Building Tools and Services to Support Research Software Preservation and Sharing
Software in the scholarly record

Nature’s top 100 papers:

“…the vast majority describe experimental methods or software that have become essential in their fields.”

At least 5 out of the top 20 describe software

http://www.nature.com/news/the-top-100-papers-1.16224
The problem...

CNI Spring 2016 plenary (Victoria Stodden):

“The Scholarly Record comprises access to, and/or the ability to regenerate, items relied on in the generation of stated results.”

“At present, access to ‘items’ underlying computational results is limited.”

“Access”

- Capture
- Preservation
- Policy
- Incentives
- Community support/advocacy

→ reproducibility, reuse
Micah Altman
Program on Information Science
MIT Libraries

Related Publications:
Informatics.mit.edu

Questions:
escience@mit.edu
DISCLAIMER

These opinions are my own, they are not the opinions of MIT, any of the project funders, nor (with the exception of co-authored previously published work) my collaborators

Secondary disclaimer:

“It’s tough to make predictions, especially about the future!”

-- Attributed to Woody Allen, Yogi Berra, Niels Bohr, Vint Cerf, Winston Churchill, Confucius, Disreali [sic], Freeman Dyson, Cecil B. Demille, Albert Einstein, Enrico Fermi, Edgar R. Fiedler, Bob Fourer, Sam Goldwyn, Allan Lamport, Groucho Marx, Dan Quayle, George Bernard Shaw, Casey Stengel, Will Rogers, M. Taub, Mark Twain, Kerr L. White, etc.
Collaborators & Co-Conspirators

• Collaborators in research:
  Alex Chassanoff; Nancy McGovern
• Thanks to:
  Lucy Taylor – Graduate Research Intern
• Research Support
  – Supported in part by the Sloan Foundation;
    CLIR; Massachusetts Institute of Technology
Why Software?

Internet Pioneer Warns Our Era Could Become The 'Digital Dark Ages'

Cluster failure: Why fMRI inferences for spatial extent have inflated false-positive rates

Anders Eklund*a,b,c,1, Thomas E. Nicholsa,c,d, and Hans Knutsson*a,c

*aDivision of Medical Informatics, Department of Biomedical Engineering, Linköping University, S-581 83 Linköping, Sweden; bDivision of Statistics and Machine Learning, Department of Computer and Information Science, Linköping University, S-581 83 Linköping, Sweden; cCenter for Medical Image Science and Visualization, Linköping University, S-581 83 Linköping, Sweden; dDepartment of Statistics, University of Warwick, Coventry CV4 7AL, United Kingdom; and eWMG, University of Warwick, Coventry CV4 7AL, United Kingdom

Edited by Emery N. Brown, Massachusetts General Hospital, Boston, MA, and approved May 17, 2016 (received for review February 12, 2016)
What is Software

Working definition:
“Part of a computer system that consists of encoded information or computer instructions” (wikipedia) that is directly executable within a system.

Corollaries

- Software generally is composed of instantiations of algorithms, heuristics, and fixed information (internal data).
- The behavior and output of software generally depends on the execution context: execution environment (software, hardware, network, networked resources), configuration parameters, and dynamic inputs.

Software curation:

- encompasses-addresses the active practices necessary/related to the creation/acquisition, appraisal/selection, description, transformation, preservation, storage, and dissemination/access/reuse of software over short- and long- periods
Modeling Methods, Analysis & Data...

Theory
(Rules, Entities, Concepts)

Algorithms
(Protocol, Operationalization)

Implementations
(Software, Coding Rules, Instrumentation Design)

Executions
(Deployment, House Survey Style, Operating System, Instrument, Computer, Starting Values, PRNG seeds)

Execution Context
(weather, compiler, Operating System, system load)

Structure

Formats

Versions/Revisions

Selections

Integrations

Instantiations
(copies)

Modeling Reproducibility from an Informatics Perspective
Some Caution About Definitions

"... if they [philosophers] do ask and they want a definition, they do not want the most natural definition, e.g. of 'chair' they do not want the definition 'something to sit on'. Why are they not satisfied with the normal definition of chair, or, to put the question in another way, why do they wish to ask for the definition of a physical object?"

Source: "From the Minutes of the Moral Science Club, 23.2.1939" in Wittgenstein in Cambridge (2008)

- Software is often tightly coupled to data
- Boundaries among software objects and systems are fuzzy & permeable
- Usefulness of software is strongly dependent on the intent of the user, knowledge and capabilities of user (documentation matters), and execution context.
## Selected software curation use case areas and goals

<table>
<thead>
<tr>
<th>Use Case Area</th>
<th>Goals</th>
</tr>
</thead>
</table>
| **Historic / cultural / social science**           | - historical/cultural/social science/organizational scholarship on software development  
- “” scholarship on use and experience of software by a community  
- “intrinsic value”/artistic value                  |
| **Replication and reproducibility**                | - evaluate evidence claims made in research  
- reduced deliberate research fraud  
- check reliability (robustness) of results  
- check validity accuracy of described methods (accuracy) |
| **Reuse**                                          | - increase speed of development  
- standards compliance  
- apply methodology to a different corpus  
- increased quality and dependability               |
| **Render other digital objects**                   | - renders other objects meaningful  
- see digital preservation use cases                 |
| **Legal**                                          | - record of licensing, ownership, copyright  
- manage legal risks/accountability  
- compliance with laws/funding mandates  
- reduce barriers to long-term access for other historic use, replication, reuse, rendering |
| **Citation and attribution**                       | - allocate credit/ evaluate individual academic career  
- analyze software development/history               |
Potential Research Questions

- How is software related to research formally disseminated?
- Which “repositories” (points for mid/long term publishing/access of software) are broadly recognized?
- What is the relative prevalence and affordances of “repositories” for software?
- What practices, requirements, or standards for software curation are recognized broadly?
- What models are broadly recognized for describing software, software development, and curation?
- How are researchers using software in their research activities? What are anticipated future use cases?
- How are libraries and archival institutions approaching curation of software?
- How do libraries and archives envision the role of software in their collections?
- What kinds of challenges do libraries and for archival institutions face in curating software?
- What curation practices can library/archival institutions embed into their stewardship activities?
- What do stakeholders (funding institutions, journals) value as metrics for good curation practices?
Preliminary Environmental Scan

Literature Review
- Data Curation, Publication and Citation
- Software research use cases
- Software repositories
- Software & scientific reproducibility
- Software Engineering Methodology

Web Research -- Repository Practice
- Review of research repositories & software directories
  - Sources: OpenDOAR, Re3Data, Sherpajuliet
  - Sources: OpenHub, OSDir, DMOZ
  - Goals: Estimate prevalence of repositories that accept research software

Web Research -- Research Software Policies
- Review of funder policies
  - Sources: Roarmap; US Federal Agency Websites
- Review of Journal Policies
  - Sources: Google Scholar, WoS, DOAJ, Software Sustainability Institute Index
Prevalence of software in research repositories appears low...
Most of what is in research repositories appears limited.
Preliminary findings: State of Software Curation

“Nothing Exists” - Parmenides (ca. 500 BCE)

1. No comprehensive indices of software archives
2. Orders of magnitude fewer software archives than data archives. (Corollary: most repositories containing software offer little functionality for discovering, understanding, or reusing it)
3. Very small proportion of funders have policies addressing software curation
4. There is little available advice for those who wish to curate, cite, & preserve software
Contrast with Industry -- Early Impressions

- Enterprise Software Management
  - Mature tools and practices for enterprise license management, deployment, inventory, updating
  - Models & documentation are proprietary, implicit, unstandardized
- Enterprise Software Development
  - Mature tools and practices for revision control; testing; QA
  - Standardized modeling language for modeling software design (e.g. UML) -- with broad tool support for large efforts
  - Less support for model-driven development, testing, deployment
  - Tools and practices focused on product development, QA, and internal reuse -- not on other use cases important to curation and research
Contrast with Data Curation
-- Lack of Progress

- Compliance
  - Funder: data management plans, open data
  - Publishers: data access/archiving/citation

- Norms & practices
  - Joint data citation principles
  - Recognition of data in funder biosketches
  - Increased recognition of reproducibility gaps
  - Increased recognition of open data/open science

- Technical infrastructure
  - Data repository directories
  - Data citation indices
  - ORCID researcher identifier and registry

- Recognition
  - Data citation indices
  - Virtual branded archives
  - High-profile data publications
Some Exemplars and Promising Initiatives

- Citation and publisher policies
  - FORCE 11 Software Citation Principles
    www.force11.org/software-citation-principles
  - ACM New Publication Policies on Software Reproducibility and Contributorship
    www.acm.org/publications/policies
  - PLOS
    journals.plos.org/plosone/s/materials-and-software-sharing
- Practices for reusable and reproducible software
  - Software carpentry:
    software-carpentry.org
  - Open software foundation:
    osf.io
- Long Term Access efforts:
  - www.softwareheritage.org
  - www.softwarepreservationnetwork.org
  - Github/Zenodo:
    guides.github.com/activities/citable-code
  - Internet Archive:
    archive.org/details/softwarelibrary
- Formal Software Publication:
  - Software X:
    www.journals.elsevier.com/softwarex
  - Journal of Statistical Software:
    www.jstatsoft.org/
  - Open Research Software:
    openresearchsoftware.metajnl.com
Takeaways...

- Software boundaries and definitions are often contested -- critical for curation efforts to identify targeted use cases
- Some mature practices and tools for software development, maintenance, and deployment exist in industry
- Practices are rapidly emerging in software citation
- There are number of early-stage initiatives in curation and preservations worth close attention
- Software curation for research, history and culture is at early stages -- trails data curation by a decade or more
- Curation efforts should recognize the practices and tools above, although they cover only a portion of the target use cases and lifecycle
Jeff Spies

- COS - Co-founder, Chief Technology Officer
- SHARE - Co-lead
- UVA - Asst. Prof., Dept. of Engineering & Society

jeff@cos.io  @jeffspies
Incentives for individual success are focused on getting it published, not getting it right.

Nosek, Spies, & Motyl, 2012
Context is important

• The problems we deal with are challenging
• We are human
  – Biased
  – Make decisions and interpret results based upon unique experiences
  – Influenced by incentives
  – Forgetful
  – Busy
We need context for . . .

- Reproducibility
- Replicability
- Extensibility
- Meta-scholarship

in order to increase research . . .

- Efficiency
- Quality
- Accessibility
- Diversity
Publish Report

Write Report

Search / Discovery

Develop Idea

Design Study

Store Data

Collect Data

Analyze Data
Computing is pervasive, and we must capture it as context of the scholarly workflow.
But this isn’t easy. Luckily, we have a partner:

DevOps
DevOps [emphasizes] collaboration and communication... while automating the process of software delivery and infrastructure changes... [such that] building, testing, and releasing software can happen rapidly, frequently, and more reliably.
We need context for . . .

- Reproducibility
- Replicability
- Extensibility
- Meta-scholarship

in order to increase research . . .

- Efficiency
- Quality
- Accessibility
- Diversity
DevOps needs context for software . . .

- Reproducibility
- Replicability
- Extensibility
- Management

in order to increase software . . .

- Efficiency
- Quality
- Accessibility
- Diversity
DevOps + Librarians =

Research Software Preservation and Sharing
http://osf.io
Collaboration
Documentation
Archiving
Study 3: Gupta et al. 2010, Nature

Contributors: Tim Errington, Elizabeth Iorns, William Gunn, Fraser Elisabeth Tan, Sarah Statt, Joelle Lomax, Nicole Perfito

Date Created: 2013-10-22 02:04 PM | Last Updated: 2015-01-20 06:16 PM

Category: Project

Wiki

This project contains all information pertaining to the replication of key experiments from this paper. It includes the detailed protocols, including reagents and author clarifications. We also include any comments from other contributors, researchers from the Science Exchange network, and further information with the original authors that we have learned since the beginning of the project. When experimental studies begin all data collected will also be deposited here, including data analysis...

Citation

Components

Coded Paper

Errington, Iorns, Gunn & 2 more

8 contributions

Recent Activity

All times displayed at -0700 UTC offset.

2015-01-20 06:16 PM  Tim Errington added Nicole Perfito as contributor(s) to
Version Control
### Component: Presentations

- **OSF Storage**
  - 2015.10.GHC.general.sharing.pptx
  - 20150107_cendi_spies.pptx
  - 20160128_uva_dev_psychology.pptx
  - 20160205_rpl_rcos_spies.pptx
  - Bowman.ACS.2015.08.17.pptx
  - Bowman.LJAF.2015.04.22...
Other Features

- Granular privacy/sharing
- Granular permissions
- Analytics dashboards
- Persistent, citable identifiers
- Persistent content
- Project snapshotting (i.e., registration)
- Licensing
- Forking
And we build other services with the OSF API (+ SHARE) like http://osf.io/preprints
Connects Services Researchers Use
Lepadogaster.stl
Publish Report

Search / Discovery

Develop Idea

Design Study

Collect Data

Store Data

Analyze Data

Write Report

GitHub

ownCloud

Dropbox

amazon web services

box

Google Drive

OpenSesame

Now
But computing is pervasive....
ReproZip +

Fedora™
jeff@cos.io  @jeffspies
Rick Johnson

- Co-Program Director of Digital Initiatives and Scholarship, Hesburgh Libraries, University of Notre Dame
- Visiting Program Officer (VPO) for SHARE, Association of Research Libraries

rick.johnson@nd.edu  @rick_nd
Notre Dame Perspectives

- Computational Analysis Reproducibility is complex
- Need to align preservation with existing workflows
- Simple, user-driven archival process
- Data flow between computational and preservation environments
- Reuse preserved data and software in other projects
- Cannot be solved just with technology
Notre Dame Data & Software Capture and Reproducibility Efforts & Partners

http://daspos.org

http://cos.io

http://curate.nd.edu

Umbrella, http://ccl.cse.nd.edu/

http://www.nationaldataservice.org

http://www.library.jhu.edu
NDS OSF Dashboard integration

Contact:
Ian Taylor
ian.j.taylor@gmail.com

bitbucket.org/nds-org/nds-dashboard
http://www.nationaldataservice.org/
http://ndspilot.com
Computational and Preservation Environment Data Flow

GENERATE & STORE

COLLABORATE & PROCESS

PRESERVE & SHARE

amazon web services

box

Dropbox

CRC

CENTER FOR RESEARCH COMPUTING
https://crc.nd.edu

CurateND

EXECUTE

ARCHIVE

REUSE

https://www.nationaldataservice.org
Not Just Technology

Create OSF Project

Notify Data Mgmt Consulting Team & Other Partners

Data Management Consulting

Offer Consultation
## Data Management Consulting Team - Key Support Roles

<table>
<thead>
<tr>
<th>Role</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consultant</td>
<td>Provides relevant knowledge, experience, and task execution</td>
</tr>
<tr>
<td>Data Analyst</td>
<td>Technical coordinator for data products and other computational needs associated with the project</td>
</tr>
<tr>
<td>Developer</td>
<td>Technical coordinator/programmer for data products and other high level computational needs</td>
</tr>
<tr>
<td>Domain Expert*</td>
<td>Provides project-specific contextual and cultural expertise from the perspective of the related domain</td>
</tr>
<tr>
<td>Metadata Consultant</td>
<td>Oversees project metadata components</td>
</tr>
<tr>
<td>Policy Expert*</td>
<td>Ensures project is in compliance with applicable standards and regulations</td>
</tr>
<tr>
<td>Request Lead</td>
<td>Key point of contact/responsibility for project</td>
</tr>
</tbody>
</table>
ND Virtual Data Management Consulting Team

Core Team

Office of Research Consultant
Data Analyst
Subject Librarian
Other Domain Expert
Metadata Consultant
Technology Support
Supporting software access and preservation
A library RDM services perspective
Fernando Rios - rios@jhu.edu
CLIR Fellow - JHU Data Management Services

Data
Software

Consulting + training
+ archiving/sharing

Funder requirements for open access | Preserving the scholarly record
Publisher requirements for open access | Reproducing results
Need of specific research communities | Research efficiency via reuse
Starting from scratch

Problem: what do we even need to think about?

Scan the literature, work by different communities, current preservation offerings, technology, state of policy etc.

- Software not considered 1st class research output
- Availability of software in the literature is low
  - Even in computationally-oriented fields
- The biggest challenges are more cultural and legal in nature. Technical solutions are still not trivial (eg., use of virtualization/containers)

Identifying what we could support and how -- a plan
Publishing research openly as a condition of funding

Re-analyzing data in light of new theories

Promoting good software

Doi: 10.1045/july2016-rios
Plan

Initially, support outputs tied to publication, end-of-project

- Archiving only
- What software-specific metadata to include?
- Make use of existing data infrastructure

Extend internal expertise
Archiving

Metadata: minimal/useful metadata?

- What does it do, does it do what I want, how do I use it?

Looked at ~10 schemas
Archiving

Metadata for software is deceptively easy

- Defining who should count as an “author” or contributor is not trivial.
- Granularity: at what level should it be described?

Software is highly dynamic

Workflow integration is important

- “End of project” is only one of several points of preservation
Consulting Expertise

Publication support
SW Citation Principles →

Funder guidelines/DMPs, publishers

Understanding best practices for reproducible computational work

How to communicate about software preservation

https://doi.org/10.7717/peerj-cs.86
Summary

Extending RDM services to (basic) software services is doable w/o huge changes to existing data practices

Archiving is 1st step

- full workflow capture and replay would be a dream goal
- Service models?

Explore “curation-ready” software

- Involvement in active research phase