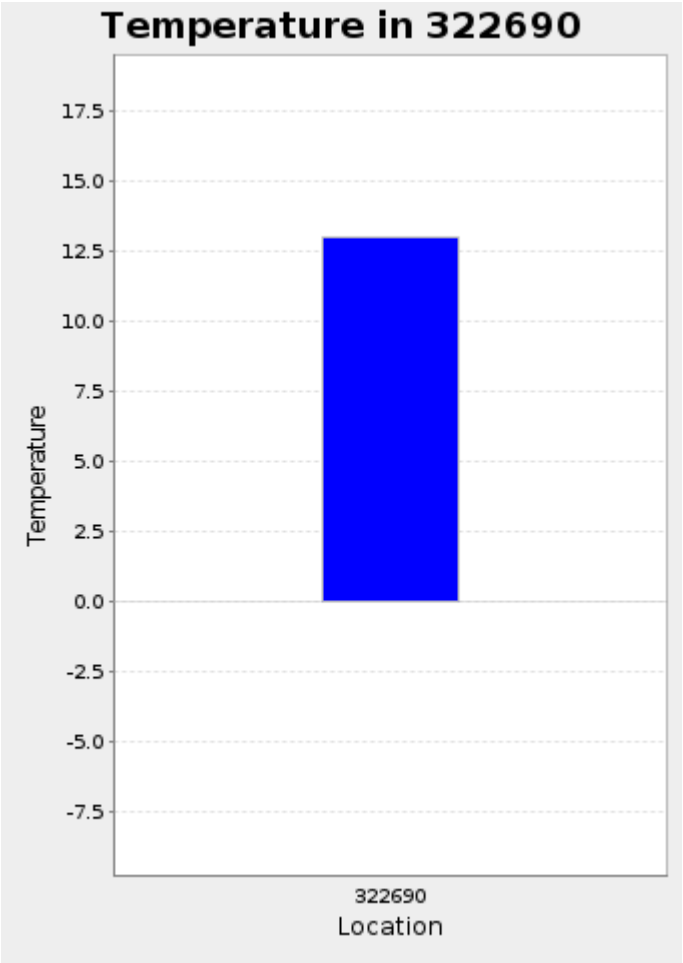
Activiti

Kepler

ifs FACULTY OF INFORMATICS

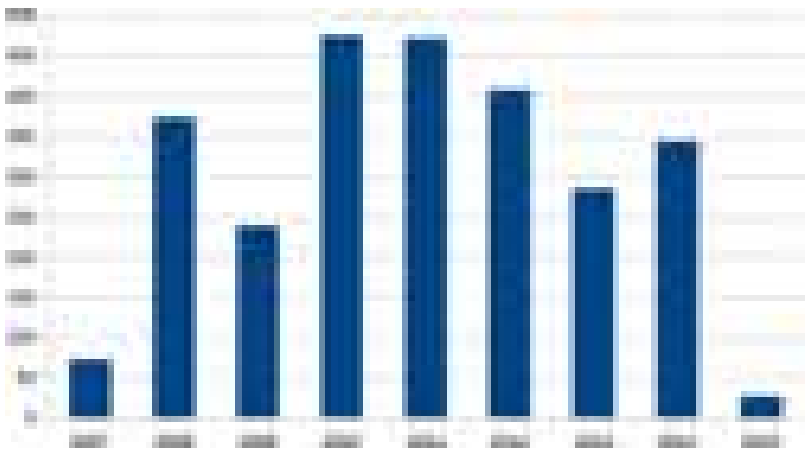
Challenges in Reproducibility





Challenges in Reproducibility

- Large scale quantitative analysis
- Obtain workflows from MyExperiments.org
  - March 2015: almost 2.700 WFs (approx. 300-400/year)
  - Focus on Taverna 2 WFs: 1.443 WFs
  - Published by authors ➔ should be „better quality“
- Try to re-execute the workflows
  - Record data on the reasons for failure along
- Analyse the most common reasons for failures



Workflow Engine	%
Taverna 2	54.7
Taverna 1	20.9
RapidMiner	10.0
Galaxy	2.0
Others	12.4

FACULTY OF INFORMATICS

Challenges in Reproducibility

Re-Execution results

- Majority of workflows fails
- Only 23.6 % are successfully executed
  - No analysis yet on correctness of results…

Processor	# WFs
Original Data Set	1443
- Missing input values	526
- Disabled processors (WSDL services)	180
- Not executable in test environment	6
Final Data Set	731

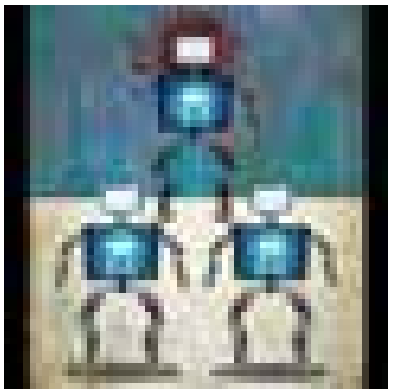
Processor	# WFs	% WFs
Not terminated >48hours	6	0.8
Execution failed	384	52.5
Execution successful	341	46.6

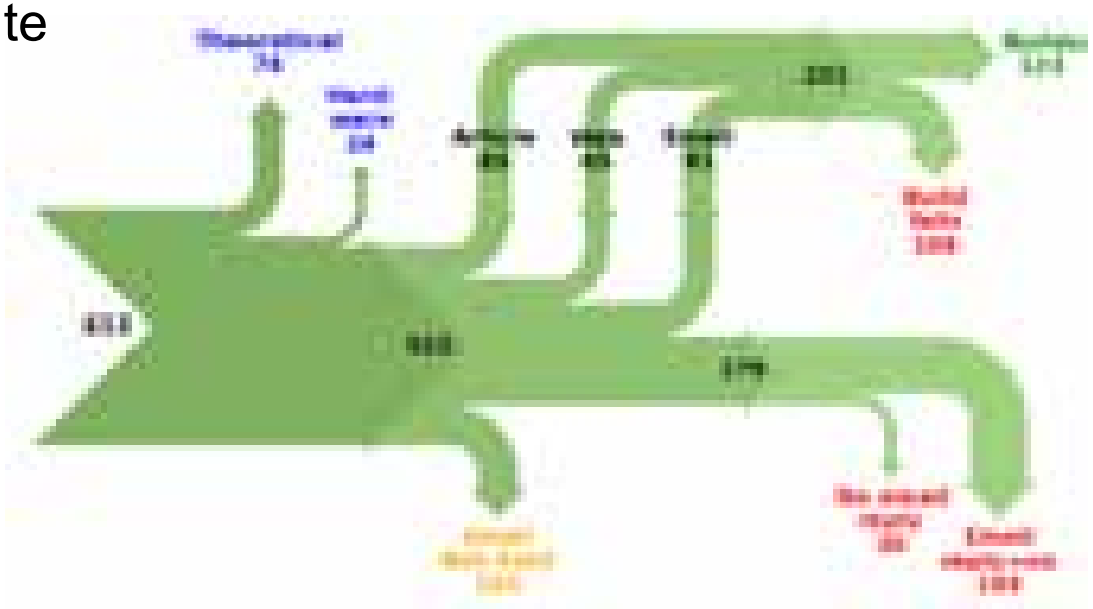
Processor	# WFs
REST unavailable	4
REST unauthenticated	5
Other unauthenticated	40
Missing Resources	14
Tool unavailable	19

Rudolf Mayer, Andreas Rauber, “A Quantitative Study on the Re-executability of Publicly Shared Scientific Workflows”, 11th IEEE Intl. Conference on e-Science, 2015.

Challenges in Reproducibility

- 613 papers in 8 ACM conferences
- Process
  - download paper and classify
  - search for a link to code (paper, web, email twice)
  - download code
  - build and execute



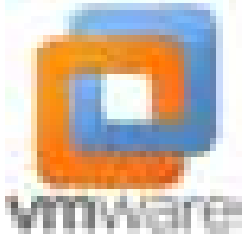


Christian Collberg and Todd Proebsting. “Repeatability in Computer Systems Research,” CACM 59(3):62-69.2016

Reproducibility – solved! (?)

- Provide source code, parameters, data, …
- Wrap it up in a container/virtual machine, …

LXC



…

- Why do we want reproducibility?
- Which levels or reproducibility are there?
- What do we gain by different levels of reproducibility?
- A simple “re-run” is usually not enough
  – otherwise, video would be sufficient….

Types of Reproducibility

- The **PRIMAD Model**[1]: which attributes can we "prime"?
  - Data
    - Parameters
    - Input data
  - Platform
  - Implementation
  - Method
  - Research Objective
  - Actors
- What do we gain by "priming" one or the other?

[1] Juliana Freire, Norbert Fuhr, and Andreas Rauber. Reproducibility of Data-Oriented Experiments in eScience. Dagstuhl Reports, 6(1), 2016.

Types of Reproducibility and Gains

Label	Data		Platform / Stack	Implementation	Method	Research Objective	Actor	Gain
	Parameters	Raw Data						
Repeat	–	+	–	–	–	–		Determinism
Param. Sweep	x	–	–	–	–	–		Robustness / Sensitivity
Generalize	(x)	x	–	–	–	–		Applicability across different settings
Port	–	–	x	–	–	–		Portability across platforms, flexibility
Re-code	–	–	(x)	x	–	–		Correctness of implementation, flexibility, adoption, efficiency
Validate	(x)	(x)	(x)	(x)	x	–		Correctness of hypothesis, validation via different approach
Re-use	–	–	–	–	–	x		Apply code in different settings, Re-purpose
Independent x (orthogonal)							x	Sufficiency of information, independent verification

Reproducibility Papers

- Aim for reproducibility: for one's own sake – and as Chairs of conference tracks, editor, reviewer, supervisor, …
  - Review of reproducibility of submitted work (material provided)
  - Encouraging reproducibility studies
  - (Messages to stakeholders in Dagstuhl Report)
- Consistency of results, not identity!
- Reproducibility studies and papers
  - Not just re-running code / a virtual machine
  - When is a reproducibility paper worth the effort / worth being published?

Challenges in Reproducibility

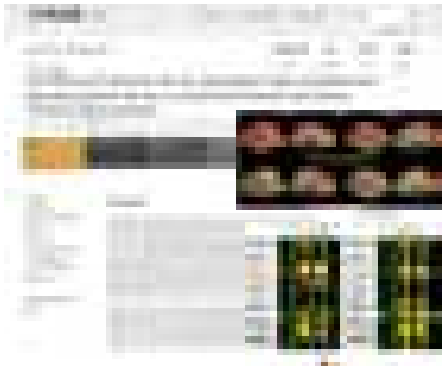
Peer Review and Verification

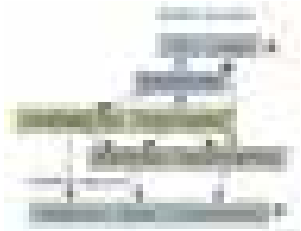
- Peer review is an established process
  - Focused on publications mainly
  - Hardly any data quality reviews
  - Even less reproducibility studies
- Reproducing or replicating experiments is not considered original research
  - No recognition
  - No money
  - A lot of work
- Encourage reproducibility studies
- Needed beyond science!

Challenges in Reproducibility

In a nutshell – and another aspect of reproducibility:

Source: xkcd





ifs FACULTY OF INFORMATICS

Why to cite data?

- Data is the basis for almost everything
  - eScience, digital humanities,
  - Industry 4.0
  - Driving policies, society, …

- Why should we cite data?
  - Prevent Scientific misconduct (“extrinsic”) ?
  - Give credit (“altruistic”) ?
  - Show solid basis (“egoistic”) ?
  - Enable reproducibility, re-use (extrinsic + altruistic + egoistic) ?
  - Because it’s what you do if you do good work, speeding up the process of scientific discovery, efficiency! (“intrinsic”)

Why to cite data?

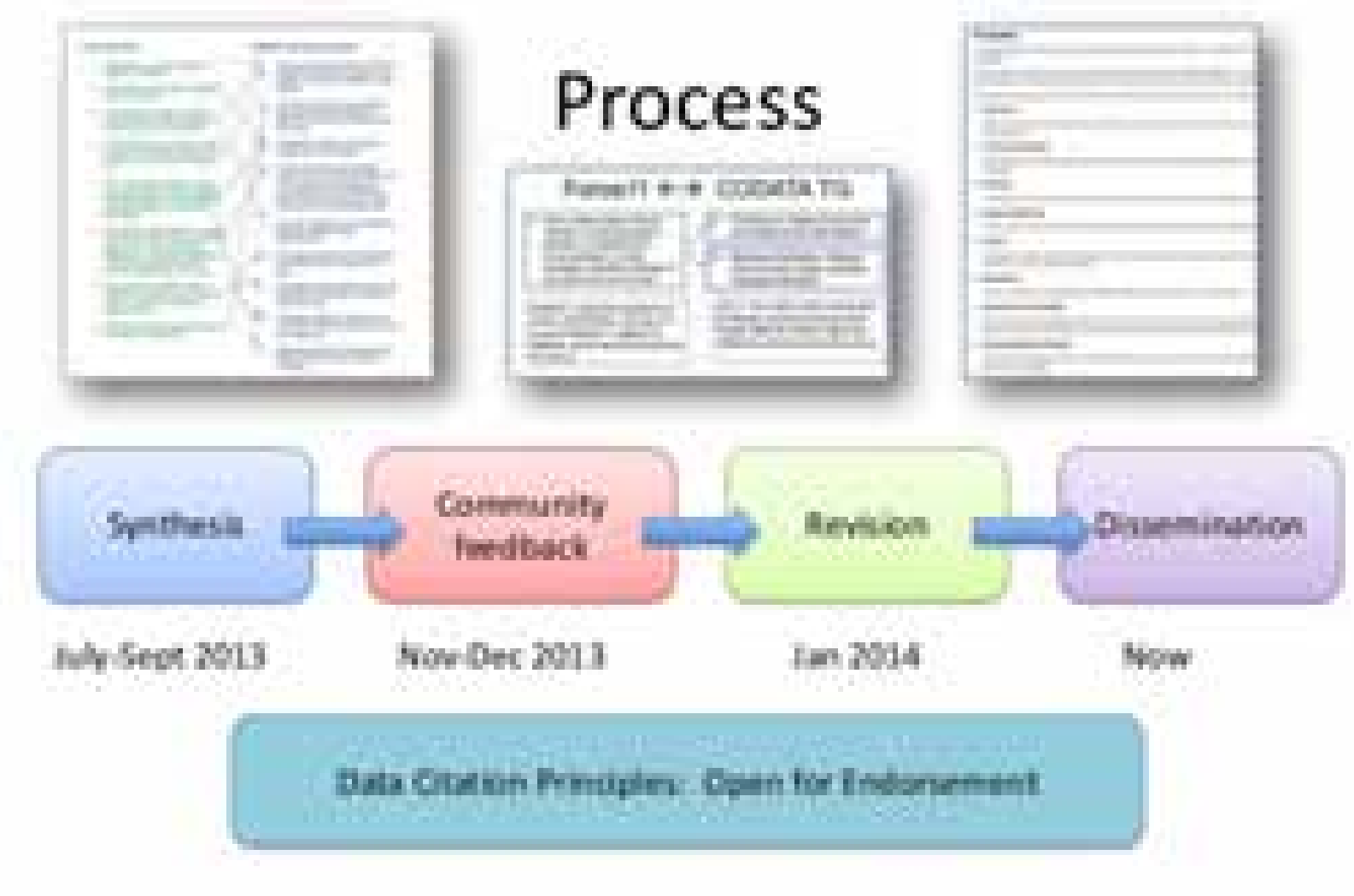
- It's what you do! – Lots of benefits
  - Makes live easier because you can build on a solid foundation
  - Speeds up the process because you can re-use existing stuff
  - Helps avoiding / detecting mistakes, improves quality, comparability
  - Reuse increases citations, visibility ("currency")

- But:
  - To achieve this it must be easy, straightforward, "automatic"
  - Citing Papers is easy…
  - …what about data?
    (more about this later… first: "we should just do it")

Joint Declaration of Data Citation Principles

- 8 Principles created by the Data Citation Synthesis Group
- https://www.force11.org/datacitation
- The Data Citation Principles cover purpose, function and attributes of citations
- Goal: Encourage communities to develop practices and tools that embody uniform data citation principles

Joint Declaration of Data Citation Principles



Joint Declaration of Data Citation Principles (cont'd)

1) **Importance**

Data should be considered legitimate, citable products of research. Data citations should be accorded the same importance as publications.

2) **Credit and Attribution**

Data citations should facilitate giving credit and normative and legal attribution to all contributors to the data.

Joint Declaration of Data Citation Principles (cont'd)

3) Evidence

Whenever and wherever a claim relies upon data, the corresponding data should be cited.

4) Unique Identification

A data citation should include a persistent method for identification that is machine actionable, globally unique, and widely used by a community.

Joint Declaration of Data Citation Principles (cont'd)

5) Access

Data citations should facilitate access to the data themselves and to such associated metadata, documentation, code, and other materials, as are necessary for both humans and machines to make informed use of the referenced data.

6) Persistence

Unique identifiers, and metadata describing the data, and its disposition, should persist - even beyond the lifespan of the data they describe.

Joint Declaration of Data Citation Principles (cont'd)

7) Specificity and Verifiability

Data citations should facilitate identification of, access to, and verification of the specific data that support a claim. Citations or citation metadata should include information about provenance and fixity sufficient to facilitate verifying that the specific time slice, version and/or granular portion of data retrieved subsequently is the same as was originally cited.

Joint Declaration of Data Citation Principles (cont'd)

8) Interoperability and flexibility

Data citation methods should be sufficiently flexible to accommodate the variant practices among communities, but should not differ so much that they compromise interoperability of data citation practices across communities.

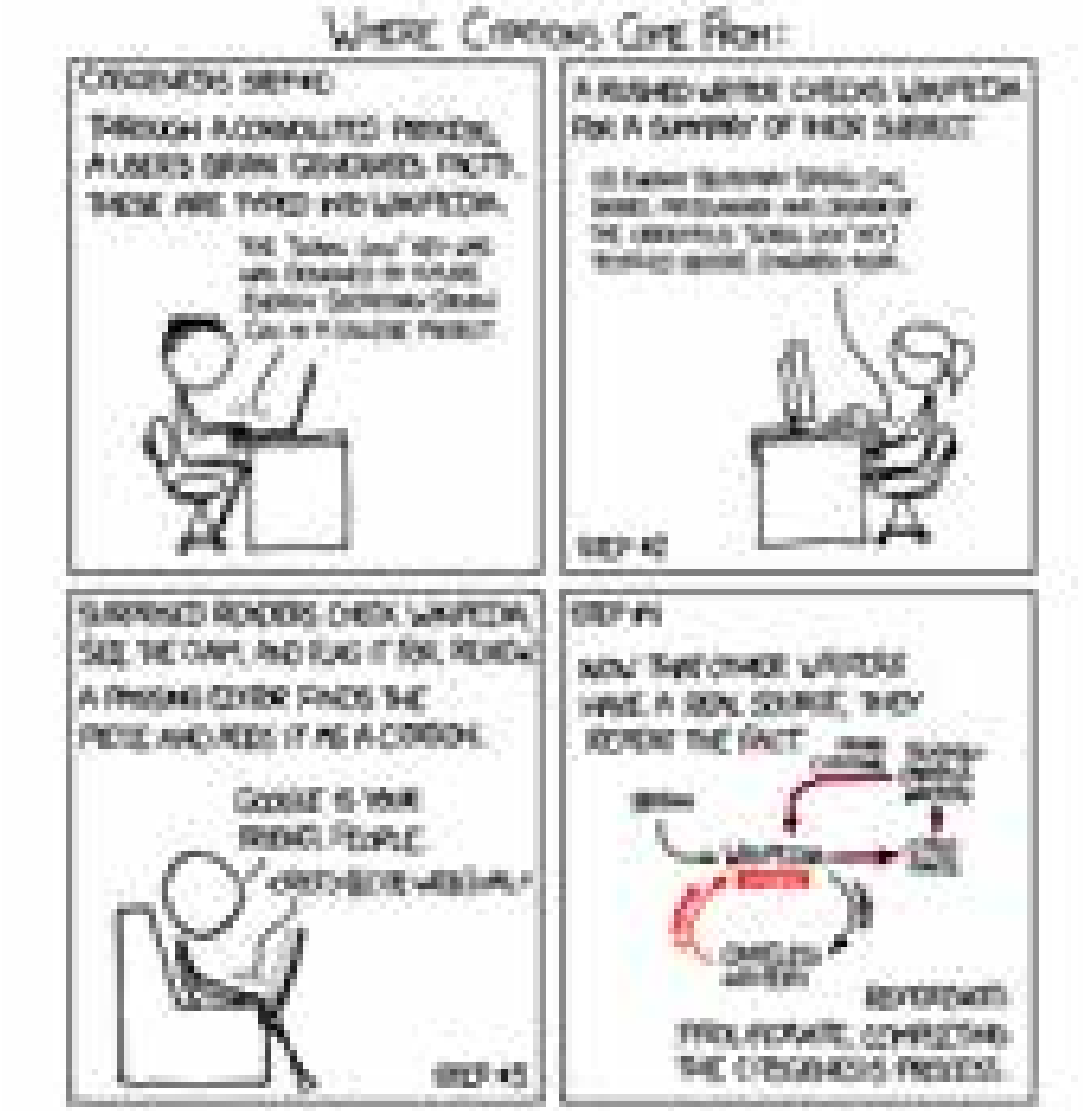
Joint Declaration of Data Citation Principles (cont'd)

- **Glossary**

  https://www.force11.org/node/4770

Benefits of Citation

- Identification
- Documentation
- Context
- Impact
- Transparency
- Reproducibility
- Reuse



https://xkcd.com/978/

Outline

- Why should we want to cite data?
- What are the challenges in data identification and citation?
- How should we do it, according to the RDA WG?
- Who is doing it so far, and how?
- Summary

Why to cite data?

- It's what you do! – Lots of benefits
  - Makes live easier because you can build on a solid foundation
  - Speeds up the process because you can re-use existing stuff
  - Helps avoiding / detecting mistakes, improves quality
  - Reuse increases citations, visibility, currency

Deja-vue

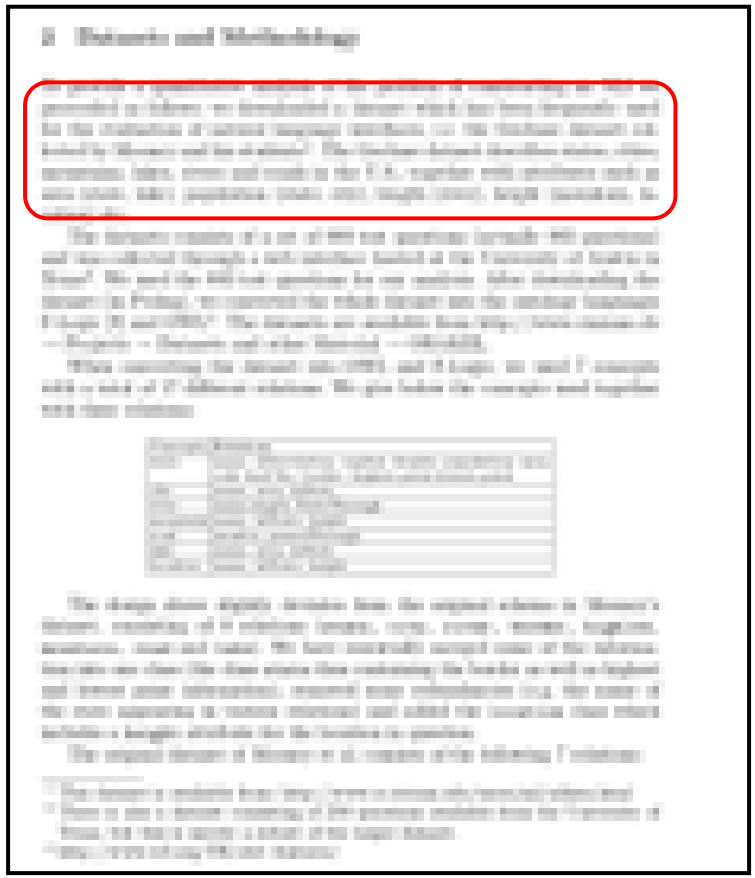
- But:
  - To achieve this it must be easy, straightforward, "automatic"
  - Citing Papers is easy…
  - …what about data?
    (more about this later… first: "we should just do it")

How to cite data?

- Referencing research papers is well established

Example: Web Page Download

- Example: Web-page download



Example: Web Page Download

To provide a quantitative analysis of the problem of constructing an NLI we proceeded as follows: we downloaded a dataset which has been frequently used for the evaluation of natural language interfaces, i.e. the Geoquery dataset collected by Mooney and his students[1]. The Geoquery dataset describes states, cities, mountains, lakes, rivers and roads in the U.S., together with attributes such as area (state, lake), population (state, city), length (river), height (mountain, location) etc.

The dataset consists of a set of 880 test questions (actually 886 questions) and was collected through a web interface hosted at the University of Texas[2]. We used the 883 test questions for our analysis. After downloading the dataset (in Prolog), we converted the whole dataset into the ontology languages F-Logic [9] and OWL[3]. The datasets are available from http://www.cimianet.de — Projects — Datasets and other Material — OBAKEL.

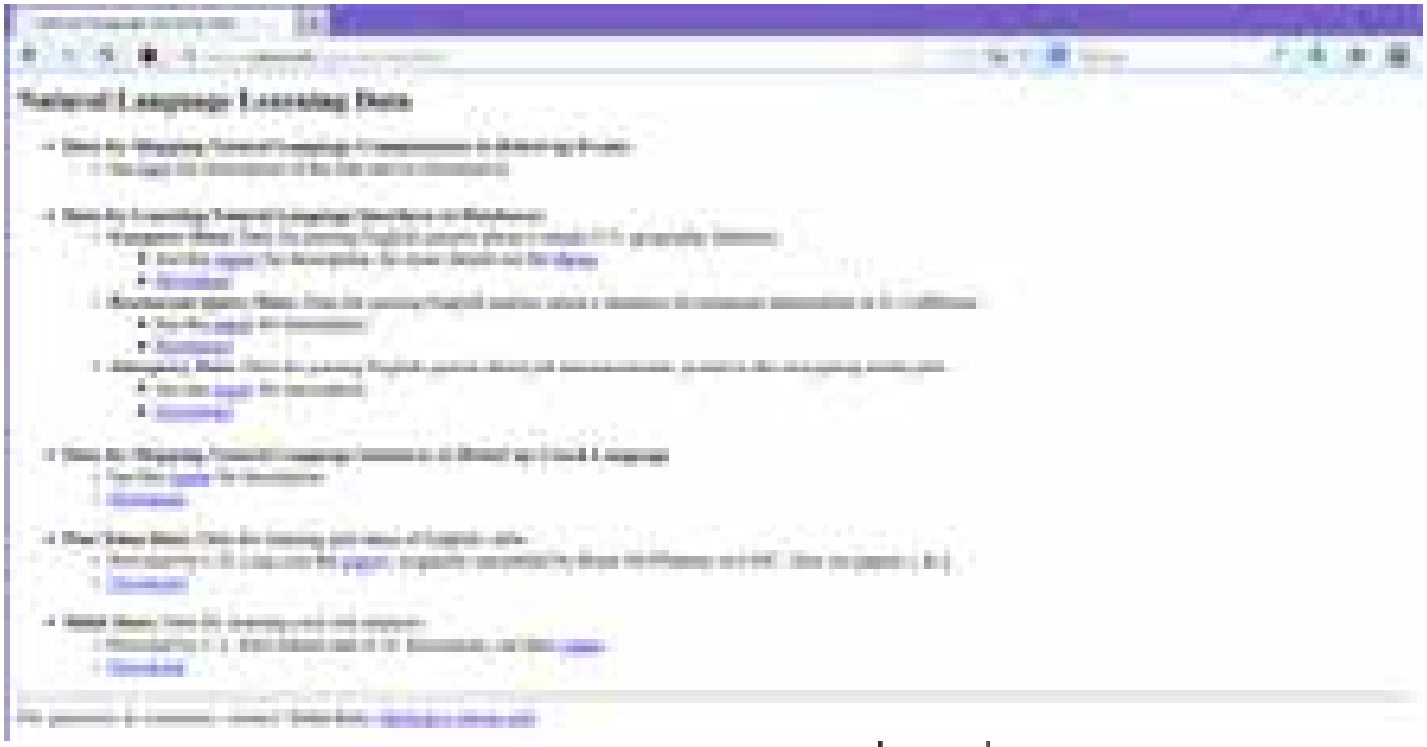
[1] This dataset is available from: http://www.cs.utexas.edu/users/ml/nldata.html

[2] There is also a dataset consisting of 250 questions available from the University of Texas, but this is merely a subset of the larger dataset.

[3] http://www.w3.org/TR/owl-features/

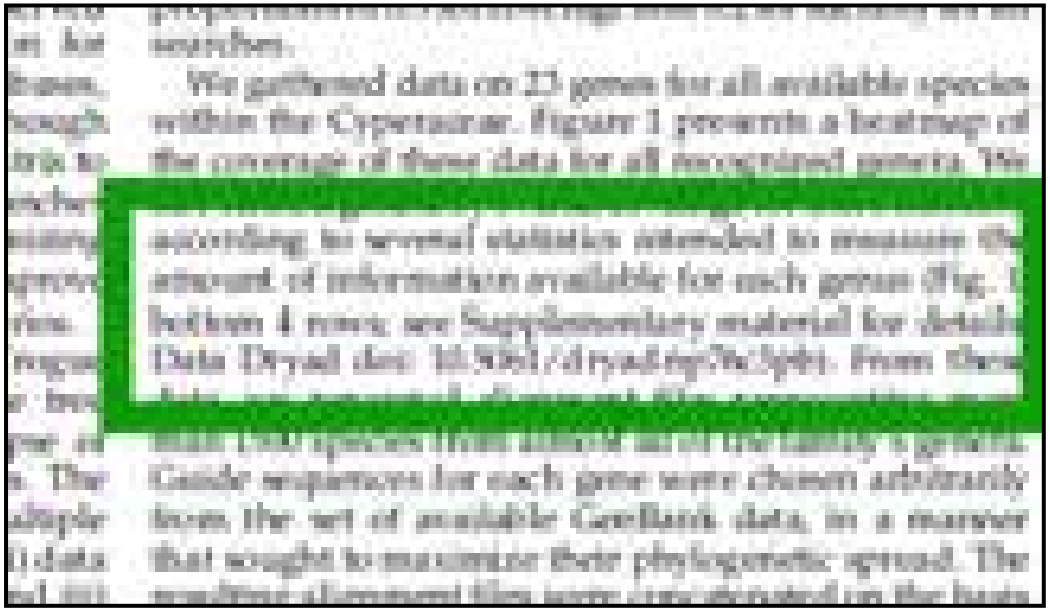
Example: Web Page Download

1 This dataset is available from: http://www.cs.utexas.edu/users/ml/nldata.html
2 There is also a dataset consisting of 250 questions available from the University of Texas, but this is merely a subset of the larger dataset.
3 http://www.w3.org/TR/webarch/



Example: Sharing Platform

- Example: Data sharing platforms



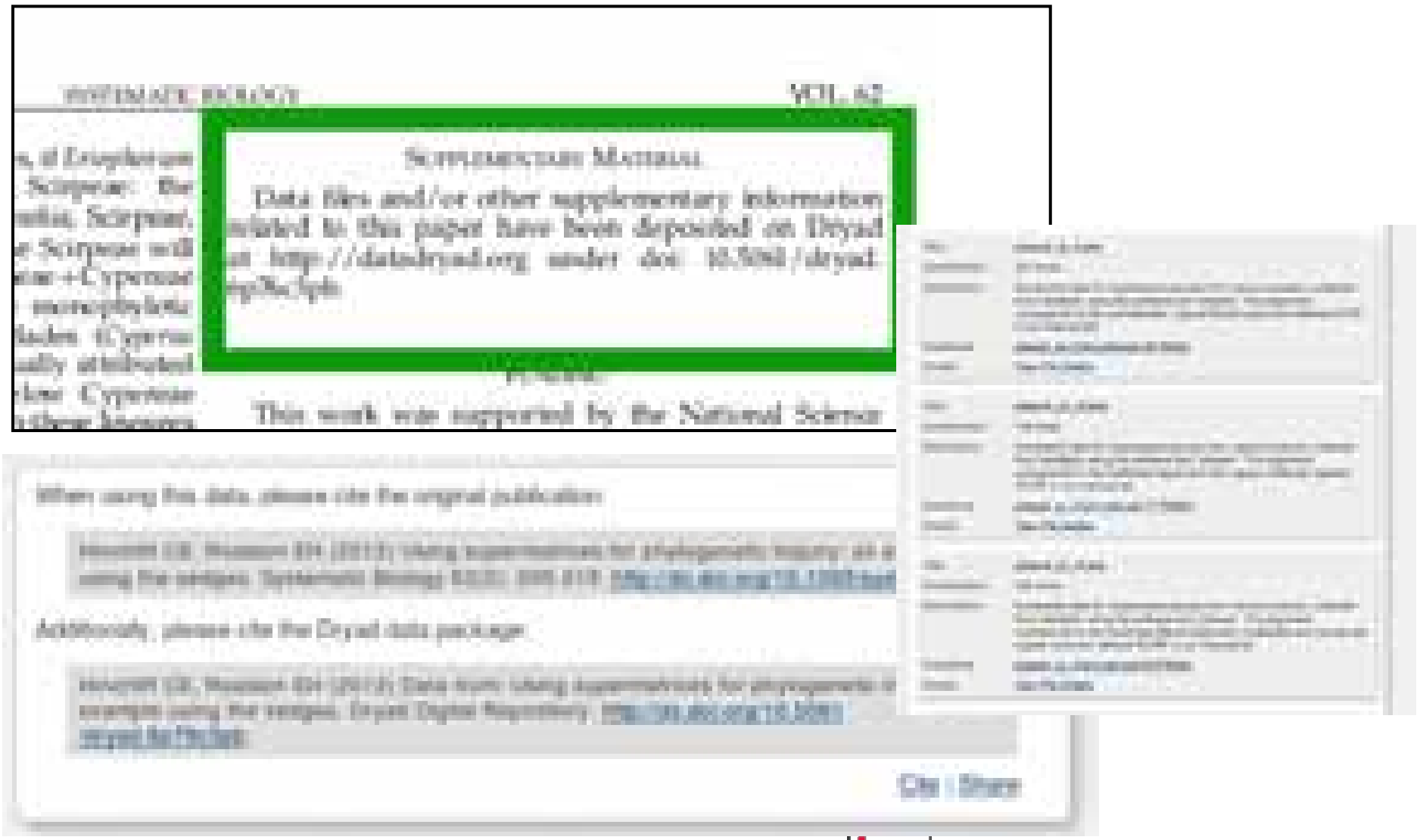
http://datadryad.org/resource/doi:10.5061/dryad.6p76c3pb/1

Example: Sharing Platform

- Example: Data sharing platforms

http://datadryad.org/resource/doi:10.5061/dryad.6p76c3pb/1

Example: Sharing Platform



http://datadryad.org/resource/doi:10.5061/dryad.6p76c3pb/1

Subset Citation in Papers

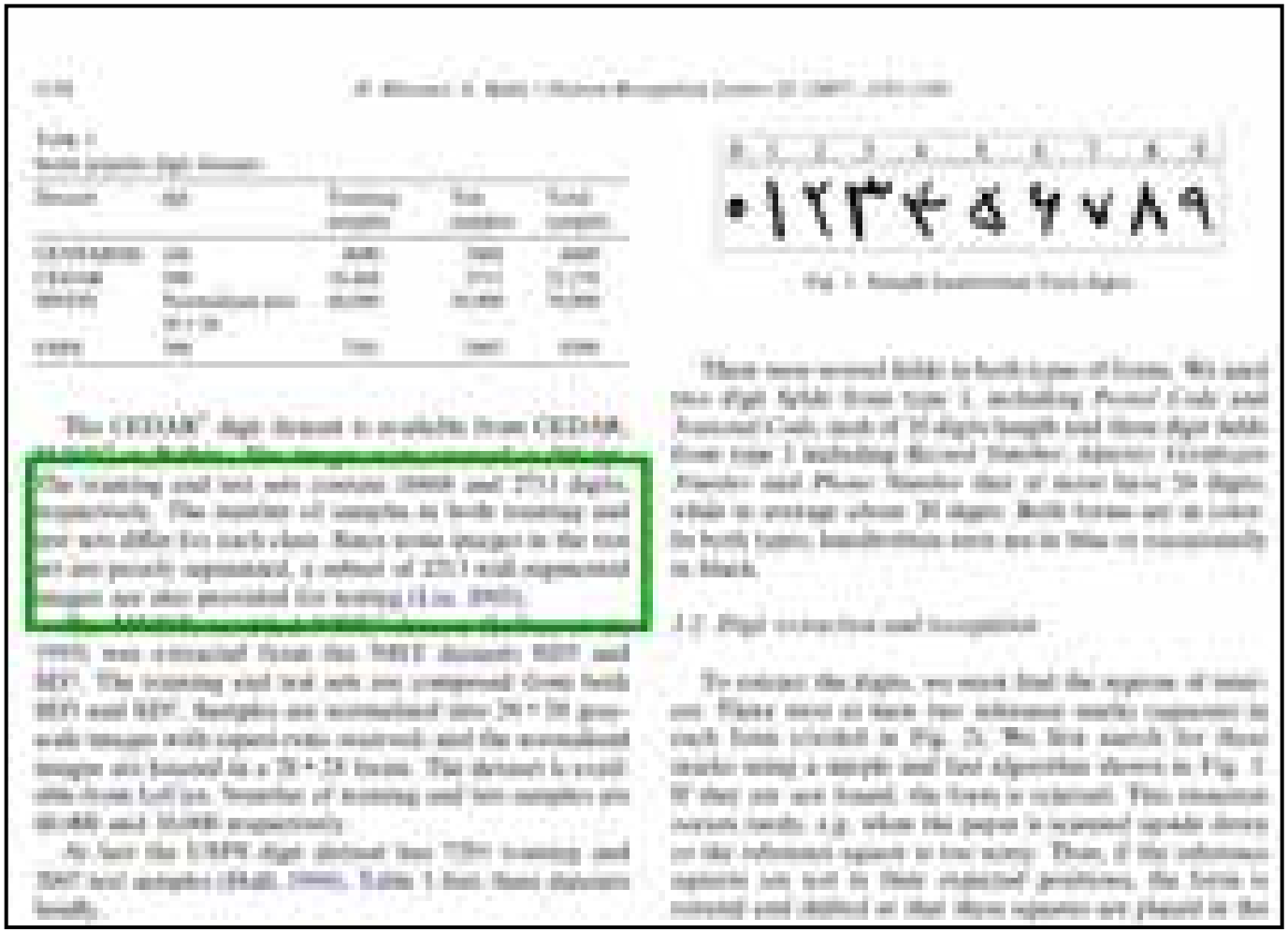
- Example: Subsets of data

Introducing a very large dataset of handwritten Farsi digits and a study on their varieties

Khosravi, Hossein, and Ehsanollah Kabir. "Introducing a very large dataset of handwritten Farsi digits and a study on their varieties." Pattern Recognition Letters 28.10 (2007): 1133-1141.

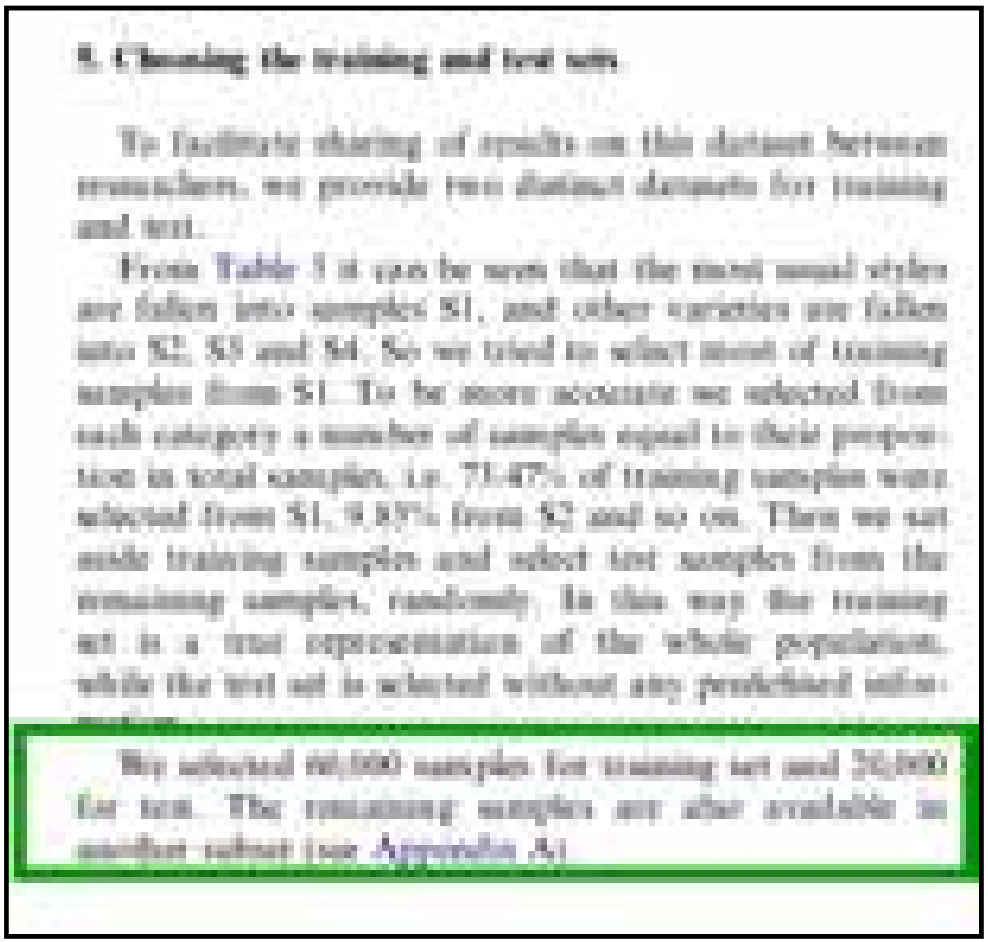
Subset Citation in Papers

- Example: Subsets of data



Subset Citation in Papers

- Example: Subsets of data



9. Choosing the training and test sets

To facilitate sharing of results on this dataset between researchers, we provide two distinct datasets for training and test.

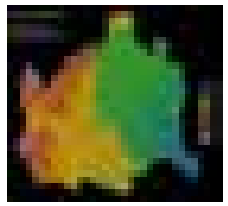
From Table 1 it can be seen that the most usual styles are fallen into samples S1, and other varieties are fallen into S2, S3 and S4. So we tried to select most of training samples from S1. To be more accurate we selected from each category a number of samples equal to their proportion in total samples, i.e. 71.47% of training samples were selected from S1, 9.83% from S2 and so on. Then we set aside training samples and select test samples from the remaining samples, randomly. In this way the training set is a true representation of the whole population, while the test set is selected without any predefined information.

We selected 60,000 samples for training set and 20,000 for test. The remaining samples are also available in another subset (see Appendix A).

Khosravi, Hossein, and Ehsanollah Kabir. "Introducing a very large dataset of handwritten Farsi digits and a study on their varieties." Pattern Recognition Letters 28.10 (2007): 1133-1141.

Motivation

- Research data is fundamental for science/industry/…
  - Data serves as input for workflows and experiments
  - Data is the source for graphs and visualisations in publications
  - Decisions are based on data
- Data is needed for Reproducibility
  - Repeat experiments
  - Verify / compare results
- Need to provide specific data set
  - Service for data repositories

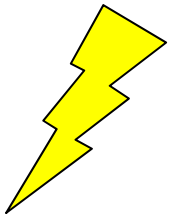


https://commons.wikimedia.org/w/index.php?curid=30978545

1. Put data in data repository,
2. Assign PID (DOI, Ark, URI, …)
3. Make is accessible
   → done!?

Identification of Dynamic Data

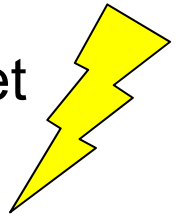
- Usually, datasets have to be static
  - Fixed set of data, no changes:
    no corrections to errors, no new data being added
- But: (research) data is **dynamic**
  - Adding new data, correcting errors, enhancing data quality, …
  - Changes sometimes highly dynamic, at irregular intervals
- Current 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 a specific point in time**

Granularity of Subsets

- What about the **granularity** of data to be identified?
  - Enormous amounts of CSV data
  - Researchers use specific subsets of data
  - Need to identify precisely the subset used
- Current approaches
  - Storing a copy of subset as used in study -> scalability
  - Citing entire dataset, providing textual description of subset -> imprecise (ambiguity)
  - Storing list of record identifiers in subset -> scalability, not for arbitrary subsets (e.g. when not entire record selected)



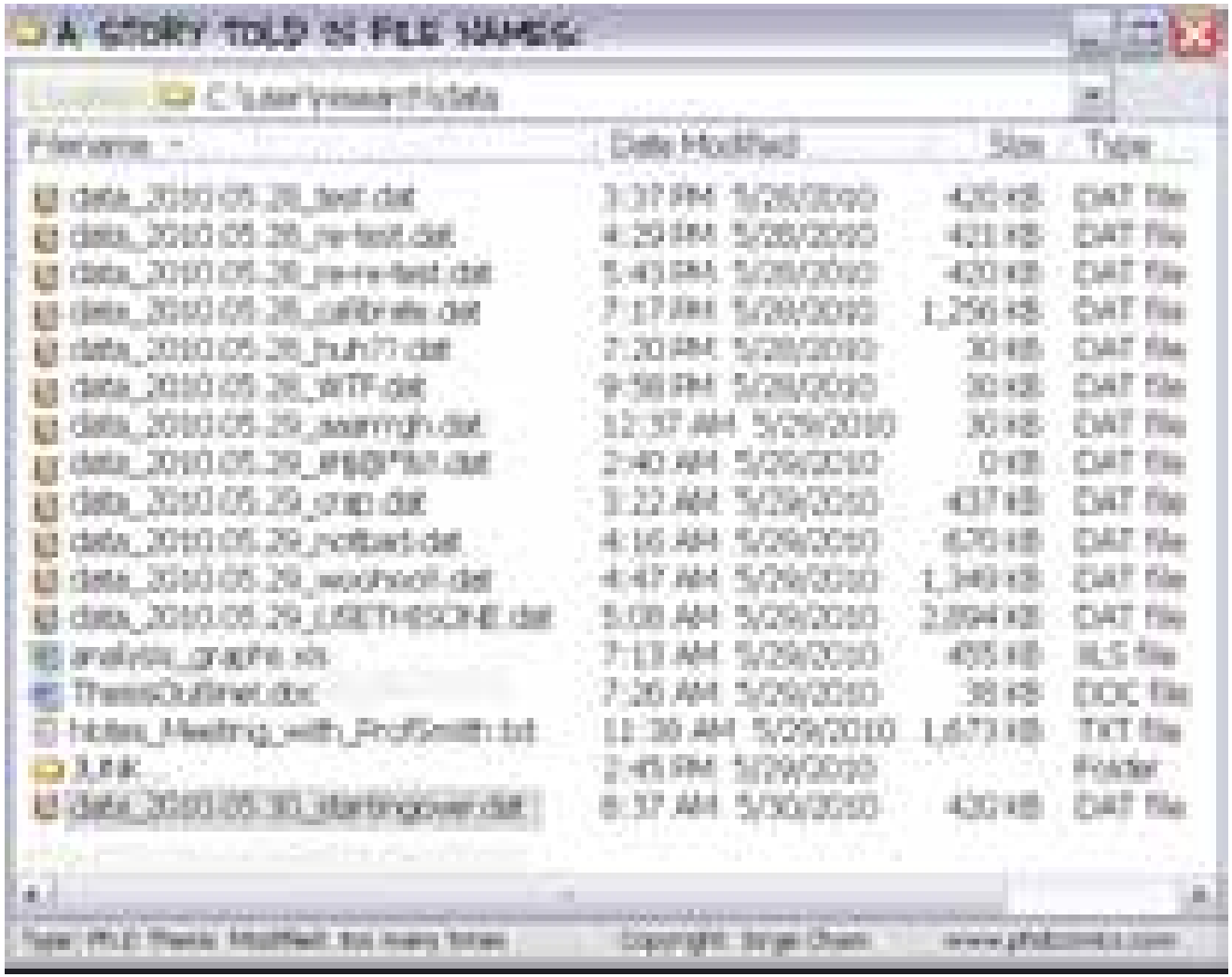
- Would like to be able to identify precisely the **subset of (dynamic) data used** in a process

Data Citation – Requirements

- Dynamic data
  - corrections, additions, …
- 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
- 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!

What we do NOT want…

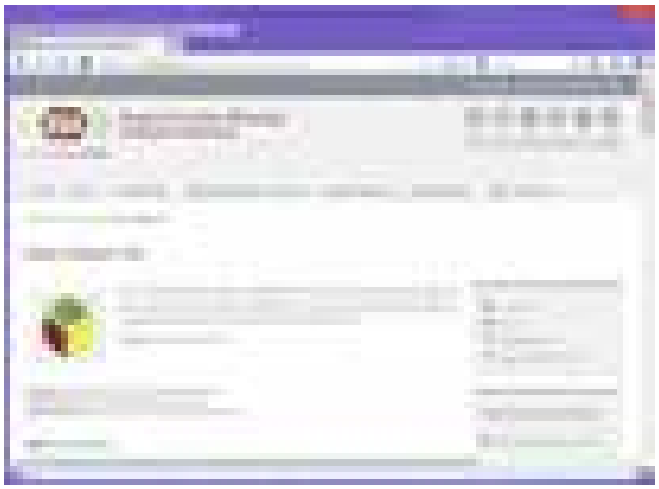
- Common approaches to data management…
  (from PhD Comics: A Story Told in File Names, 28.5.2010)
  Source: http://www.phdcomics.com/comics.php?f=1323



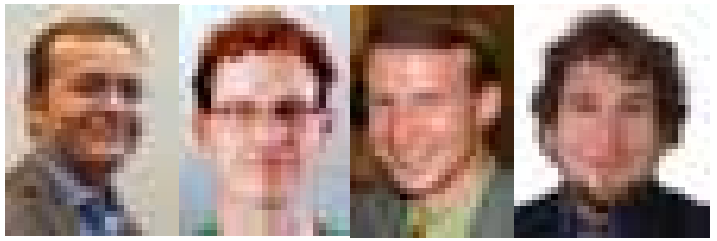
Outline

- Why should we want to cite data?
- What are the challenges in data identification and citation?
- How should we do it, according to the RDA WG?
- Who is doing it so far, and how?
- Summary

RDA WG Data Citation



- Research Data Alliance
- WG on **Data Citation: Making Dynamic Data Citeable**
- March 2014 – September 2015
  - Concentrating on the problems of **large, dynamic (changing) datasets**
- Final version presented Sep 2015 at P7 in Paris, France
- Endorsed September 2016 at P8 in Denver, CO
- Since then: supporting adopters



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

ifs FACULTY OF INFORMATICS

RDA WGDC - Solution

- **We have**
  - Data & some means of access („query“)

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 → versioned (history, with time-stamps)

Researcher creates working-set via some interface:

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)

Researcher creates working-set via some interface:

2. Access → **store & assign PID to "QUERY"**, enhanced with
   - **Time-stamping** for re-execution against versioned DB
   - **Re-writing** for normalization, unique-sort, mapping to history
   - **Hashing** result-set: verifying identity/correctness

   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

Data Citation – Deployment

- Researcher 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

Data Citation – Deployment

- Note: query string provides excellent provenance information on the data set!
- ubset of data
- er gets
  - Data (package, acce s API, …)
  - PID (e.g. DOI) (Que is time-stamped and stored)
  - Hash value compute over the data for local storage
  - Recommended citati n text (e.g. BibTeX)
- PID resolves to land g 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

Data Citation – Deployment

- Note: query string provides excellent provenance information on the data set!
- ubset of data
- er gets
  - Data (pac
  - PID (e.g.
  - Hash valu
  - Recommended citat[illegible] text (e.g. BibTeX)
- This is an important advantage over traditional approaches relying on, e.g. storing a list of identifiers/DB dump!!!
- PID resolves to land[illegible]g 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

Data Citation – Deployment

- Note: query string provides excellent provenance information on the data set!
- ubset of data
- er gets
  - Data (pac
  - PID (e.g.
  - Hash valu
  - Recommended citat[illegible] text (e.g. BibTeX)
- This is an important advantage over traditional approaches relying on, e.g. storing a list of identifiers/DB dump!!!
- PID resolves
  - Provides det
  - Option to ret
- Identify which parts of the data are used. If data changes, identify which queries (studies) are affected
- Upon activating PID associated w[illegible]n a data citation
  - Query is re-executed against time-st[illegible]mped and versioned DB
  - Results as above are returned
- Query store aggregates data usage

Data Citation – Output

- 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
  https://rd-alliance.org/recommendations-working-group-data-citation-revision-oct-20-2015.html
- More detailed report: Bulletin of IEEE TCDL 2016
  http://www.ieee-tcdl.org/Bulletin/v12n1/papers/IEEE-TCDL-DC-2016_paper_1.pdf

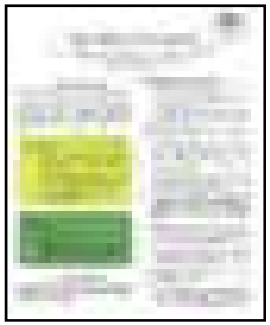
Data Citation – Recommendations

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

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

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

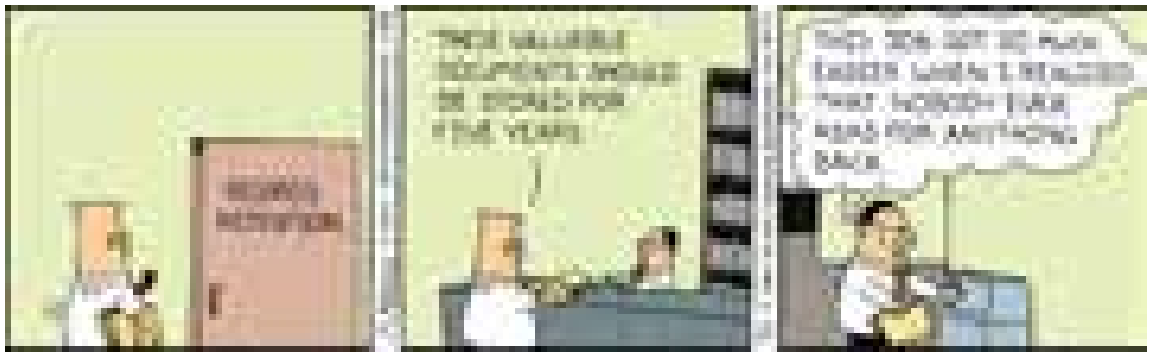
Upon Modifications to the Data Infrastructure
- R13 – Technology Migration
- R14 – Migration Verification



ifs FACULTY OF INFORMATICS

R1: Data Versioning

- **Apply versioning to ensure earlier states of the data can be retrieved**
- Versioning allows tracing the changes
  (static data: no changes – principle still applies)
- No in-place updates or deletes
  - Mark record as deleted, re-insert new record instead of update
  - Keep old versions – only way to be able to “go back”
- Do we really need to keep everything?
  - (“*changes that were never read never existed*”)



Src: http://dilbert.com/strip/1995-09-02

R2: Data Timestamping

- **Ensure that operations on data are timestamped, i.e. any additions, deletions are marked with a timestamp**
- Timestamping is closely related to versioning
- Granularity depends on
  - Change frequency / tracking requirements
    - Per individual operation
    - Batch-operations
    - Grouped in-between read accesses ("*changes that were never read do not matter*")
  - System (data storage, databases)
    - e.g. FAT 2 seconds, NTFS 100 ns, EXT4 1 ns

https://www-03.ibm.com/ibm/history/exhibits/cc/cc_T30.html

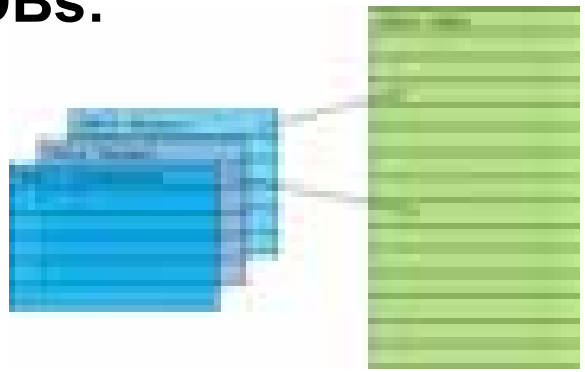
R1 & R2: Versioning / Timestamping

Note:

- R1 & R2 are already pretty much standard in many (RDBMS-) research databases
- Different ways to implement, depending on
  - data type / data structure: RDBMS, CSV, XML, LOD, …
  - data volume
  - amount and type of changes
  - number of APIs, flexibility to change them
- Distributed settings:
  - synchronized clocks, or:
  - each node keeps individual, local time
    time-stamps for distributed queries based on local times

R1 & R2: Versioning / Timestamping

Implementation options for e.g. relational DBs:



- History Table
  - Utilizes full history table
  - Also inserts reflected in history table
  - Doubles storage space, no API adaptions
- Integrated
  - Extend original tables by temporal metadata
  - Expand primary key by timestamp/version column
  - Minimal storage footprint, changes to all APIs
- Hybrid
  - Utilize history table for deleted record versions with metadata
  - Original table reflects latest version only
  - Minimal storage footprint, some API change, expensive query re-writes

- Solution to be adopted depends on trade-off
  - Storage Demand
  - Query Complexity
  - Software/API adaption

R3: Query Store

- **Provide means for storing queries and the associated metadata in order to re-execute them.**
- Approach is based upon queries.
  - Therefore we need to preserve the queries
    - Original and re-written (**R4, R5**), potentially migrated (**R13**)
  - Query parameters and system settings
  - Execution metadata
  - Hash keys (multiple, if re-written) (**R4, R6**)
  - **Persistent identifier(s) (R8)**
  - Citation text (**R10**) …
- Comparatively small,
  even for high query volumes

R4: Query Uniqueness

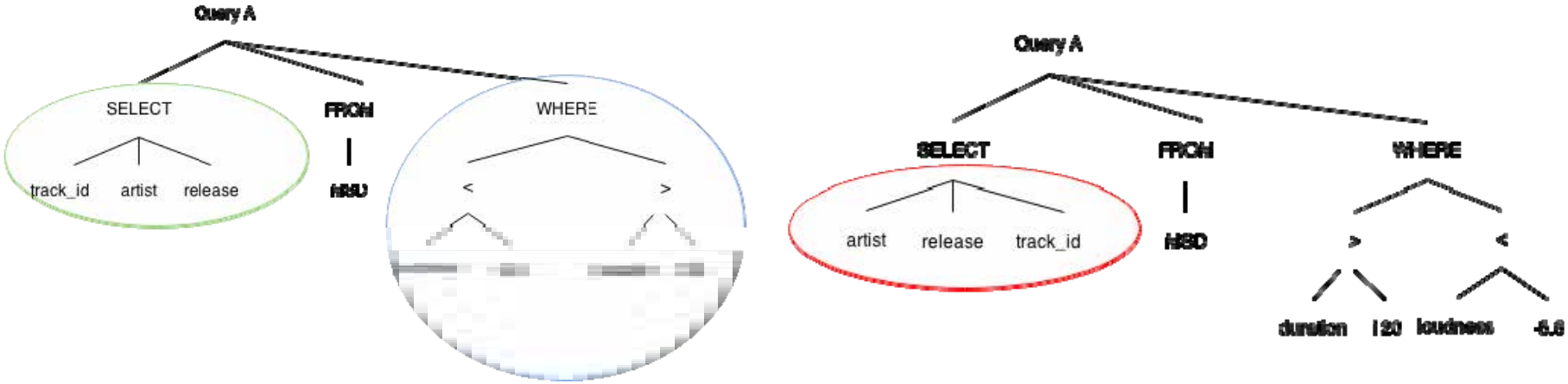
- **Re-write the query to a normalized form so that identical queries can be detected.**
  Compute checksum of the normalized query to efficiently detect identical queries
- Detecting identical queries can be challenging
  - Query semantics can be expressed in different ways
  - Different queries can deliver identical results
  - Interfaces can be used for maintaining a stable query structure
- Best effort, no perfect solution
- Usually not a problem if queries generated via standardized interfaces, e.g. workbench – optional!
- Worst case: two PIDs for semantically equivalent queries

R4: Query Uniqueness

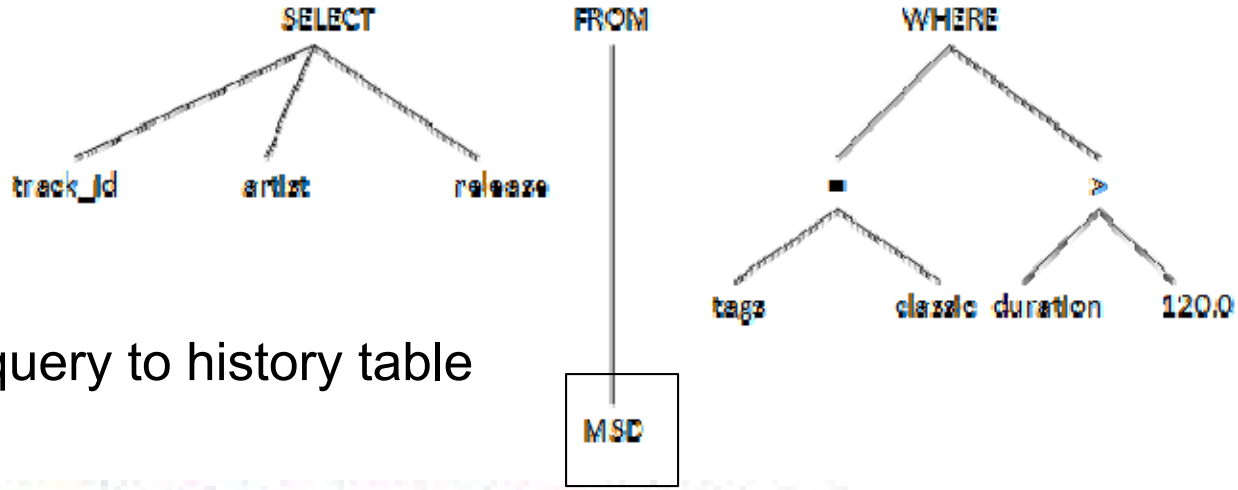
- Query re-writing needed to
  - **Standardization/Normalization** of query to help with identifying semantically identical queries
    - upper/lower case spelling, sorting of filter parameters, …
  - Re-write to **adapt to versioning approach** chosen (versioning in operational tables, separate history table, …), e.g. **identify last change to result set touched** upon (i.e. select including elements marked deleted, check most recent timestamp, to determine correct PID assignment)
  - **Add timestamp** to any select statement in query
  - **Apply unique sort** to any table touched upon in query prior to query to ensure unique sort (see **R5**)

R4: Query Uniqueness

- Normalizing queries to detect identical queries
  - WHERE clause sorted
  - Calculate query string hash
  - Identify semantically identical queries
  - → non-identical queries: columns in different order



R4: Query Uniqueness



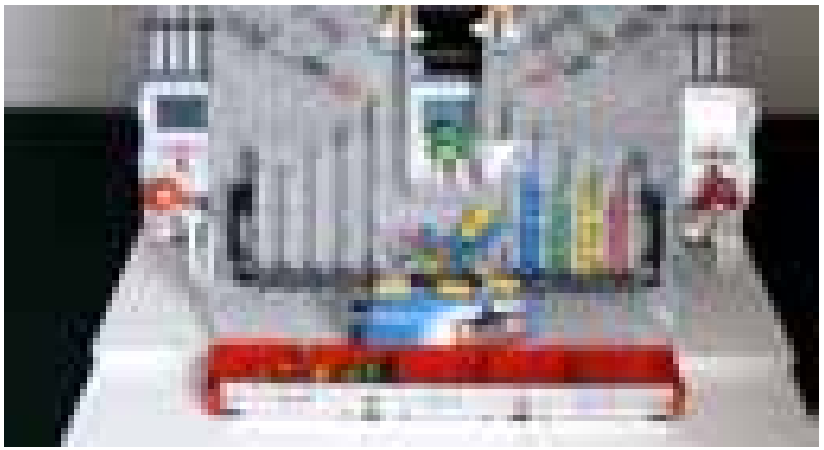
- Adapt query to history table

```
SELECT results.track_id, results.artist, results.release
        FROM MSD AS results JOIN (
                SELECT track_id, max(timestamp) AS latesttimestamp
                FROM MSD
                WHERE timestamp <= (SELECT @queryExecutionTimestamp)
                AND (track_id NOT IN
                        (SELECT track_id FROM MSD AS deletedRecords
                                WHERE deletedRecords.status_mark = 'deleted'
                                AND (deletedRecords.timestamp < @queryExecutionTimestamp))
                )
        GROUP BY track_id
) AS version ON results.track_id = version.track_id AND results.timestamp = version.latesttimestamp

WHERE
    results.tags = 'classic'  AND results.duration< 120
ORDER BY results.track_id;
```

R5: Stable Sorting

- **Ensure that the sorting of the records in the data set is unambiguous and reproducible**
- The sequence of the results in the result set may not be fixed, but data processing results may depend on sequence
  - Many databases are set based
  - The storage system may use non-deterministic features
- If this needs to be addressed, apply default sort (on id) prior to any user-defined sort
- Optional!



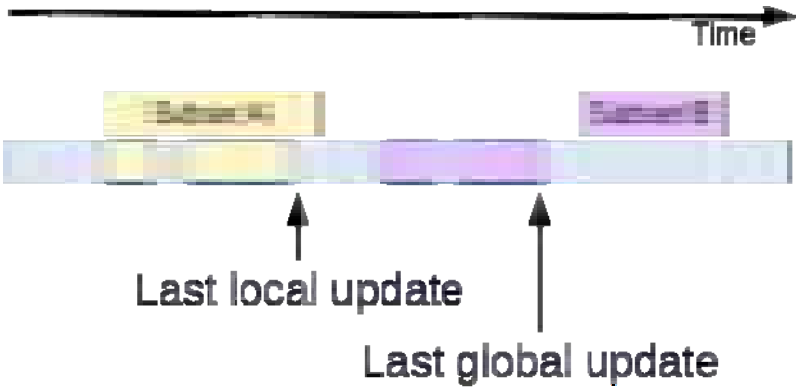
http://www.geek.com/

R6: Result Set Verification

- **Compute fixity information (also referred to as checksum or hash key) of the query result set to enable verification of the correctness of a result upon re-execution.**
- Correctness:
  - No record has changed within a data subset
  - All records which have been in the original data set are also in the re-generated data set
- Compute a hash key
  - Allows to compare the completeness of results
  - For extremely large result sets:
    potentially limit hash input data,
    e.g. only row headers + record id's

R7: Query Timestamping

- **Assign a timestamp to the query based on the last update to the entire database (or the last update to the selection of data affected by the query or the query execution time).**
- Allows to map the execution of a query to a state of the database
  - Execution time: default solution, simple, potentially privacy concerns?
  - Last global update: simple, **recommended**
  - Last update to affected subset: complex to implement
- All equivalent in functionality! (transparent to user)



R8: Query PID

- **Assign a new PID to the query if either the query is new or if the result set returned from an earlier identical query is different due to changes in the data. Otherwise, return the existing PID of the earlier query to the user.**
- **Existing PID**: Identical query (semantics) with identical result set, i.e. no change to any element touched upon by query since first processing of the query
- **New PID**: whenever query semantics is not absolutely identical
  (irrespective of result set being potentially identical!)

R8: Query PID

- Note:
  - Identical result set alone does not mean that the query semantics is identical
  - Will assign different PIDs to capture query semantics
  - Need to normalize query to allow comparison

- Process:
  - Re-write query to adapt to versioning system, stable sorting, …
  - Determine query hash
  - Execute user query and determine result set hash
  - Check query store for queries with identical query hash
    - If found, check for identical result set hash

R9: Store the Query

- **Store query and metadata (e.g. PID, original and normalised query, query and result set checksum, timestamp, superset PID, data set description, and other) in the query store.**
  - Query store is central infrastructure
  - Stores query details for long term
  - Provides information even when the data should be gone
  - Responsible for re-execution
  - Holds data for landing pages
  - Stores sensitive information

R10: Create Citation Texts

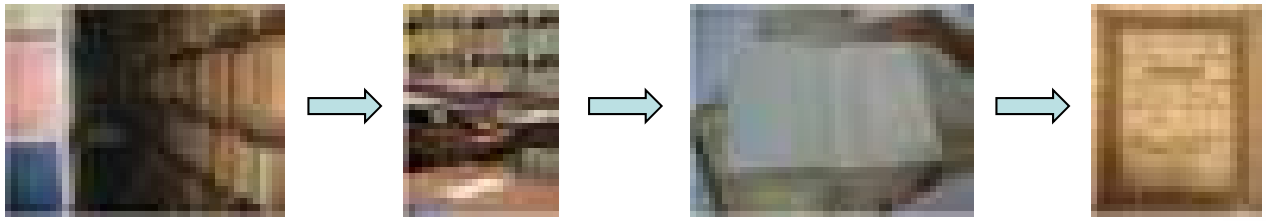
- **Generate citation texts in the format prevalent in the designated community for lowering the barrier for citing and sharing the data.**
  Include the PID in the citation text snippet.
- Researchers are “lazy”/efficient
  - Support citing by allow them to copy and paste citations for data
  - Citations contain text including PIDs and timestamps
  - Adapted for each community
- **2 PIDs!**
  - Superset: the “database” and it’s holder (repository, data center)
  - Subset: based on the query
  - Accumulate credits for subset and (dynamic) data collection/holder

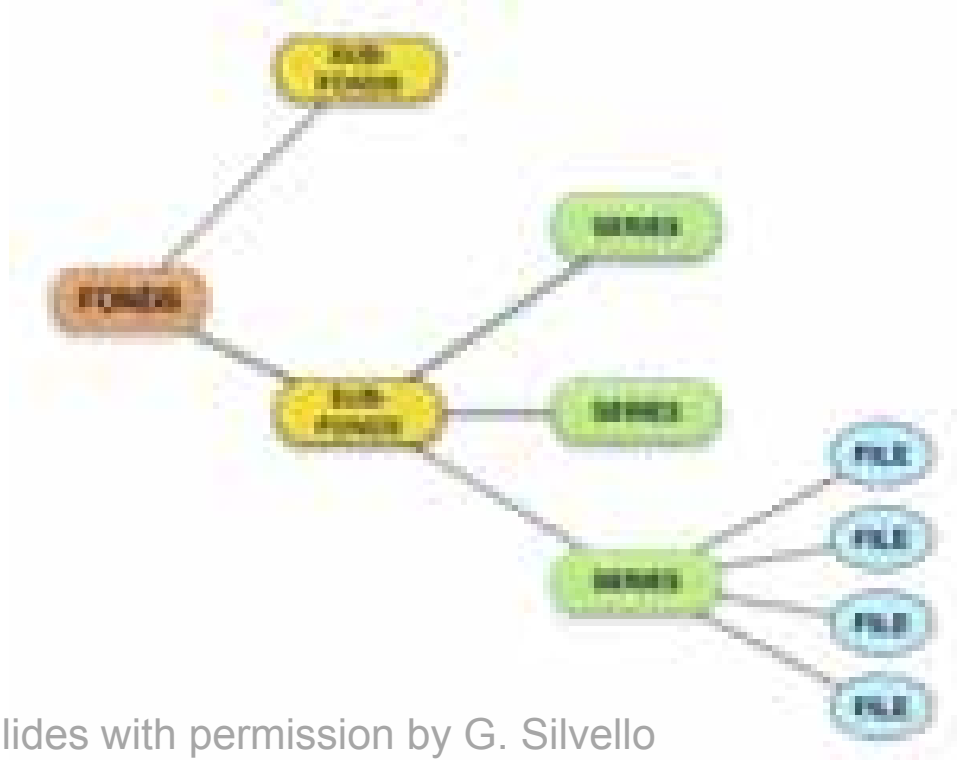
R10: Automated Citation Texts

- Can be created automatically
  - relatively simple for relational
  - more complex for hierarchical/XML
- Learning to Cite:
  - Gianmaria Silvello. Learning to Cite Framework: How to Automatically Construct Citations for Hierarchical Data. Journal of the Association for Information Science and Technology (JASIST), Volume 68 issue 6, pp. 1505-1524, June 2017.
  - http://www.dei.unipd.it/~silvello/datacitation

R10: Automated Citation Texts

- EAD: Encoded Archival Description





Slides with permission by G. Silvello

ifs FACULTY OF INFORMATICS

R10: Automated Citation Texts

- A human-readable citation:

Correspondence, 1951-1956.

"The Elements of Legal Theory" (unpublished). Books, box 135. Part II:

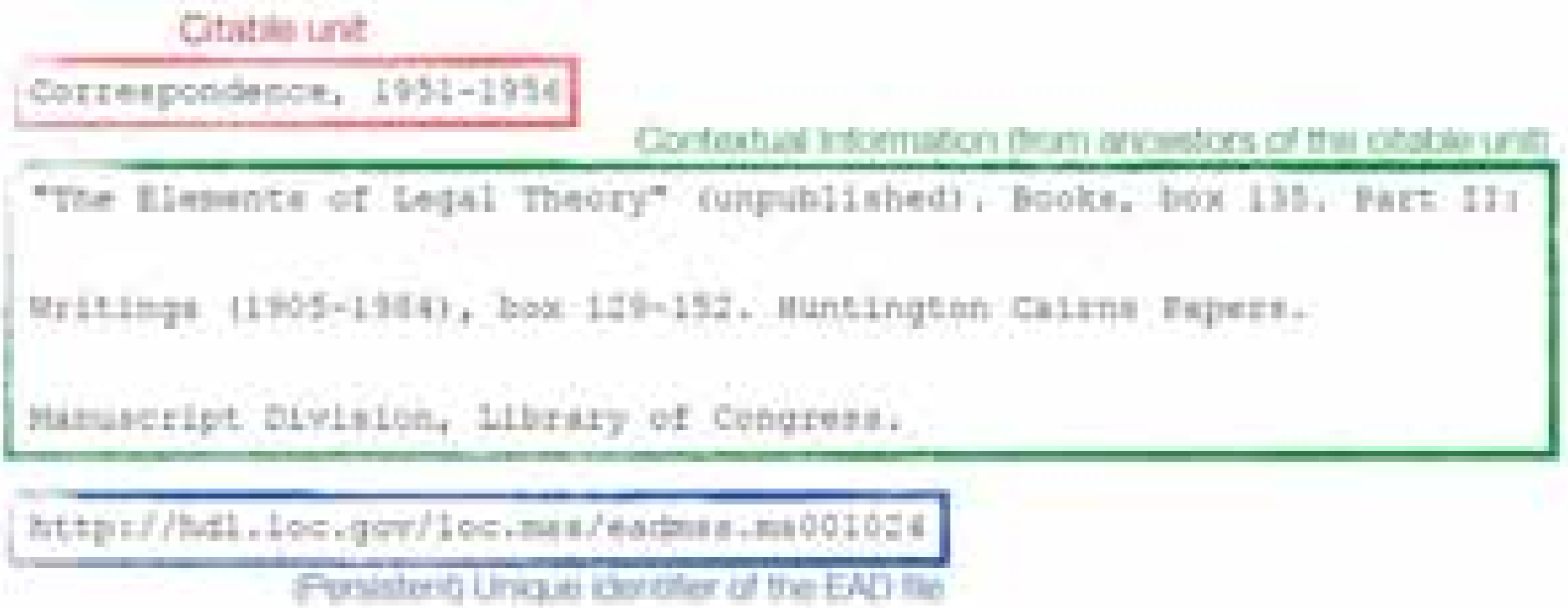
Writings (1905-1944), box 120-152. Huntington Cairns Papers.

Manuscript Division, Library of Congress.

http://hdl.loc.gov/loc.mss/eadmss.ms002024

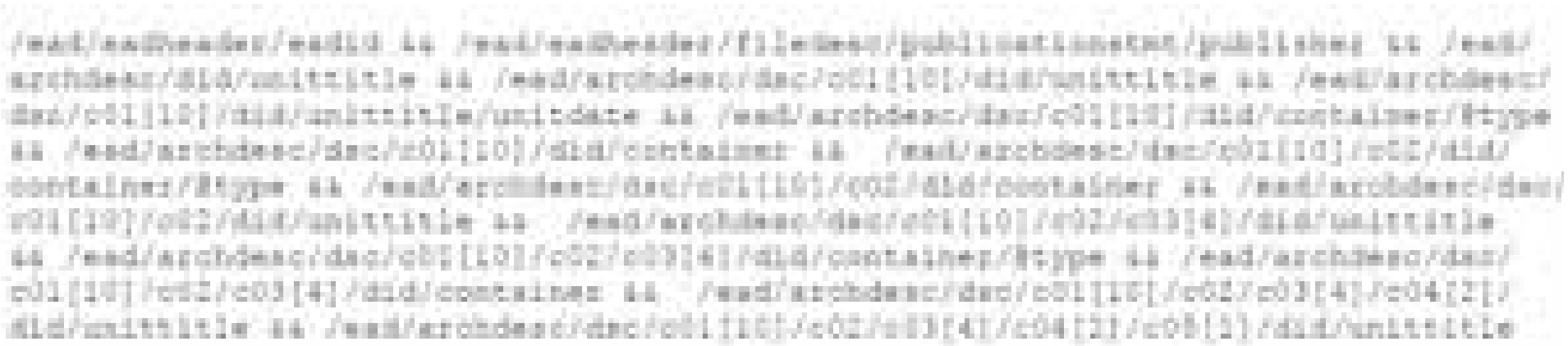
R10: Automated Citation Texts

- A human-readable citation:



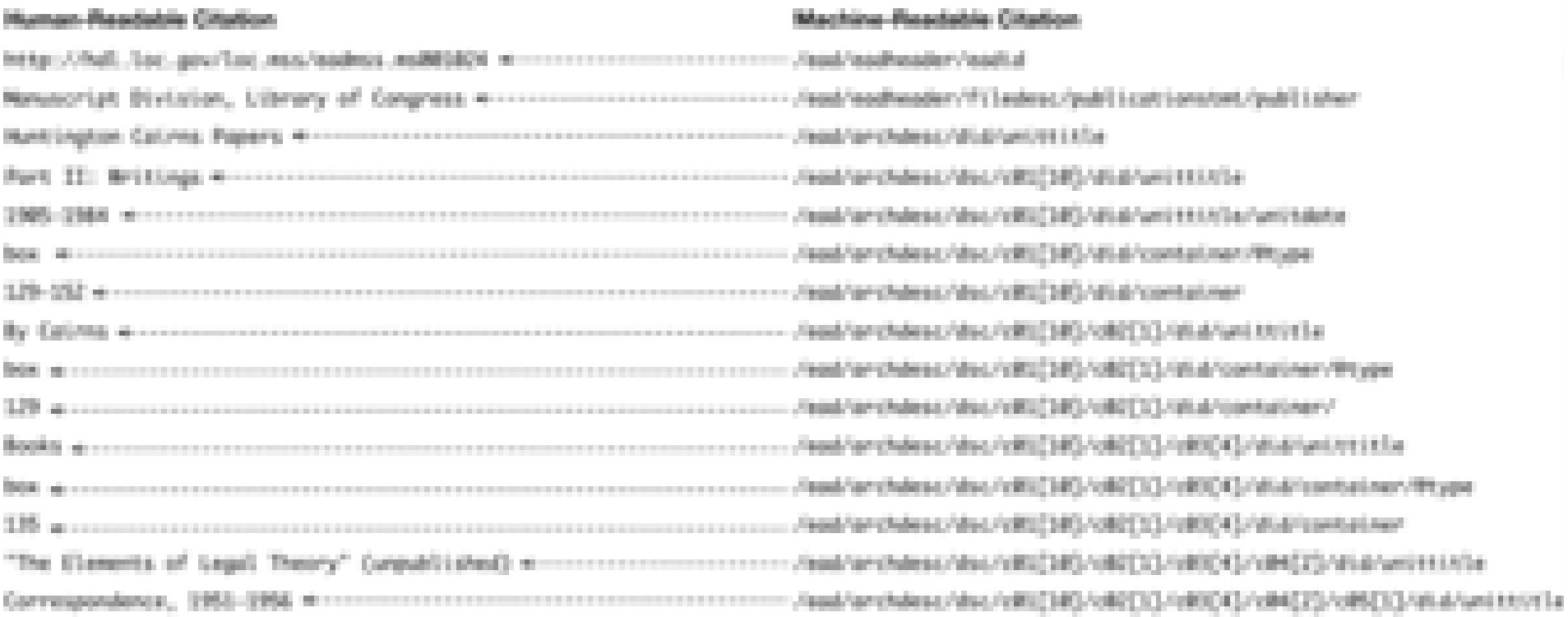
R10: Automated Citation Texts

- A machine-readable citation:
  - Conjunction of XML paths



R10: Automated Citation Texts

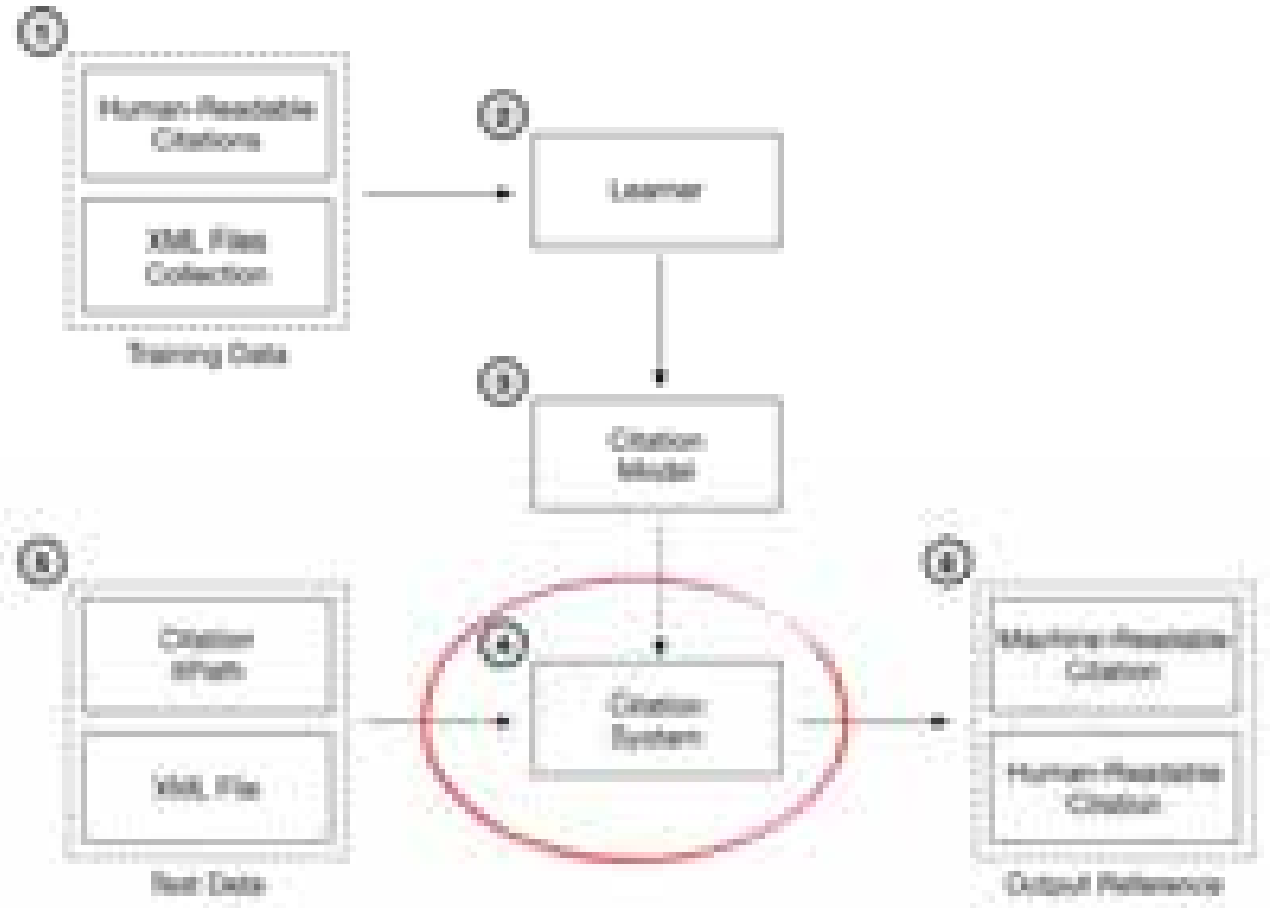
- Mapping machine-readable to human-readable:



Slides with permission by G. Silvello

R10: Automated Citation Texts

- Learning citation models

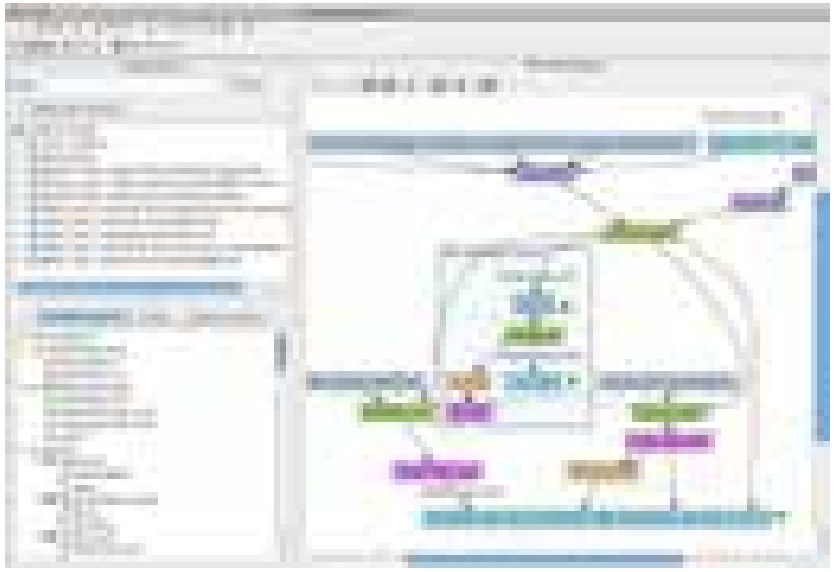


R11: Landing Page

- **Make the PIDs resolve to a human readable landing page that provides the data (via query re-execution) and metadata, including a link to the superset (PID of the data source) and citation text snippet.**
  - Data sets and subsets uniquely identifiable by their PID, which resolves to a human readable landing page.
  - Landing page reachable by a unique URL, presented in a Web browser
  - Not all information needs to be provided on landing page
    (e.g. query strings frequently not relevant / potential security threat)

R12: Machine Actionability

- **Provide an API / machine actionable landing page to access metadata and data via query re-execution.**
  - Experiments are increasingly automated
  - Machines most likely to consume data citations
  - Allows machines to resolve PIDs, access metadata and data
  - Note: does NOT imply full / automatic access to data!
    - Authentication
    - Load analysis
  - Handshake, content negotiation, …
  - Allows automatic meta-studies, monitoring, …



R13: Technology Migration

- **When data is migrated to a new representation (e.g. new database system, a new schema or a completely different technology), migrate also the queries and associated fixity information.**
  - Technology evolves and data may be moved to a new technology stack
  - Query languages change
- Migration required
  - Migrate data and the queries (both are with the data center!)
  - Adapt versioning, re-compute query hash-keys
  - Maybe decide to keep “original” queries in the provenace trace
- Note: such data migrations constitute major projects, usually happen rarely – require all APIs to be adapted, …

R14: Migration Verification

- **Verify successful data and query migration, ensuring that queries can be re-executed correctly.**
- Sanity check: After migration is done, verify that the data can still be retrieved correctly
- Use query and result set hashes in the query store to verify results
- If hash function is incompatible/cannot be computed on new system as hash input data sequence cannot be obtained, pairwise comparison of subset elements
  - May constitute new PID / data subset in this case, as subsequent processes will not be able to use it as input if result set presentation has changed, breaks processes

RDA Recommendations - Summary

- Building blocks of supporting dynamic data citation:
  - Uniquely identifiable data records
  - Versioned data, marking changes as insertion/deletion
  - Time stamps of data insertion / deletions
  - “Query language” for constructing subsets
- Add modules:
  - Persistent query store: queries and the timestamp (either: <when issued> or <of last change to data>)
  - Query rewriting module
  - PID assignment for queries that enables access
- Stable across data source migrations (e.g. diff. DBMS), scalable, machine-actionable

RDA Recommendations - Summary

- ***Benefits***
  - Allows **identifying, retrieving and citing the precise data subset** with minimal storage overhead by only storing the versioned data and the queries used for extracting it
  - Allows retrieving the data both **as it existed** at a given point in time as well as the **current view** on it, by re-executing the same query with the stored or current timestamp
  - It allows to cite even an **empty set**!
  - The query stored for identifying data subsets provides valuable **provenance data**
  - Query store collects **information on data usage**, offering a basis for data management decisions
  - **Metadata** such as checksums support the verification of the correctness and **authenticity** of data sets retrieved
  - The same principles work for **all types of data**

RDA Recommendations - Summary

- ***Some considerations and questions***
  - May data be deleted?
    Yes, of course, given appropriate policies. Queries may then not be re-executable against the original timestamp anymore
  - Does the system need to store every query?
    No, only data sets that should be persisted for citation and later re-use need to be stored.
  - Can I obtain only the most recent data set?
    Queries can be re-executed with the original timestamp or with the current timestamp or any other timestamp desired.
  - Which PID system should be used?
    Any PID system can, in principle, be applied according to the institutional policy.

Outline

- Why should we want to cite data?
- What are the challenges in data identification and citation?
- How should we do it, according to the RDA WG?
- Who is doing it so far, and how?
- Summary

Pilots / Adopters

- Several adopters
  - Different types of data, different settings, …
  - CSV & SQL reference implementation (SBA/TUW)
- Pilots:
  - Biomedical BigData Sharing, Electronic Health Records (Center for Biomedical Informatics, Washington Univ. in St. Louis)
  - Marine Research Data Biological & Chemical Oceanography Data Management Office (BCO-DMO)
  - Vermont Monitoring Cooperative: Forest Ecosystem Monitoring
  - ARGO Boy Network, British Oceanographic Data Centre (BODC)
  - Virtual Atomic and Molecular Data Centre (VAMDC)
  - UK Riverflow Riverflow Archive, Centre for Ecology and Hydrology

VAMDC consortium
BCO-DMO
ifremer
NERC
Argo
PADC
ifs
Vermont Monitoring Cooperative
FACULTY OF INFORMATICS

Pilots / Adopters

- Series of Webinars presenting implementations
  - Recordings, slides, supporting papers
  - https://www.rd-alliance.org/group/data-citation-wg/webconference/webconference-data-citation-wg.html

  - 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**

ifs FACULTY OF INFORMATICS

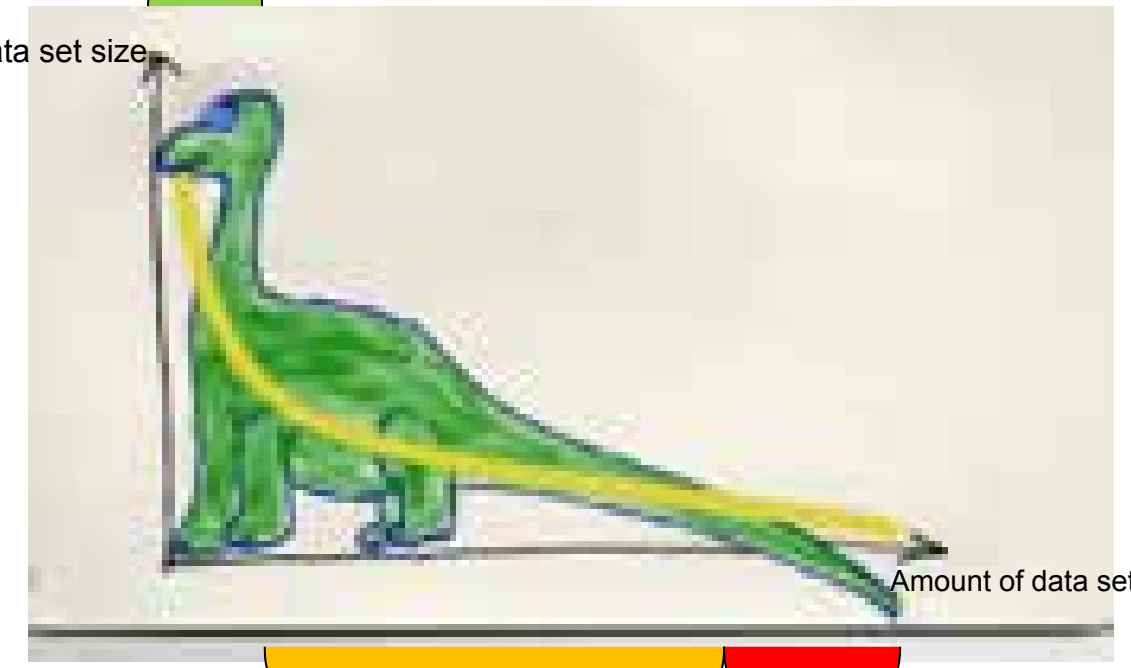

RDA
RESEARCH DATA ALLIANCE
Reference Implementation for
CSV Data (and SQL)
Stefan Pröll, SBA
Christoph Meixner, TU Wien
research data sharing without barriers
rd-alliance.org

Large Scale Research Settings

- RDA recommendations implemented in data infrastructures
- Required adaptions
  - Introduce versioning, if not already in place
  - Capture sub-setting process (queries)
  - Implement dedicated query store to store queries
  - A bit of additional functionality (query re-writing, hash functions, …)
- Done! ?
  - “Big data”, database driven
  - Well-defined interfaces
  - Trained experts available
  - “*Complex, only for professional research infrastructures*” ?

Long Tail Research Data

Big data,
well organized,
often used and cited



Less well organized,
non-standardised
no dedicated infrastructure

"Dark data"

Slide by Stefan Pröll

Prototype Implementations

- Solution for small-scale data
  - CSV files, no “expensive” infrastructure, low overhead

2 Reference implementations :

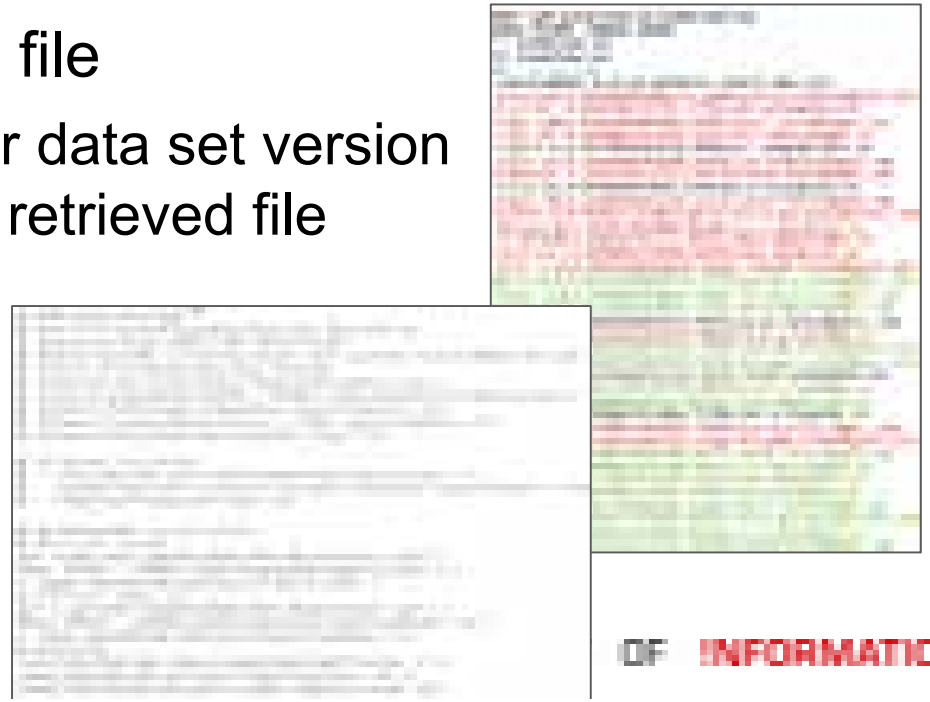
- **Git** based Prototypes: widely used versioning system
  - A) Using separate folders
  - B) Using branches
- **MySQL** based Prototype:
  - C) Migrates CSV data into relational database

- Data backend responsible for versioning data sets
- Subsets are created with scripts or queries via API or Web Interface
- Transparent to user: always CSV

Git-Based Reference Implementation

Git Implementation 1

- Upload CSV files to Git repository (versioning)
- Subsets created via scripting language (e.g. R)
  - Select rows/columns, sort, returns CSV + metadata file
  - Metadata file with script parameters stored in Git
  - (Scripts stored in Git as well)
- PID assigned to metadata file
  - Use Git to retrieve proper data set version and re-execute script on retrieved file



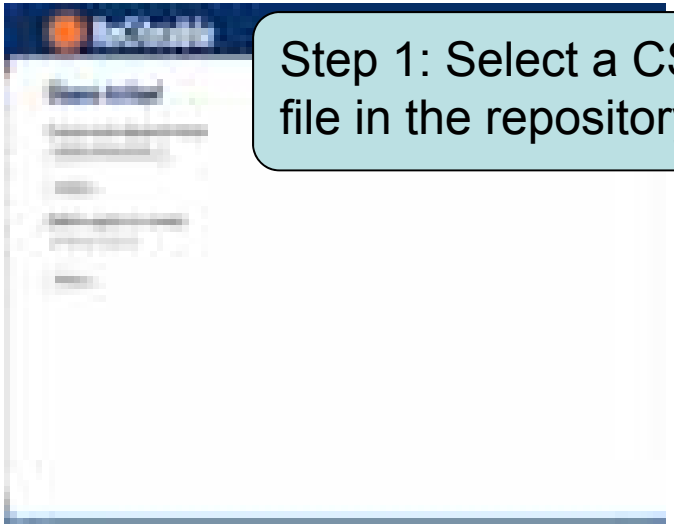
Git-Based Reference Implementation

Git Implementation 2

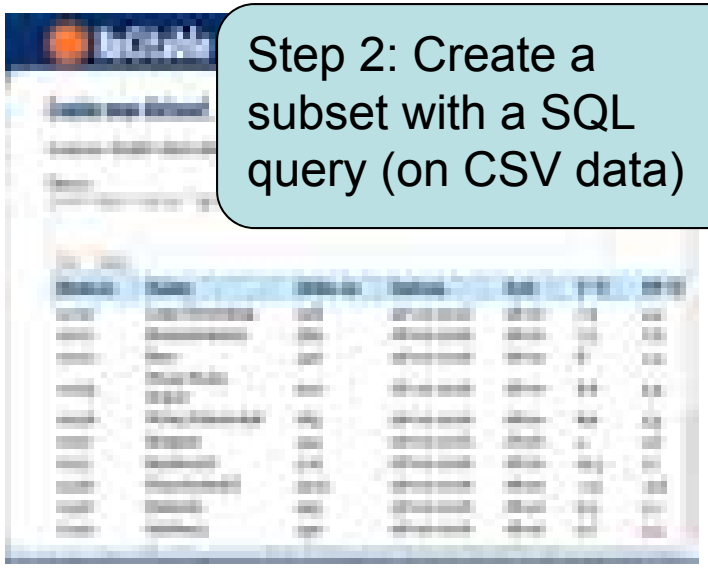
- Addresses issues
  - common commit history, branching data
- Using Git branching model:
  Orphaned branches for queries and data
  - Keeps commit history clean
  - Allows merging of data files
- Web interface for queries (CSV2JDBC)
- Use commit hash for identification
  - Assigned PID hashed with SHA1
  - Use hash of PID as filename (ensure permissible characters)

Git-Based Reference Implementation

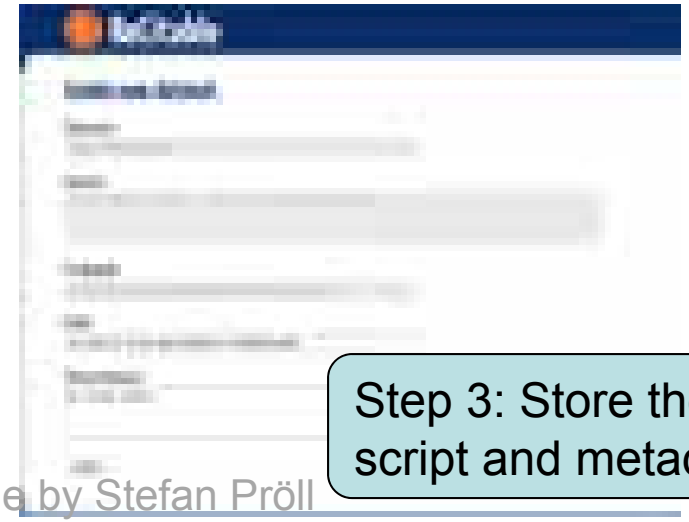
- Prototype: https://github.com/Mercynary/recitable



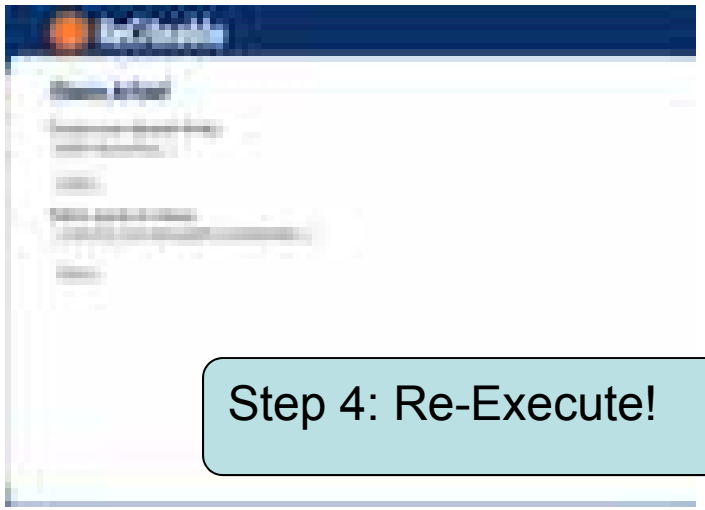
Step 1: Select a CSV file in the repository



Step 2: Create a subset with a SQL query (on CSV data)



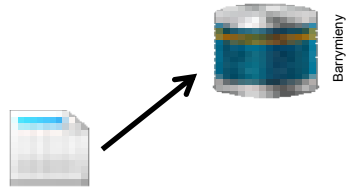
Step 3: Store the query script and metadata



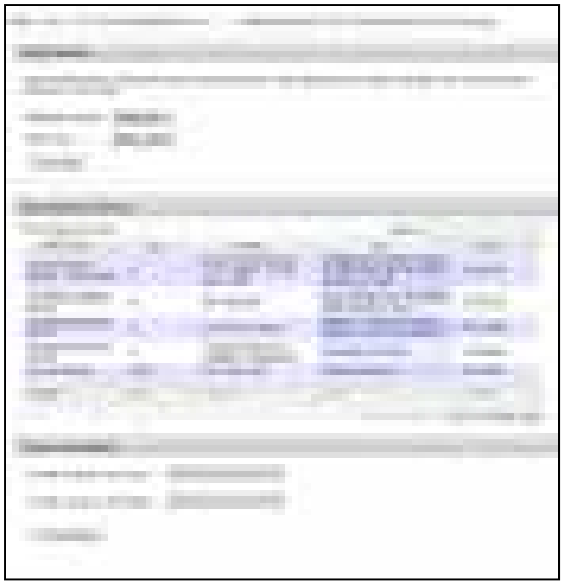
Step 4: Re-Execute!

MySQL-Based Reference Implementation

MySQL Prototype



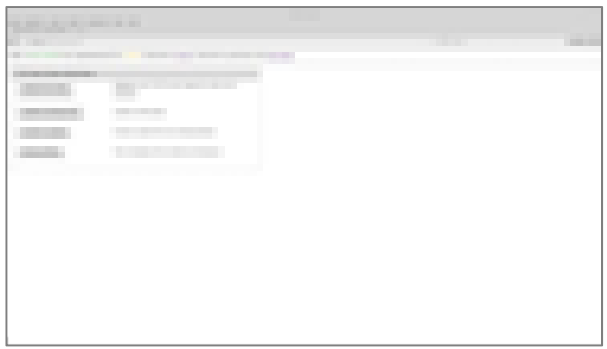
- Data upload
  - User uploads a CSV file into the system
- Data migration from CSV file into RDBMS
  - Generate table structure
  - Add metadata columns (versioning)
  - Add indices (performance)

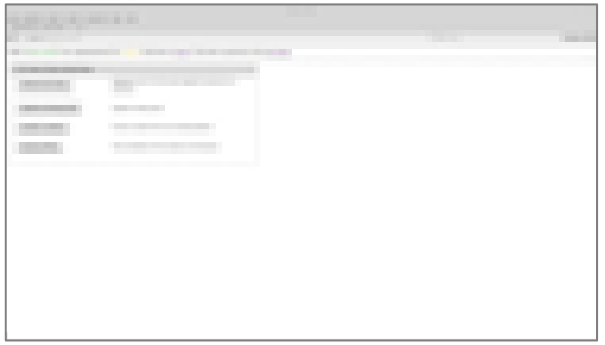


- Dynamic data
  - Insert, update and delete records
  - Events are recorded with a timestamp
- Subset creation
  - User selects columns, filters and sorts records in web interface
  - System traces the selection process
  - Exports CSV

MySQL-Based Reference Implementation

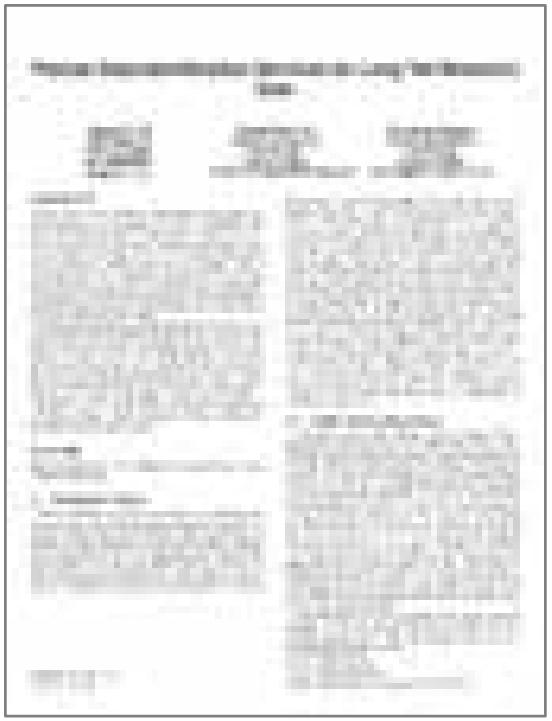
- Source at Github:
  - https://github.com/datascience/RDA-WGDC-CSV-Data-Citation-Prototype
- Videos:
  - Login: https://youtu.be/EnraIwbQfM0
  - Upload: https://youtu.be/xJruifX9E2U
  - Subset: https://www.youtube.com/watch?v=it4sC5vYiZQ
  - Resolver: https://youtu.be/FHsvjsUMiiY
  - Update: https://youtu.be/cMZ0xoZHUyI





CSV Reference Implementations

- Stefan Pröll, Christoph Meixner, Andreas Rauber
  Precise Data Identification Services for Long Tail Research Data.
  Proceedings of the intl. Conference on Preservation of Digital Objects (iPRES2016), Oct. 3-6 2016, Bern, Switzerland.
- Source at Github:
  https://github.com/datascience/RDA-WGDC-CSV-Data-Citation-Prototype
- Videos:
  - Login: https://youtu.be/EnraIwbQfM0
  - Upload: https://youtu.be/xJruifX9E2U
  - Subset: https://www.youtube.com/watch?v=it4sC5vYiZQ
  - Resolver: https://youtu.be/FHsvjsUMiiY
  - Update: https://youtu.be/cMZ0xoZHUyI




RDA
RESEARCH DATA ALLIANCE
WG Data Citation Pilot
CBMI @ WUSTL
Cynthia Hudson Vitale, Leslie McIntosh,
Snehil Gupta
Washington University in St.Luis
research data sharing without barriers
rd-alliance.org

Biomedical Adoption Project Goals

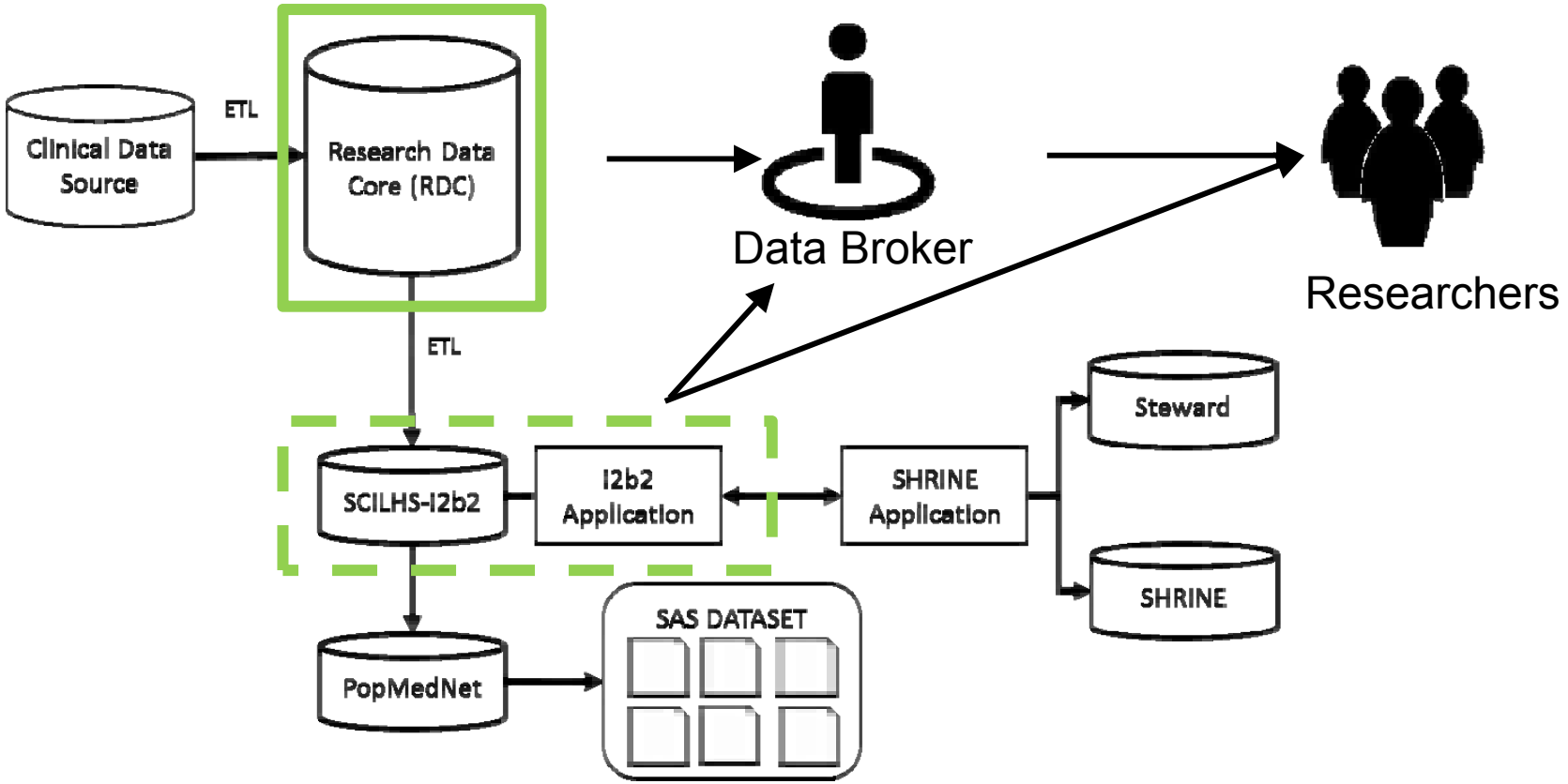
- Implement RDA Data Citation WG recommendation to local Washington U i2b2
- Engage other i2b2 community adoptees
- Contribute source code back to i2b2 community

- Repository
  https://github.com/CBMIWU/Research_Reproducibility
- Slides
  http://bit.ly/2cnWorU
- Bibliography
  https://www.zotero.org/groups/biomedical_informatics_resrepro

Slide by Leslie McIntosh

ifs FACULTY OF INFORMATICS

RDA-MacArthur Grant Focus



Slide by Leslie McIntosh

R1 and R2 Implementation

1

PostgreSQL Extension
“temporal_tables”

2

triggers

RDC.table

c1	c2	c3

sys_period

3

RDC.hist_table*

c1	c2	c3	sys_period

*stores history of
data changes

Return on Investment (ROI) - Estimated

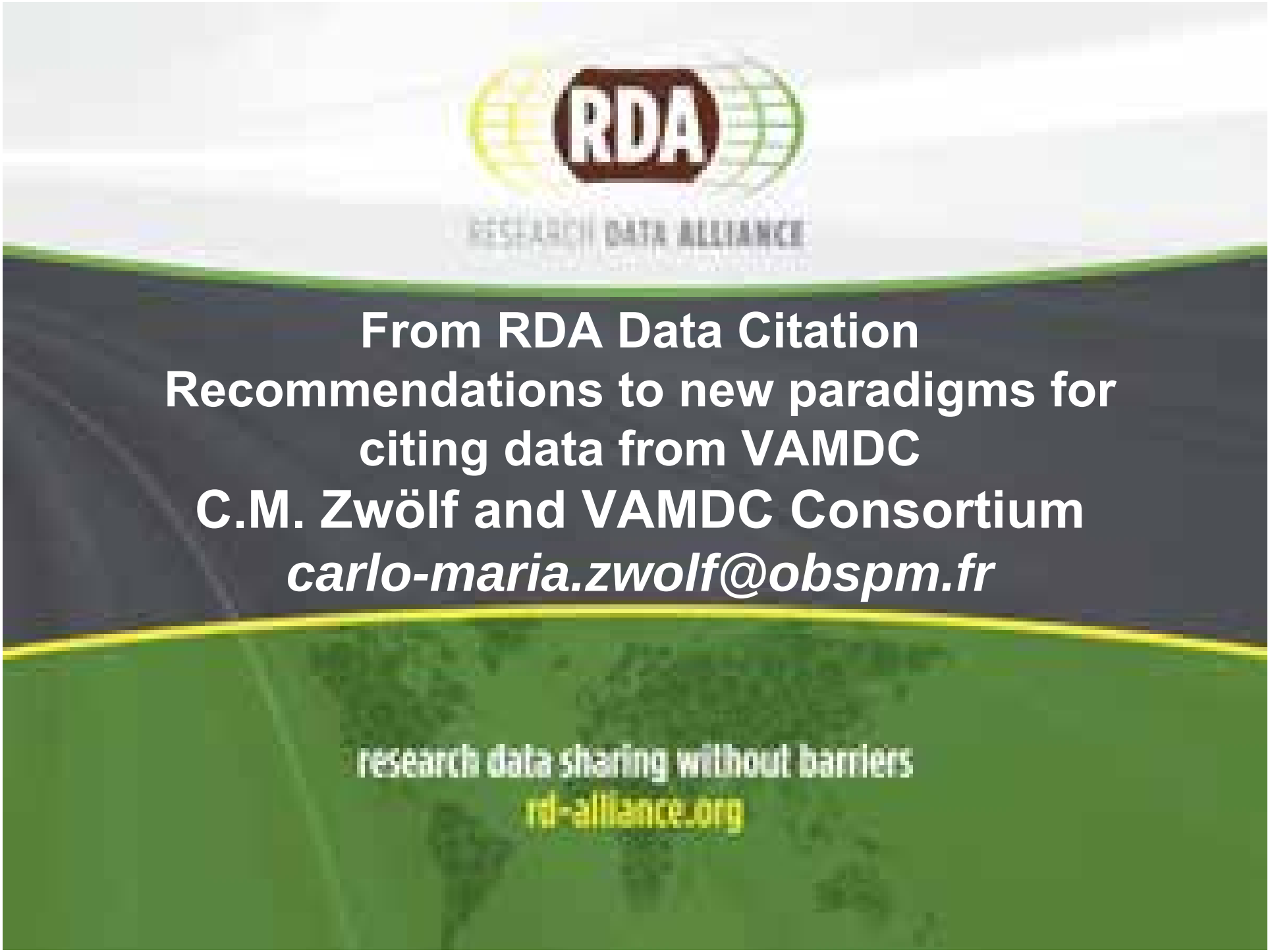
- 20 hours to complete 1 study
- $150/hr (unsubsidized)
- $3000 per study
- 115 research studies per year
- **14 replication studies**

CBMI Research Reproducibility Resources

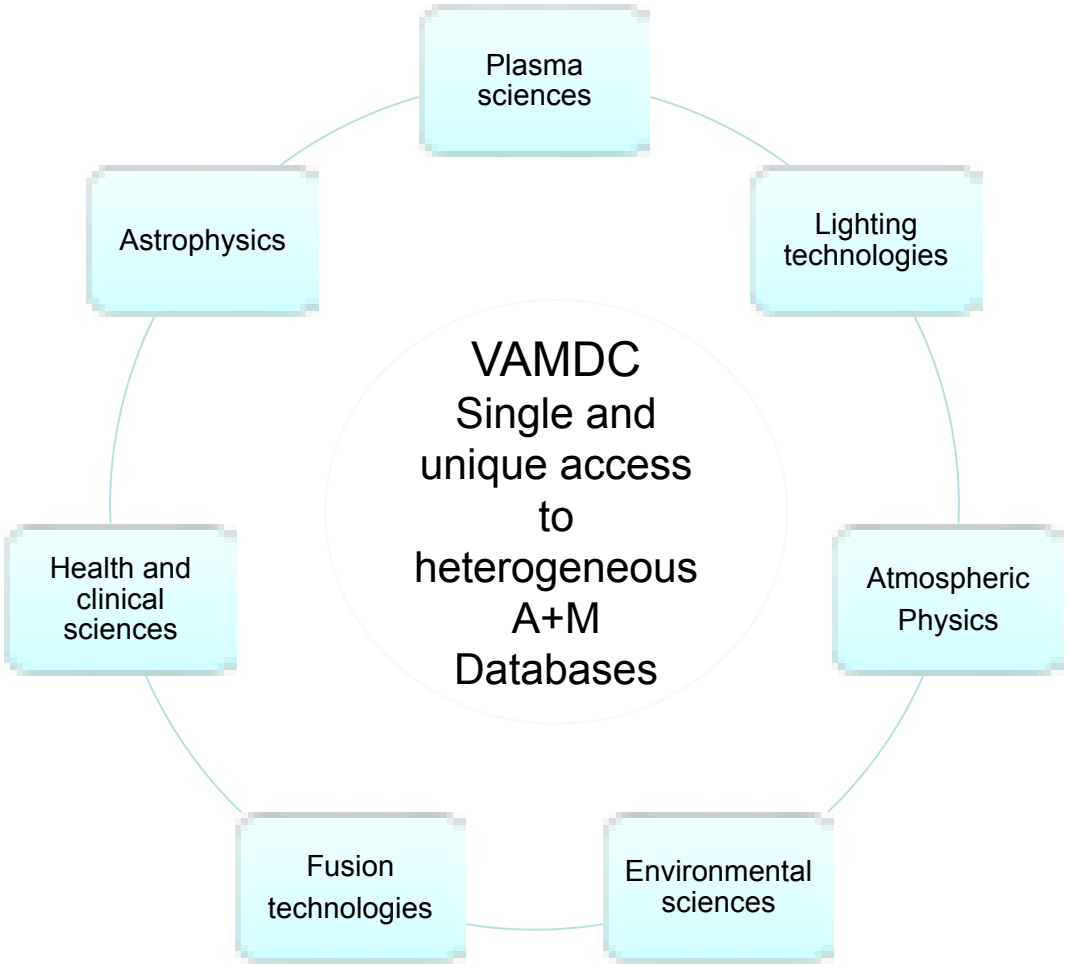
- Repository
  https://github.com/CBMIWU/Research_Reproducibility
- Slides
  http://bit.ly/2cnWorU
- Bibliography
  https://www.zotero.org/groups/biomedical_informatics_resrepro

Slide by Leslie McIntosh

ifs FACULTY OF INFORMATICS


RDA
RESEARCH DATA ALLIANCE
From RDA Data Citation Recommendations to new paradigms for citing data from VAMDC
C.M. Zwölf and VAMDC Consortium
carlo-maria.zwolf@obspm.fr
research data sharing without barriers
rd-alliance.org

The Virtual Atomic and Molecular Data Centre

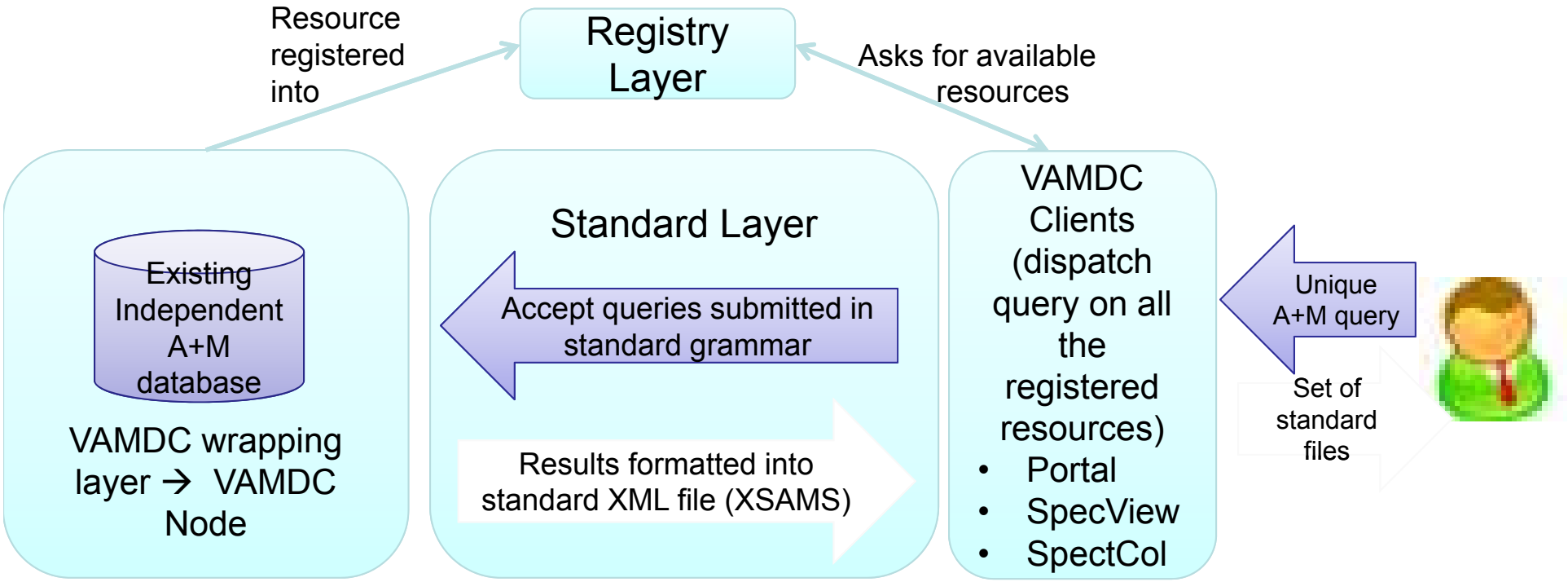


- Federates 29 heterogeneous databases http://portal.vamdc.org/
- The "V" of VAMDC stands for Virtual in the sense that the e-infrastructure does not contain data. The infrastructure is a wrapping for exposing in a unified way a set of heterogeneous databases.
- The consortium is politically organized around a Memorandum of understanding (15 international members have signed the MoU, 1 November 2014)
- High quality scientific data come from different Physical/Chemical Communities
- Provides data producers with a large dissemination platform
- Remove bottleneck between data-producers and wide body of users

Slide by Carlo Maria Zwölf

IfS FACULTY OF INFORMATICS

VAMDC Infrastructure Technical Architecture



- VAMDC is agnostic about the local data storage strategy on each node.
- Each node implements the access/query/result protocols.
- There is no central management system.
- Decisions about technical evolutions are made by consensus in Consortium.

➢ It is both technical and political challenging to implement the WG recommendations.

Slide by Carlo Maria Zwölf

ifs FACULTY OF INFORMATICS

Let us implement the recommendation!!

Implementation will be an overlay to the standard / output layer, thus independent from any specific data-node

Tagging versions of data

Two layers mechanisms

1 → Fine grained granularity:
Evolution of XSAMS output standard for tracking data modifications

2 → Coarse grained granularity:
At each data modification to a given data node, the version of the Data-Node changes

With the second mechanism we know that something changed : in other words, we know that the result of an identical query may be different from one version to the other. The detail of which data changed is accessible using the first mechanisms.

Slide by Carlo Maria Zwölf

Let us implement the recommendation!!

Implementation will be an overlay to the standard / output layer, thus independent from any specific data-node

Tagging versions of data	Query Store
Two layers mechanisms	
1 → Fine grained granularity: Evolution of XSAMS output standard for tracking data modifications	Is built over the versioning of Data
2 → Coarse grained granularity: At each data modification to a given data node, the version of the Data-Node changes	Is plugged over the existing VAMDC data-extraction mechanisms
	Due to the distributed VAMDC architecture, the Query Store architecture is similar to a log-service

With the second mechanism we know that something changed : in other words, we know that the result of an identical query may be different from one version to the other. The detail of which data changed is accessible using the first mechanisms

Slide by Carlo Maria Zwölf

ifs FACULTY OF INFORMATICS

Data-Versioning: Overview of the fine grained mechanisms

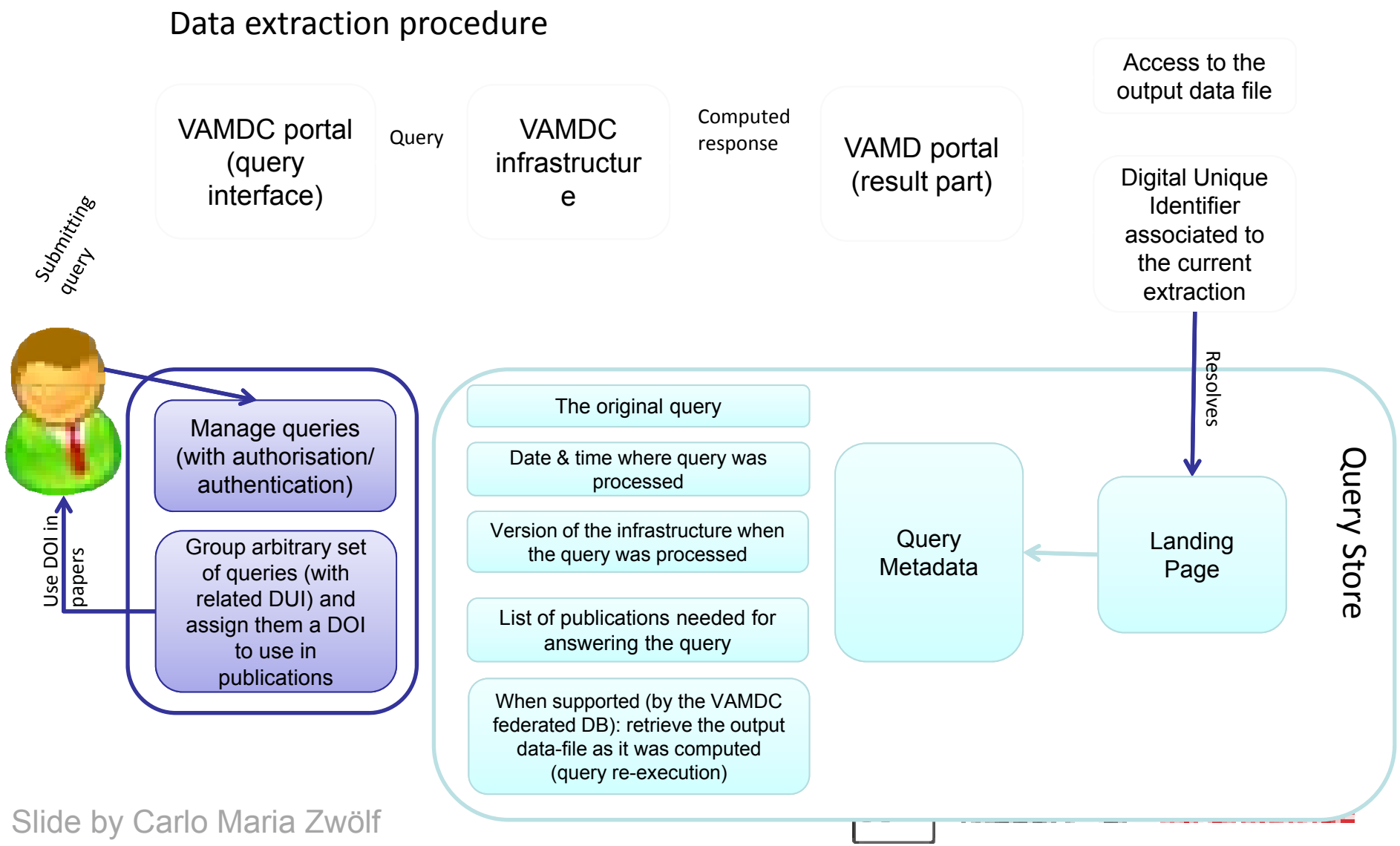
This approach has several advantages:

- It solves the data tagging granularity problem
- It is independent from what is considered a dataset
- The new files are compliant with old libraries & processing programs
  - We add a new feature, an overlay to the existing structure
  - We induce a structuration, without changing the structure

New model for datasets citation and extraction reproducibility in VAMDC, C.M. Zwölf, N. Moreau, M.-L. Dubernet, *J. Mol. Spectrosc. (2016)*, http://dx.doi.org/10.1016/j.jms.2016.04.009 Arxiv version: https://arxiv.org/abs/1606.00405

Let us focus on the query store:

Sketching the functioning – From the final-user point of view:



Slide by Carlo Maria Zwölf

Let us focus on the query store:

The difficulty we have to cope with:

- Handle a query store in a distributed environment (RDA did not design it for these configurations)
- Integrate the query store with the existing VAMDC infrastructure

The implementation of the query store is the goal of a jointly collaboration between VAMDC and RDA-Europe

- Development started during spring 2016
- Final product released during 2017

Collaboration with Elsevier for embedding the VAMDC query store into the pages displaying the digital version of papers.
Designing technical solution for

- Paper / data linking at the paper submission (for authors)
- Paper / data linking at the paper display (for readers)


RDA
RESEARCH DATA ALLIANCE
Climate Change Centre Austria
(CCCA)
Chris Schubert
chris.Schubert@ccca.ac.at
research data sharing without barriers
rd-alliance.org

CCCA

Climate Change Centre Austria

- Climate research network for sustained, high-quality Austrian climate research.
- 28 members (11 universities, 13 non-univ. institutions, 4 supporting members)
- Structure: Coordination Office (Vienna, BOKU), Service Centre (Univ. Graz), Data Centre (ZAMG, Vienna)
- Service available at http://data.ccca.ac.at

Slide by Chris Schubert

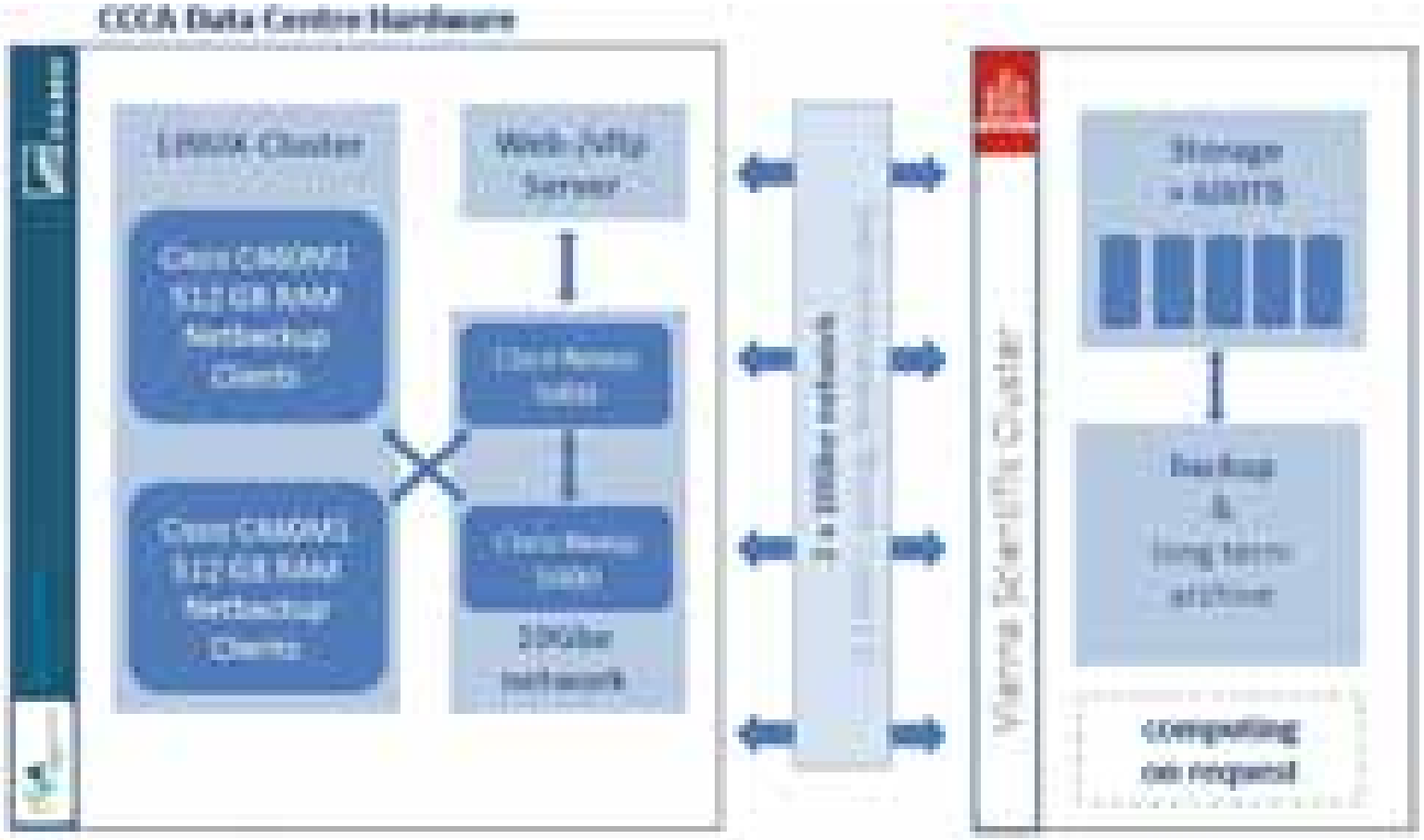
CCCA

CCCA Data Centre

- **provision** of climate-relevant information, data, algorithms, reports
- **interoperable interfaces** to international portals, standards, legislation (e.g. INSPIRE)
- conception for **long term archiving** of research data & repositories
- capacity building, consultancy and **support** for **data sharing**

Slide by Chris Schubert

CCCA



Slide by Chris Schubert

FACULTY OF INFORMATICS

CCCA

... a data portal among many others?

FEATURE No. 4 & 5

- handle® Service implemented to serve persistent identifier (PID) -> fundamental for DataCitation

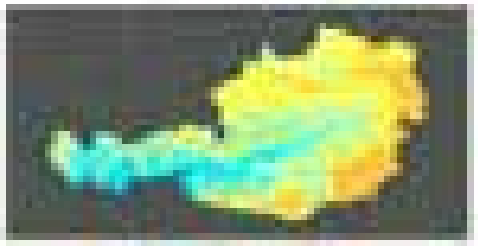
hdl:handle.net/20.500.12736/7b933f4de

Cite this research:

Your Publication

formal
Data Citation

Your Data

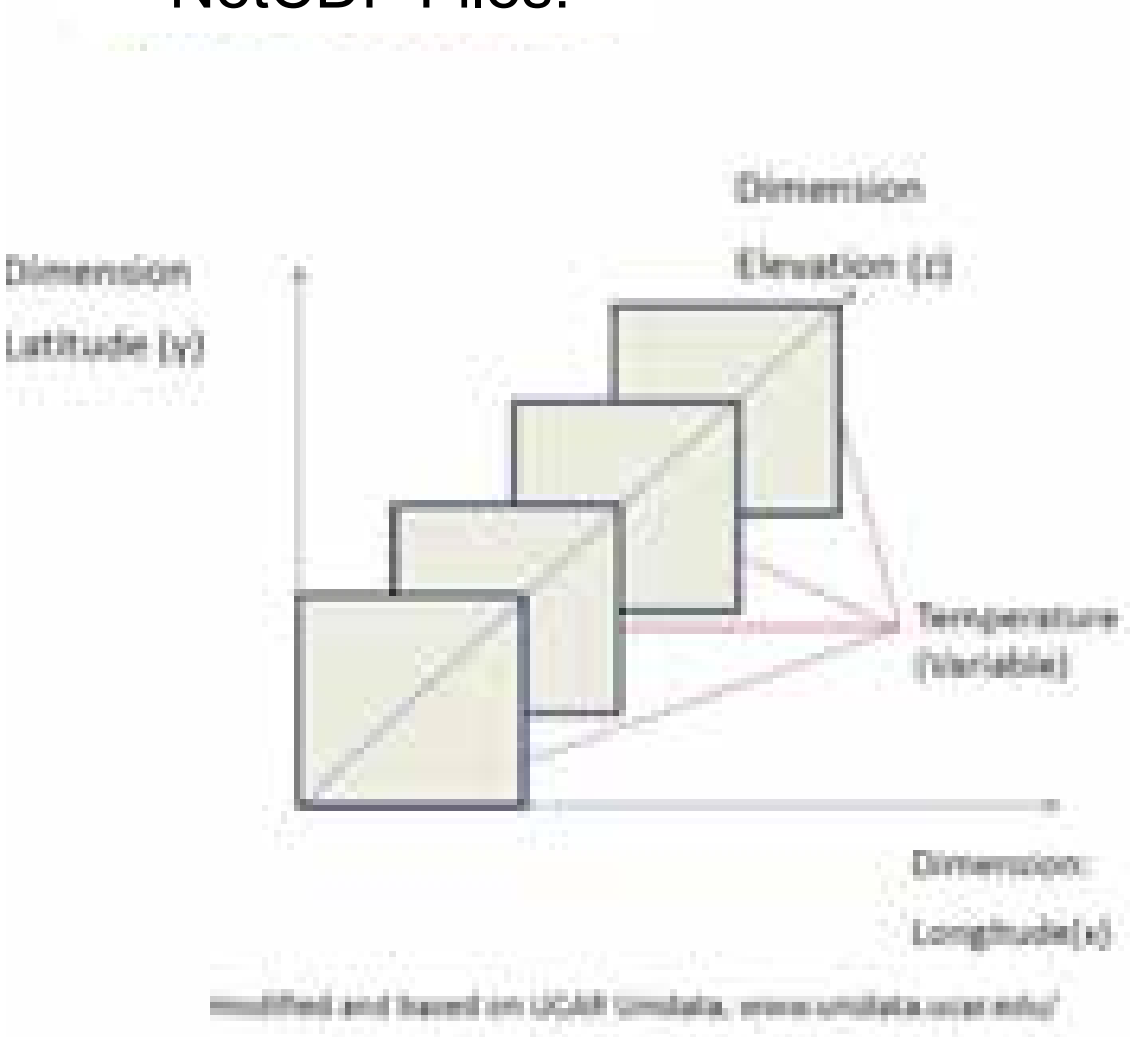


Slide by Chris Schubert

ifs FACULTY OF INFORMATICS

CCCA

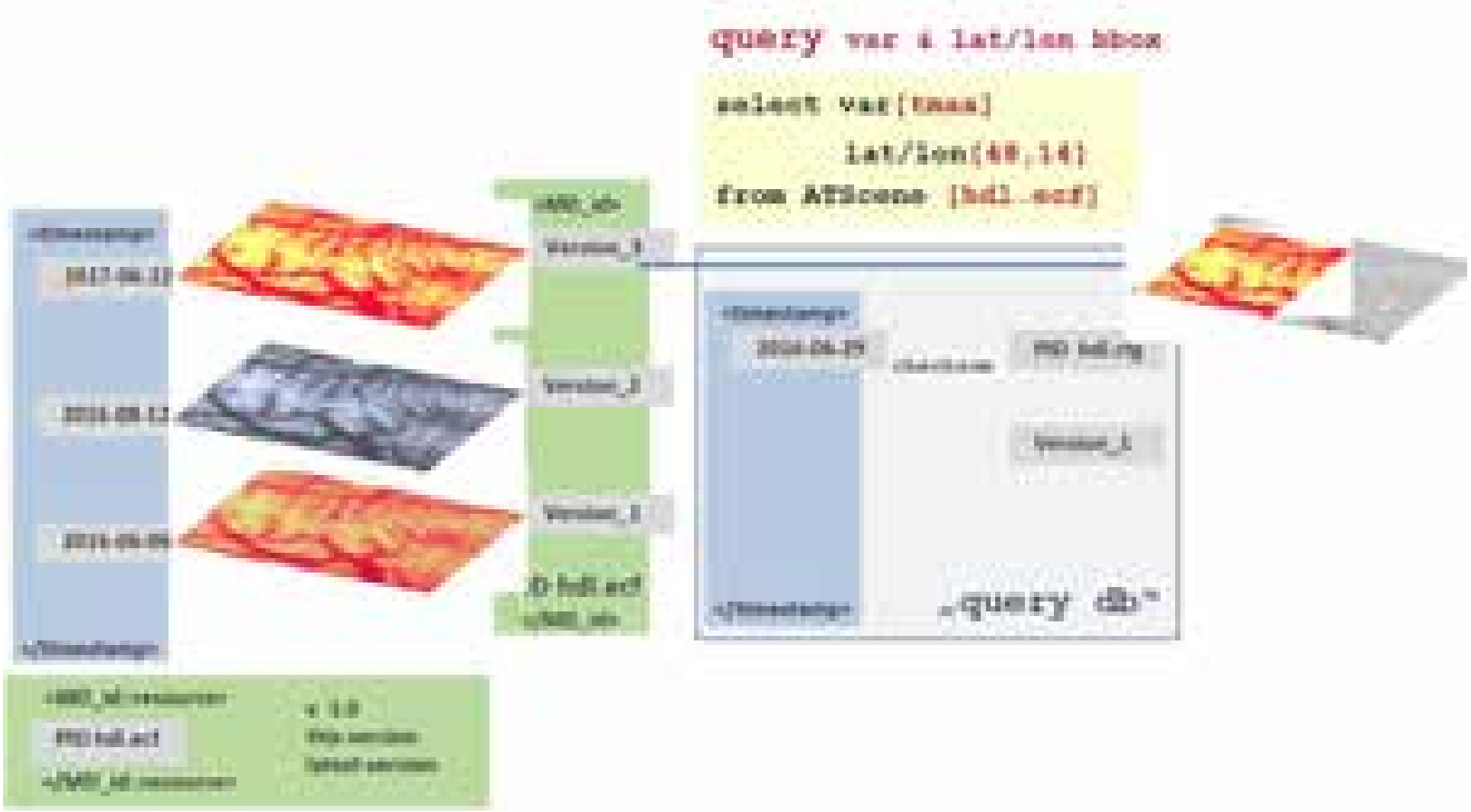
- NetCDF Files:



Slide by Chris Schubert

FACULTY OF INFORMATICS

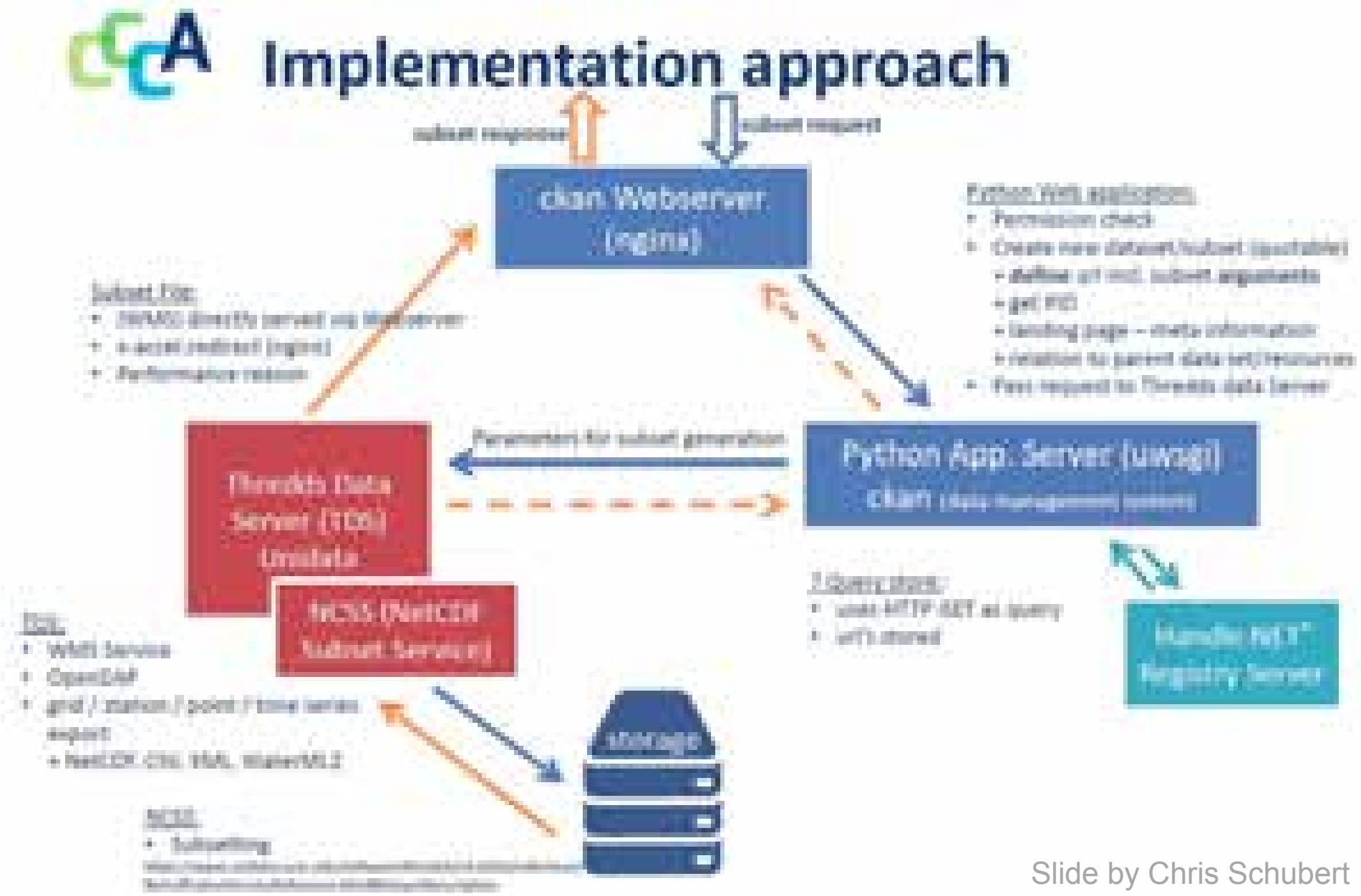
CCCA



Slide by Chris Schubert

FACULTY OF INFORMATICS

CCCA



Slide by Chris Schubert

CCCA

- for subsetting datasets
- uses **HTTP GET** as query in following scheme:
  http://{host}/{context}/{service}/{**dataset**}[/dataset.html | {?query}]

Subsetting parameter used:

- **var** - names of our layer
- **north, south, east, west** - for the geographical extend, the bounding box
- **time_start, time_end, time_duration** - for time extend, limited only on 5 years interval
- **accept** - specify the returned format

All "http get" stored as url in our ckan data store

PID:
hdl.handle.net/20.500.11756/93887acf

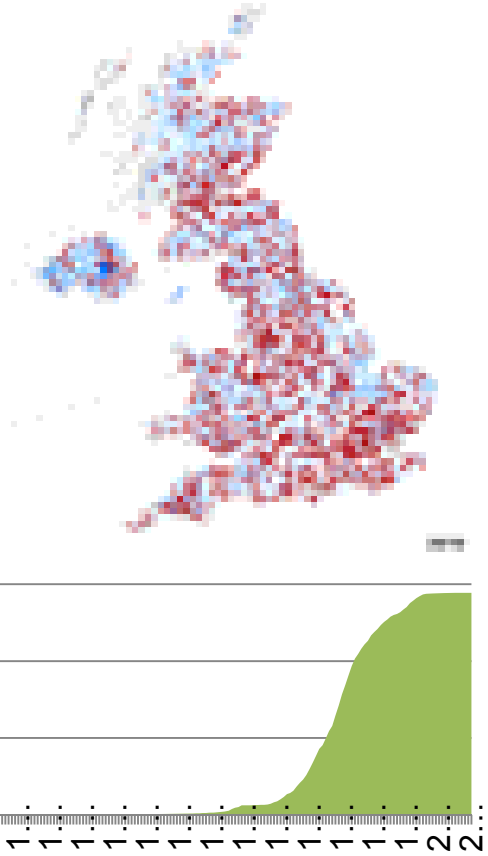
https://data.ccca.ac.at/tds_proxy/ncss/**1d8a5262-4580-4fa1-a1ac-cfb0b94a7670**?**north**=47.711688225506**&west**=9.023105988277664**&accept**=netCDF**&var**=tas**&east**=12.03185990452766**&south**=46.77724203092812

Slide by Chris Schubert


RDA
RESEARCH DATA ALLIANCE
UK Riverflow Archive
Matthew Fry
mfry@ceh.ac.uk
research data sharing without barriers
rd-alliance.org

UK National River Flow Archive

- Curation and dissemination of regulatory river flow data for research and other access
- Data used for significant research outputs, and a large number of citations annually
- Updated annually but also regular revision of entire flow series through time (e.g. stations resurveyed)
- Internal auditing, but history is not exposed to users



UK National River Flow Archive

- ORACLE Relational database
- Time series and metadata tables
- ~20M daily flow records, + monthly / daily catchment rainfall series
- Metadata (station history, owners, catchment soils / geology, etc.)
- Total size of ~5GB
- Time series tables automatically audited,
  - But reconstruction is complex
- Users generally download simple files
- But public API is in development / R-NRFA package is out there
- Fortunately all access is via single codeset

Our data citation requirements

- Cannot currently cite whole dataset
- Allow citation a subset of the data, as it was at the time
- Fit with current workflow / update schedule, and requirements for reproducibility
- Fit with current (file download) and future (API) user practices
- Resilient to gradual or fundamental changes in technologies used
- Allow tracking of citations in publications

Options for versioning / citation

- “Regulate” queries:
  - limitations on service provided
- Enable any query to be timestamped / cited / reproducible:
  - does not readily allow verification (e.g. checksum) of queries (R7), or storage of queries (R9)
- Manage which queries can be citable:
  - limitation on publishing workflow?

Versioning / citation solution

- Automated archiving of entire database – version controlled scripts defining tables, creating / populating archived tables (largely complete)
- Fits in with data workflow – public / dev versions – this only works because we have irregular / occasional updates
- Simplification of the data model (complete)
- API development (being undertaken independently of dynamic citation requirements):
  - allows subsetting of dataset in a number of ways – initially simply
  - need to implement versioning (started) to ensure will cope with changes to data structures
- Fit to dynamic data citation recommendations?
  - Largely
  - Need to address mechanism for users to request / create citable version of a query
- Resource required: estimated ~2 person months

Outline

- Why should we want to cite data?
- What are the challenges in data identification and citation?
- How should we do it, according to the RDA WG?
- Who is doing it so far, and how?
- Summary

Summary

- Data citation essential for **solid** and **efficient** science (but not just for science!)
- It is more than just giving credit
- Human-readable and machine-actionable
- RDA recommendations
  - Time-stamp and version data if it is evolving
  - Provide PIDs to arbitrary subsets via selection mechanism (“query”) (rather than statically assigned PIDs to pre-defined subsets)
- 2 PIDs:
  - for evolving intellectual object
  - for precise, static subset

Benefits

- Retrieval of precise subset with low storage overhead
- Subset as cited or as it is now (including e.g. corrections)
- Query provides provenance information
- Query store supports analysis of data usage
- Fixity information (hash/checksum) supports verification
- Same principles applicable across all settings
  - Small and large data
  - Static and dynamic data
  - Any subset (including empty set!)
  - Different data representations (RDBMS, CSV, XML, LOD, …)
- Relatively straight-forward to implement

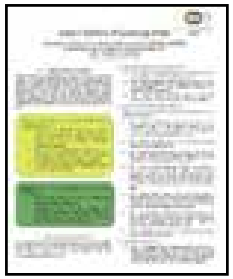
Interested in implementing?

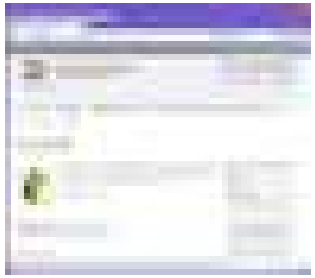
- If you have a **dataset** / are operating a **data center** that
  - has **dynamic data**

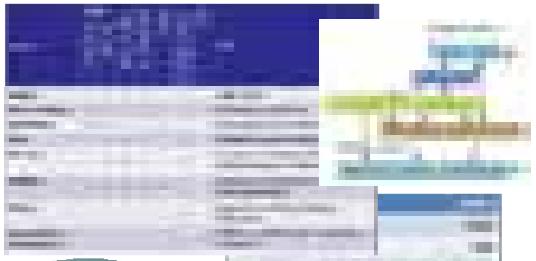
  and / or
  - allows researchers to **select subsets** of this data for studies
- and would **like to support precise identification / citability**
- Let us know!
  - Join RDA WGDC: https://www.rd-alliance.org/groups/data-citation-wg.html
  - Support for adoption pilots
  - Cooperate on implementing and deploying recommendations
- Collecting feedback, identifying potential issues, …

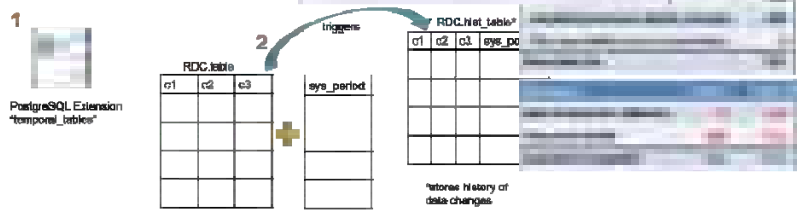
ifs FACULTY OF INFORMATICS

Thank you!









Thanks!

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