Preserving Complex Scientific Objects: Process Capture and Data Identification

Andreas Rauber
J.Binder, T.Miksa, R.Mayer, S.Pröll, S.Strodl, M.Unterberger
Vienna University of Technology
&
Secure Business Austria
rauber@ifs.tuwien.ac.at
http://www.ifs.tuwien.ac.at/~andi
Outline

- What is the “Complex Scientific Object” to preserve?
- How to capture a process and its context?
- How can we precisely identify the data used?
- Summary
Preserving Research

- Why do we want to preserve research/scientific objects?

- Repeatability
- Verification
- Trust
- Reuse

Documentation
Preserving Research

- Preservation:
  - “keeping useable over time”
  - fighting technical & semantic obsolescence

- Research:
  Which “Scientific Objects”

- What have we got?
  - Research Objects
  - Repositories for papers, data and code
  - Data Management Plans

Done?
Excursion: Scientific Processes
From Data to Processes

- Excursion: scientific processes

set1_freq440Hz_Am11.0Hz

set1_freq440Hz_Am12.0Hz

set1_freq440Hz_Am05.5Hz

Java

Matlab
Excursion: Scientific Processes

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

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

The Effects of FreeSurfer Version, Workstation Type, and Macintosh Operating System Version on Anatomical Volume and Cortical Thickness Measurements

Ed H. B. M. Gionenschild, Petra Habets, Heidi I. L. Jacobs, Ron Mangelers, Nico Rozendaal, Jim van Os, Machiel Marcelis

Abstract

FreeSurfer is a popular software package to measure cortical thickness and volume of neuroanatomical structures. However, little if any is known about measurement reliability across various data processing conditions. Using a set of 30 anatomical T1-weighted 3T MRI scans, we investigated the effects of data processing parameters such as FreeSurfer version (v4.3, v5.0, and v5.6.0), workstation (Macintosh and Hewlett-Packard), and Macintosh operating system version (OSX 10.5 and OSX 10.6). Significant differences were revealed between FreeSurfer version v5.0.0 and the two earlier versions. These differences were on average 8.8±6.6% (range 1.3–64.0%) (volume) and 2.8±1.3% (1.1–7.7%) (cortical thickness). About a factor two smaller differences were detected between Macintosh and Hewlett-Packard workstations and between OSX 10.5 and OSX 10.6. The observed differences are similar in magnitude as effect sizes reported in accuracy evaluations and neurodegenerative studies.

The main conclusion is that in the context of an ongoing study, users are discouraged to update to a new major release of either FreeSurfer or operating system or to switch to a different type of workstation without repeating the analysis; results thus give quantitative support to successive recommendations stated by FreeSurfer developers over the years. Moreover, in view of the large and significant cross-version differences, it is concluded that formal assessment of the accuracy of FreeSurfer is desirable.
A simpler example

- Image conversion from jpg to tiff using *ImageMagick*

<table>
<thead>
<tr>
<th></th>
<th>View Path #1</th>
<th>View Path #2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data formats</strong></td>
<td>Raw JPEG Stream (fmt/41); Portable Network Graphics (fmt/13)</td>
<td>Raw JPEG Stream (fmt/41); Portable Network Graphics (fmt/13)</td>
</tr>
<tr>
<td><strong>Application</strong></td>
<td>ImageMagick 6.8.9-7 Q16 Microsoft Visual C++ 2010</td>
<td>ImageMagick 6.8.9-7</td>
</tr>
<tr>
<td><strong>JVM</strong></td>
<td>Java SE 6 Update 45</td>
<td>Java SE 7 Update 10</td>
</tr>
<tr>
<td><strong>Operating System</strong></td>
<td>Windows 7 Enterprise SP1</td>
<td>OS X 10.9.4</td>
</tr>
<tr>
<td><strong>Hardware</strong></td>
<td>3.3GHz Intel Core i3 8GB 1600MHz DDR3 NVIDIA GT630 2GB</td>
<td>2.3GHz Intel Core i5 4GB 1333MHz DDR3 Intel HD Graphics 3000 384MB</td>
</tr>
</tbody>
</table>
From Data to Processes

Original jpg

TIFF Migration on Windows 7

Diff

TIFF Migration on OSX
Process Management Plans

- Need to preserve the process, not (only) the outputs!

- “Process Management Plans” (PMPs)?
  - Go beyond data (DMPs) to cover research process:
    - ideas, steps, tools, documentation, results, …
    - data is only one (important) element,
      usually a result of a research (pre-)process
  - Ensure re-executability, re-usability
  - Should be machine-actionable & verifiable
  - Basis for preservation and re-use of research
  - Similar to “research objects”, “executable papers”, …
  - Should be created semi-automatically
Outline

- What is the “Complex Scientific Object” to preserve?
- How to capture a process and its context?
- How can we precisely identify the data used?
- Summary
Process Management Plans

Need to create

- Models for representing such “process management plans” (PMPs)
- Should be machine-readable and machine-actionable
- Identify “minimum set” of information
- Devise means to automate (most of) the activity in creating and maintaining those PMPs
- Establish them to replace (enhance / subsume / …) Data Management Plans
Process Management Plans

Structure of PMPs (following concept of DMPs):

1. Overview and context
2. Description of process and its implementation
   - Process description | Process implementation | Data used and produced by process
3. Preservation
   - Preservation history | Long term storage and funding
4. Sharing and reuse
   - Sharing | Reuse | Verification | Legal aspects
5. **Monitoring** and external dependencies
6. Adherence and Review
Process Context Model

- Establish what to document and how: Context Model
- Meta-model for describing process & context
  - Extensible architecture integrated by core model
  - Reusing existing models as much as possible
  - Implemented using OWL
Application Example: Steps

- Acquisition of music & ground-truth data
- Extraction of numeric features
- Training of machine learning model
- Analysis of classification performance
- Repetition of experiment with variations
  - Finally leading to publication
Automatic Model Generation

- Bottom up: tracing of specific execution
  - Captures all resources accessed (files, ports, ...)
  - Linux prototype (http://ifs.tuwien.ac.at/dp/process/projects/pmf.html)
  - Captures verification data of process execution instance
Automatic Model Generation

- **Top-down:** capturing of execution environment
  - Software applications & dependencies (Linux Packages & Windows DLLs)
  - Licenses (mostly Open Source)
  - File Formats (DROID) & Link to registries (PRONOM)
  - Hardware (Linux & Windows)

http://opensourceprojects.eu/p/timbus/
Process Capture

Preservation and Re-deployment

- "Encapsulate" as complex Research Object (RO)
- DP: Re-Deployment beyond original environment
  - Format migration of elements of ROs
  - Cross-compilation of code
  - Emulation-as-a-Service
- Verification upon re-deployment
Outline

- What is the “Complex Scientific Object” to preserve?
- How to capture a process and its context?
- How can we precisely identify the data used?
- Summary
Data and Data Citation

- So far focus on the process
- Processes work with data
- Data as a “1st-class citizen” in science
- We need to be able to
  - preserve data and keep it accessible
  - cite data to give credit and show which data was used
  - identify precisely the data used in a study/process for repeatability, verifyability,…
- Why is this difficult?
  (after all, it’s being done…)
Granularity of Data Identification

- What about the **granularity** of data to be identified?
  - Databases collect enormous amounts of data over time
  - 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 cite precisely the subset of (dynamic) data used in a study
Identification of Dynamic Data

- Citable 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 cite precisely the **data as it existed at any point in time**
- Research Data Alliance
- WG on Data Citation: Making Dynamic Data Citeable
- WG officially endorsed in March 2014
  - Concentrating on the problems of large, dynamic (changing) datasets
  - Focus! Identification of data!
  - Not: PID systems, metadata, citation string, attribution, ...
  - Liaise with other WGs and initiatives on data citation (CODATA, DataCite, Force11, ...)

- https://rd-alliance.org/working-groups/data-citation-wg.html
Making Dynamic Data Citeable

Data Citation: Data + Means-of-access

- Data → time-stamped & versioned (aka history)

Researcher creates working-set via some interface:

- Access → 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

Prototype for CSV: http://datacitation.eu/
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
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

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

- Upon executing selection ("download") a 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

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!!!
Data Citation – Recommendations

- 2-page flyer, more extensive doc to follow
- **14 Recommendations**
- Grouped into **4 phases:**
  - Preparing data and query store
  - Persistently identifying specific data sets
  - Upon request of a PID
  - Upon modifications to the data infrastructure
- **History**
  - First presented March 30 2015
  - Major revision after workshop April 20/21
  - Series of webinars (next: June 24, 18:00 CEST)
Summary

- Trustworthy and efficient e-Science
- Need to move beyond preserving data
- Need to move beyond the focus on description
- Process Management Plans (PMPs)
- Preservation (and verification)
- Support for citing arbitrary subsets of dynamic data
- Data and process re-use as basis for data driven science
  - evidence
  - investment
  - efficiency

Done!?
Acknowledgements

- Johannes Binder
- Rudolf Mayer
- Tomasz Miksa
- Stefan Pröll
- Stephan Strodl
- Marco Unterberger

- TIMBUS
- SBA: Secure Business Austria
- RDA: Research Data Alliance
References

Thank you!

http://www_ifs_tuwien_ac_at/imp