Improving Reproducibility Through Better Software Practices

Better Scientific Software Tutorial

David E. Bernholdt

Oak Ridge National Laboratory

ISC High Performance Conference

June 16, 2019

See slide 2 for license details

License, citation, and acknowledgments

License and Citation

- Requested citation: **David E. Bernholdt and Michael A. Heroux, Improving Reproducibility Through Better Software Practices, in Better Scientific Software Tutorial, ISC High Performance Conference, Frankfurt, Germany, 2019. DOI: https://doi.org/10.6084/m9.figshare.8242859**

Acknowledgements

- This work was supported by the U.S. Department of Energy Office of Science, Office of Advanced Scientific Computing Research (ASCR), and by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.
- This work was performed in part at the Oak Ridge National Laboratory, which is managed by UT-Battelle, LLC for the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.
- Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

Outline

- Reproducibility taxonomies.
- Increasing focus on reproducibility.
- Role of better software practices.
- Publication requirements.
- Trustworthiness at Scale.
- Personal Productivity Commitment.

Reproducible vs Replicable

Addressing Confusion in Taxonomies

SANDIA REPORT
SAND2018-11186
Unlimited Release
Printed October 2018

Toward a Compatible Reproducibility Taxonomy for Computational and Computing Sciences

Michael A. Heroux, Lorena A. Barba, Manish Parashar, Victoria Stodden and Michela Taufer

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

Table 1: Definitions of Reproducible and Replicable

Table 1: Claerbout/Donoho/Peng (Claerbout) and ACM definitions of Reproducible and Replicable. Claerbout definitions are prevalent in the computational science literature and have been used since the 1990s. The ACM definitions are used by ACM in its Artifact Review and Badging effort and first appeared in February 2013.

Term	Claerbout	ACM
Reproducible	Authors provide all the necessary data and the computer codes to run the analysis again, re-creating the results.	(Different team, different experimental setup.) The measurement can be obtained with stated precision by a different team, a different measuring system, in a different location on multiple trials. For computational experiments, this means that an independent group can obtain the same result using artifacts which they develop completely independently.
Replicable	A new study arrives at the same scientific findings as a previous study, collecting new data (with the same or different methods) and completes new analyses.	(Different team, same experimental setup.) The measurement can be obtained with stated precision by a different team using the same measurement procedure, the same measuring system, under the same operating conditions, in the same or a different location on multiple trials. For computational experiments, this means that an independent group can obtain the same result using the author's own artifacts.

Reproducibility is essential

Reproducibility

SCIENCE

Many Psychology Findings Not as Strong as Claimed

By BENEDICT CAREY AUG. 27, 2015

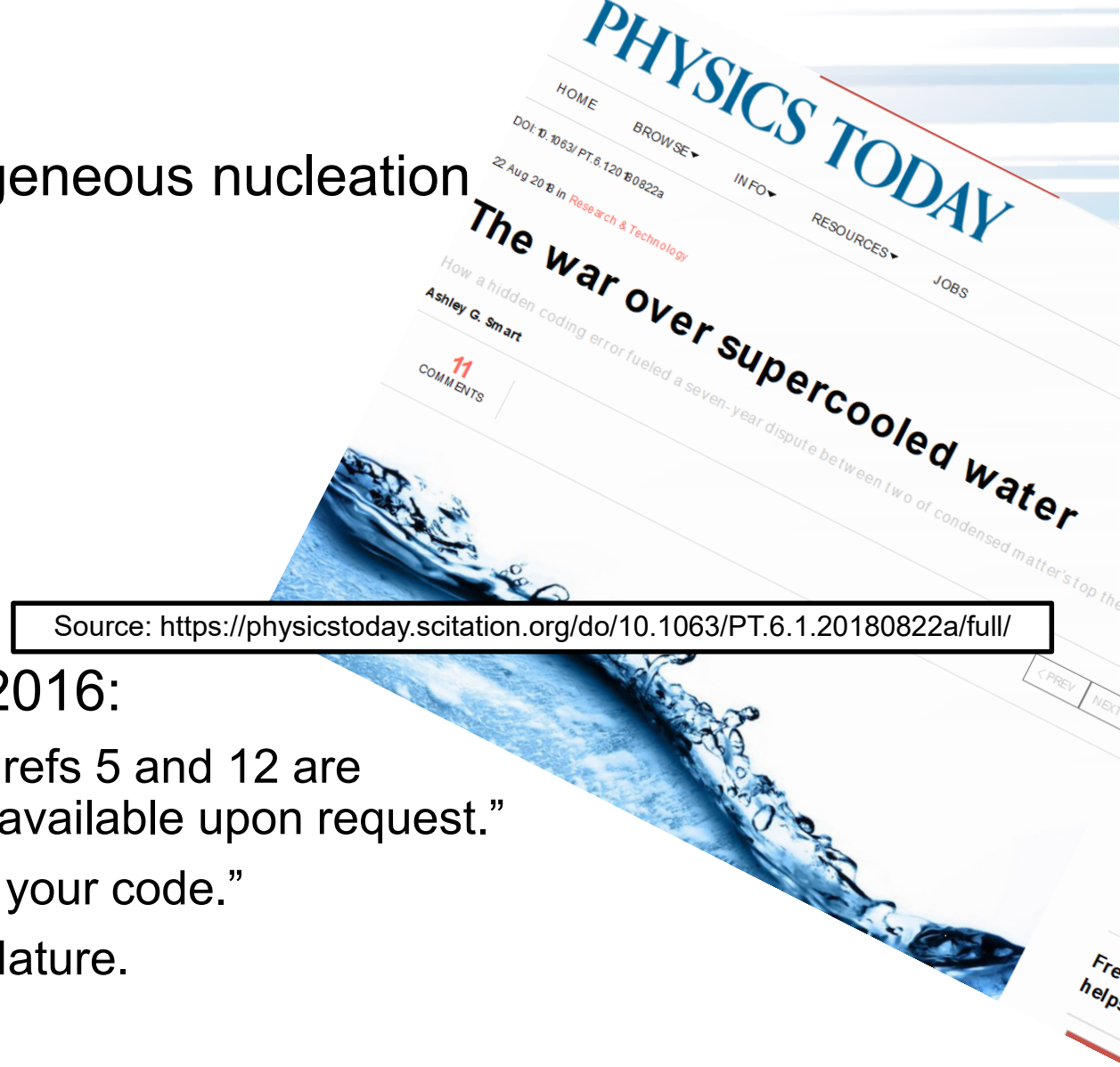
Staff of the the Reproducibility Project at the Center for Open Science in Charlottesville, Va., from left: Mallory Kidwell, Courtney Soderberg, Johanna Cohoon and Brian Nosek. Dr. Nosek and his team led an attempt to replicate the findings of 100 social science studies. Andrew Shurtleff for The New York Times

- NY Times highlights “problems”.
- Only one of many cited examples.
- Computational science ***had*** been spared this “spotlight”.

http://www.nytimes.com/2015/08/28/science/many-social-science-findings-not-as-strong-as-claimed-study-says.html?_r=0

Computational Science Example

- Behavior of pure water just above homogeneous nucleation temperature (~ - 40 C/F).
- Debenedetti/Princeton (2009):
  - 2 possible phases: High or low density.
- Chandler/Berkeley (2011):
  - Only 1 phase: High density.
- No sharing of details across teams until 2016:
  - Chandler in Nature: “LAMMPS codes used in refs 5 and 12 are standard and documented, with scripts freely available upon request.”
  - Debenedetti with colleague Palmer: ”Send us your code.”
  - Received code after requests and appeal to Nature.



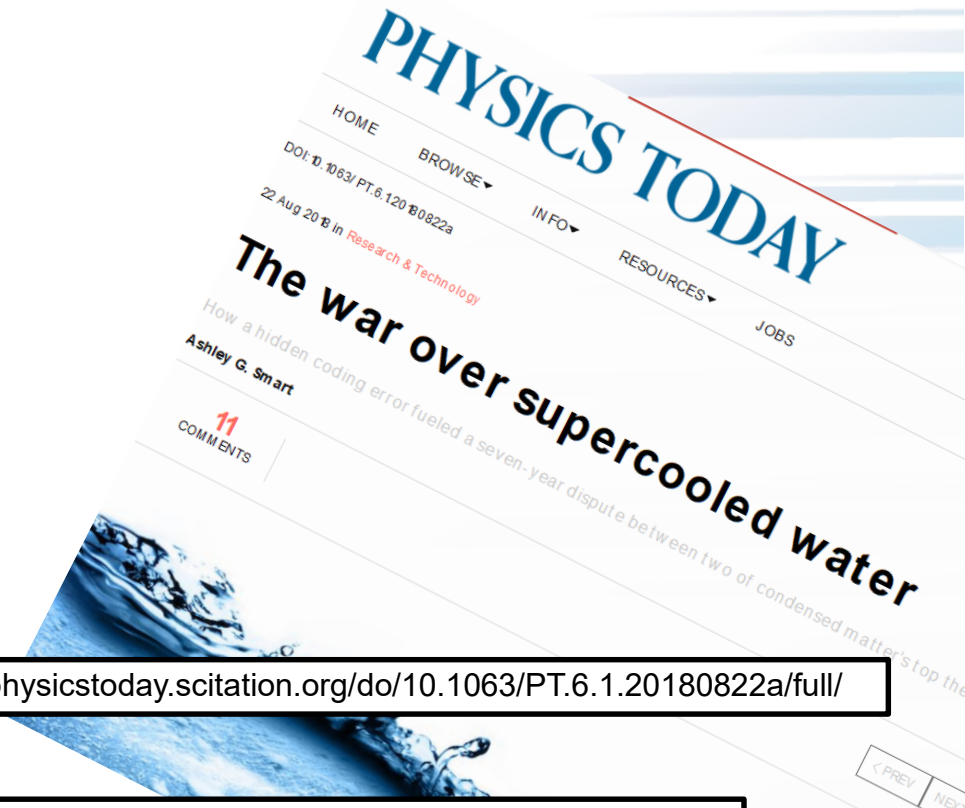
Source: https://physicstoday.scitation.org/do/10.1063/PT.6.1.20180822a/full/

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

Computational Science Example

- Palmer located bug/feature in Berkeley code.
- Used to speed up LAMMPS execution.
- Replaced with more standard approach.
- Obtained result similar to Debenedetti 2009.
- Resolution took 7 years.



Source: https://physicstoday.scitation.org/do/10.1063/PT.6.1.20180822a/full/

For Palmer, the ordeal exemplifies the importance of transparency in scientific research, an issue that has recently drawn heightened attention in the science community. "One of the real travesties," he says, is that "there's no way you could have reproduced [the Berkeley team's] algorithm—the way they had implemented their code—from reading their paper." Presumably, he adds, "if this had been disclosed, this saga might not have gone on for seven years."

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT

Productivity and Sustainability

What do we mean?

Objectives

- Productivity – Output per unit input.
- Sustainability – The future cost of usability.
- Goals for today:
  - Learn how to improve
    - Developer productivity.
    - Software sustainability.
  - For the purposes of better scientific productivity,
  - Using tools, processes and practices.

Tradeoffs: Better, faster, cheaper

- "Better, faster, cheaper: Pick two of the three."
  - Scenario: (Today)
    You are behind in developing a sophisticated new model in your software that you want to use for results in an upcoming paper.
  - Which of these could be reasonable choices?
    - Develop a simpler model for the paper.
    - Set other work aside and spend more time on development.
    - Ask for an extension on the paper deadline.
    - Develop sophisticated model, but don't test its correctness.
    - Develop sophisticated model, but don't document it or check it in.

Improved developer productivity

"Better, faster, cheaper: Pick all three." – Near term.

Scenario: (6 months later)
After investing in **developer productivity improvements**, you are on time in developing a sophisticated new model in your software that you want to use for results in an upcoming paper.

Invest in developer tools, processes, practices.

Improved software sustainability

"Better, faster, cheaper: Pick all three." – Long term.

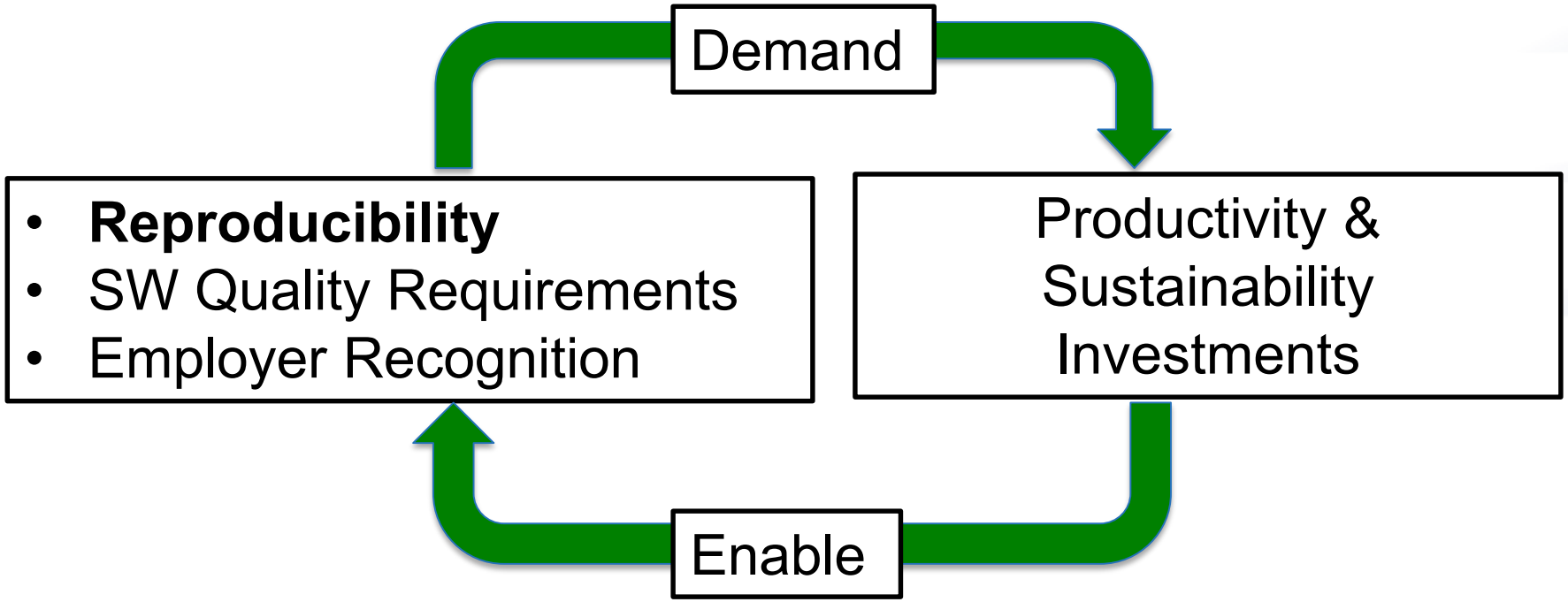
Scenario: (3 years later)
After investing in **software sustainability improvements**, you are on time in developing **several** sophisticated new models in your software that you want to use for results in upcoming papers.

Invest in testing, documentation, integration for long-term software usability.

Which of These Enhance Reproducibility?

- Code written by first-year, untrained grad student.
- Tuning for high performance.
- Dynamic parallelism of modern processors.
- Better software testing.
- Source code and versioning management.
- Investing in developer productivity.
- Investing in software sustainability.

Incentives To Change



Common statement: “I would love to do a better job, but I need to:

- Get this paper submitted.
- Complete this project task.
- Do something my employer values more.

Goal: Change incentives to include value of better software.

Reproducibility Terminology

V. Stodden, D. H. Bailey, J. Borwein, R. J. LeVeque, W. Rider, and W. Stein. 2013. Setting the Default to Reproducible: Reproducibility in Computational and Experimental Mathematics. (2013). https://icerm.brown.edu/tw12-5-rcem/icerm_report.pdf

- **Reviewable Research.** The descriptions of the research methods can be independently assessed and the results judged credible. (This includes both traditional peer review and community review, and does not necessarily imply reproducibility.)
- **Replicable Research.** Tools are made available that would allow one to duplicate the results of the research, for example by running the authors' code to produce the plots shown in the publication. (Here tools might be limited in scope, e.g., only essential data or executables, and might only be made available to referees or only upon request.)
- **Confirmable Research.** The main conclusions of the research can be attained independently without the use of software provided by the author. (But using the complete description of algorithms and methodology provided in the publication and any supplementary materials.)
- **Auditable Research.** Sufficient records (including data and software) have been archived so that the research can be defended later if necessary or differences between independent confirmations resolved. The archive might be private, as with traditional laboratory notebooks.
- **Open or Reproducible Research.** Auditable research made openly available. This comprised well-documented and fully open code and data that are publicly available that would allow one to (a) fully audit the computational procedure, (b) replicate and also independently reproduce the results of the research, and (c) extend the results or apply the method to new problems.

ACM TOMS Replicated Computational Results (RCR)

- Submission: Optional RCR option.
- Standard reviewer assignment: Nothing changes.
- RCR reviewer assignment:
  - Concurrent with standard reviews.
  - As early as possible in review process.
  - Known to and works with authors during the RCR process.
- RCR process:
  - Multi-faceted approach, Bottom line: Trust the reviewer.
- Publication:
  - Replicated Computational Results Designation.
  - The RCR referee acknowledged.
  - Review report appears with published manuscript.

SC18 Reproducibility Initiative

- Two appendices:
  - Artifact description (AD).
    - Blue print for setting up your computational experiment.
    - Makes it easier to rerun computations in future.
    - AD appendix will be mandatory for SC19 paper submissions.
  - Artifact Evaluation (AE).
    - Targets ”boutique” environments.
    - Improves trustworthiness when re-running hard, impossible.
- Details:
  - https://collegeville.github.io/sc-reproducibility/

Coming to Your World Soon: Reproducibility Requirements

- These conferences expect artifact evaluation appendices (most optionally):
  - CGO, PPoPP, PACT, RTSS and SC.
  - http://fursin.net/reproducibility.html
- ACM Replicated Computational Results (RCR).
  - ACM TOMS, TOMACS.
  - http://toms.acm.org/replicated-computational-results.cfm
- ACM Badging.
  - https://www.acm.org/publications/policies/artifact-review-badging

How can you prepare?

Improving Trustworthiness at Scale

What if we can’t re-run a computational experiment?

Reproducibility and Supercomputing

Scenario:
You compute a “hero” calculation using 5M core-hours on Mira and submit your results for publication. During the review process, a referee questions the validity of your results. What options are feasible:

- The reviewer re-runs your code on a laptop or cluster.
- The reviewer re-runs your code on Mira.
- You re-run your code on Mira.
- Your results are rejected.
- Your results are accepted, but with risk.

Sources for meta-computations

- Synthetic operators with known:
  - Spectrum (Huge diagonals).
  - Rank (by constructions).
- Invariant subspaces:
  - Example: Positional/rotational invariance (structures).
- Conservation principles:
  - Example: Flux through a finite volume.
- General:
  - Pre-conditions, post-conditions, invariants.

Can you think of something for your problems?

Personal Expectations

Calling out the best in team members

A Few Concrete Recommendations

Show me the person making the most commits on an undisciplined software project and I will show you the person who is injecting the most technical debt.

- GitHub stats: Easy to find who made the most commits.
  - Some people: Pride in their high ranking.

- Instead, be the person who ranks high in these ways:
  - Writes up requirements, analysis and design, even if simple.
  - Writes good GitHub issues, tracks their progress to completion.
  - Comments on, tests and accepts pull requests.
  - Provide good wiki, gh-pages content, responses to user issues.

(Personal) Productivity++ Initiative

Ask: *Is My Work _____ ?*

https://github.com/trilinos/Trilinos/wiki/Productivity---Initiative

Summary

- Reproducibility demands are coming.
  - Conferences first, journals slower.
- HPC software is particularly challenging:
  - Hardware variation.
  - Code optimization.
  - Dynamic parallelism.
- Better software practices:
  - Improve chances for reproducibility.
  - Lower its cost.
- Many tools emerging to enable reproducibility.

Agenda

Time	Module	Topic	Speaker
2:00pm-2:40pm	01	Overview of Best Practices in HPC Software Development	Anshu Dubey, ANL
2:40pm-3:20pm	02	Better (Small) Scientific Software Teams	David E. Bernholdt, ORNL
3:20pm-4:00pm	03	Improving Reproducibility through Better Software Practices	David E. Bernholdt, ORNL
4:00pm-4:30pm		*Break*	
4:30pm-5:15pm	04	Verification & Refactoring	Anshu Dubey, ANL
5:15pm-6:00pm	05	Git Workflow & Continuous Integration	Jared O'Neal, ANL

https://r.isc-hpc.com/tut130

(Please note the R before our domain)

IDEAS productivity

ECP EXASCALE COMPUTING PROJECT