Fighting the Reproducibility Crisis

Sustainable research software and RRR for computer-based experiments

Jens Saak

2020-04-21

COMPUTE Seminar
Lund University

Supported by:

"Sustainability of research software" call
pyMOR — Sustainable Software for Model Order Reduction
1. Motivation

2. RRR to FAIR

3. Proposed Development Practices
The Crew

Jörg Fehr
Uni Stuttgart

Jan Heiland
MPI Magdeburg

Christian Himpe
MPI Magdeburg

Stephan Rave
Uni Münster

Jens Saak
MPI Magdeburg

Jens Saak, saak@mpi-magdeburg.mpg.de

Fighting the Reproducibility Crisis
Together about one century of programming experience
"Tower of Doom" (by: S. Rave)
“The Void” (by: C. Himpe)
Our Aim

Improve Computer-Based Experiments (CBEx):

- Create problem-awareness and
- Ensure *scientificity* and *progress*
Our Aim

Improve Computer-Based Experiments (CBEx):

- Create problem-awareness and
- Ensure *scientificity* and *progress*
- Define terminology
- Establish best-practices
- Formulate discipline-agnostic practical guidelines
Our Aim

Improve Computer-Based Experiments (CBEx):

- Create problem-awareness and
- Ensure scientficity and progress
- Define terminology
- Establish best-practices
- Formulate discipline-agnostic practical guidelines
- Improve availability and quality of research software

Jens Saak, saak@mpi-magdeburg.mpg.de
What is a CBEx?

What is a scientific CBEx?
What is a CBEx?

- Any result obtained by a computer.
- No matter if it is:
  - supporting or illustrative results,
  - pointwise confirmation,
  - or computational proof.

What is a scientific CBEx?
What is a CBEx?

- Any result obtained by a computer.
- No matter if it is:
  - supporting or illustrative results,
  - pointwise confirmation,
  - or computational proof.

What is a scientific CBEx?

- Any CBEx by which the authors’ claim is verifiable.
Sorted by increasing commonality:

- Hardware not available
- Software stack not available
- Reporting not sufficient
- Archiving not stable
- Provisioning not sufficient
- Lack of education
The following is not a strict set of rules.

View it as a collection of best-practices.

Adapt these ideas to your use-case.
RRR to FAIR

based on

1. Replicability
2. Reproducibility
3. Reusability

Each R has:
- Minimal requirements
- Optional recommendations
The attribute **Replicability** describes the ability to repeat a CBEx and to come to the same (in a numerical sense) results. Sometimes the equivalent term **Repeatability** is used for this experimental property.

- **Replicability** is a basic requirement of reliable software as well as of its result as it shows a certain **robustness** of the procedure against
  - statistical influences
  - and bias of the observer.

- Also, **only replicable CBEx can serve as a benchmark** to which new methods can be compared, cf. [Vitek & Kalibera ’11].
The Essence of Replicability (aka Repeatability)

- You are able
- to repeat
- your experiment
- on your computer.
The Essence of Replicability (aka Repeatability)

- You are able
- to repeat
- your experiment
- on your computer.

Minimal Requirements

Basic Documentation:
- Recipe to obtain (numerical) results
- Recipe for post-processing of data
- Recipe for creating visualizations
The Essence of Replicability (aka Repeatability)

- You are able
- to repeat
- your experiment
- on your computer.

**Minimal Requirements**

**Basic Documentation:**
- Recipe to obtain (numerical) results
- Recipe for post-processing of data
- Recipe for creating visualizations

**Optional Recommendations**

**Automation and Testing:**
- Machine-readable recipes
- For example (shell) scripts
- Sanity tests
Reproducibility of a CBEx means that it can be repeated by a different researcher in a different computer environment. This is an adaptation of the general concept of Reproducibility that is key in any science that relies on experiments, that is a subject in the theory of science, and which absence in a significant fraction of publications in many research areas has shaped the term Reproducibility crisis in recent years [Marcus '13]; cf. also [Collberg, Proebsting, & Warren '04] on Reproducibility in computer science.

(https://en.wikipedia.org/wiki/Replication_crisis collects > 100 references across the sciences.)
The Essence of Reproducibility

- Someone else is able
- to repeat
- your experiment
- on their computer.
The Essence of Reproducibility

- Someone else is able
- to repeat
- your experiment
- on their computer.

Minimal Requirements

Detailed Documentation:
- Environment description
- Versions of system and dependencies
- Building instructions (if applicable)
The Essence of Reproducibility

- Someone else is able
- to repeat
- your experiment
- on their computer.

Minimal Requirements

Detailed Documentation:
- Environment description
- Versions of system and dependencies
- Building instructions (if applicable)

Optional Recommendations

Availability:
- Location with long-term storage
- Storage is not bound to author
- Persistent identifier is provided
**Definition**

In the sphere of CBEx, **Reusability** refers to the possibility to reuse the software or parts thereof for different purposes, in different environments, and by researchers other than the original authors.

- In particular, Reusability enables the utilization of the test setup or parts of it for other experiments or related applications.

- Although theoretically, any bit of a software can be reused for different purposes, here, Reusability applies only for reproducible parts.
The Essence of Reusability

- Someone else is able
- to use your experiment
- on their computer.
- for something else.
The Essence of Reusability

- Someone else is able to use your experiment on their computer. for something else.

Minimal Requirements

Accessibility:

- Availability (Code, Howto)
- Remote access required
- Binaries available (if applicable)
The Essence of Reusability

- Someone else is able
- to use your experiment
- on their computer.
- for something else.

Minimal Requirements

Accessibility:

- Availability (Code, Howto)
- Remote access required
- Binaries available (if applicable)

Optional Recommendations

Modularity, Software Management and Licensing:

- Modular design
- Project management facilities
- License considerations
• Replicability
  Required: Basic Documentation
  Recommended: Automation & Testing

• Reproducibility
  Required: Extensive Documentation
  Recommended: Availability

• Reusability
  Required: Accessibility
  Recommended: Software Management, Modularity & Licensing
Replicability ← Verifies your findings

Sustainable software is:
- Findable
- Accessible
- Interoperable
- Reusable

Jens Saak, saak@mpi-magdeburg.mpg.de
The Road to Sustainability

- Replicability ← Verifies your findings
- Reproducibility ← Ensures it is science

Sustainable software is:
- Findable,
- Accessible,
- Interoperable,
- Reusable
The Road to Sustainability

- Replicability ← Verifies your findings
- Reproducibility ← Ensures it is science
- Reusability ← Enables scientific progress

Sustainable software is:
- Findable,
- Accessible,
- Interoperable,
- Reusable

Jens Saak, saak@mpi-magdeburg.mpg.de
The Road to Sustainability

- Replicability ← Verifies your findings
- Reproducibility ← Ensures it is science
- Reusability ← Enables scientific progress

Sustainable software is:

Findable, Accessible, Interoperable, Reusable
Proposed Development Practices

based on
Proposed Development Practices

- small project
  - paper code, thesis project code

- large project
  - groups in-house tool, community code, ...
Proposed Development Practices

- **small project** ← often single developer and user
  
  paper code, thesis project code

- **large project**
  
  groups in-house tool, community code, ...
Proposed Development Practices

- **small project** ← often single developer and user
  
  paper code, thesis project code

- **large project** ← separate developer and user groups

  groups in-house tool, community code, ...
Small Project

developer 1

copyright

developer 2

documentation

check list

source code

repository

hand over time
Small Project: Requirements

- Code availability
  (recoverable from central institute repository)

- Working example(s)
  (RUNME, easier handover, usable for testing)

- Code ownership
  (institution? supervisor? developer?)

- Execution environment
  (documentation of soft- and hardware for compilation and execution)

- Minimal documentation
  (README)
Small Project: Recommendations

- Public release
  (License? Find community repositories: https://re3data.org/)

- Version control
  (track changes, named revisions, BACKUP!)

- Basic code cleanup
  (obscure constants, dead code, hard-paths)

- Reproducible execution environment
  (virtual machine, container, step-by-step guide, . . .)

- Integration into larger project
  (e.g. in-house or community code / modularity? interfaces?)
Large Project

Copyright entity

Maintainer

Source code

Check list

Repository

Main documentation

Stable release

Hand over

Period

Merge

Developer

Issues

Reporter

Jens Saak, saak@mpi-magdeburg.mpg.de

Fighting the Reproducibility Crisis
Large Project: Requirements

- Software license

- Code ownership of contributions
  (re-licensing, availability of copyright holders, ...)

- Access to project resources
  (website, code repo, mailing list, support desk, ...)
  (developer hierarchy, responsibilities)

- Development in branches
  (stable master, management of branches, ...)

- Changelog
  (compressed history for smooth handover)
Large Project: Recommendations

- Code maintainability
  (Code reviews, automatic testing and deployment (CI))

- Code of Conduct
  (handover guidelines, new and leaving maintainers, . . . )

- Contribution Policy
  (who? how? required skills?)

- Citation Policy
  (Do developers/authors get the credits?)
Wrap-up!

As an author make your . . .

- . . . CBEx replicable, reproducible, reusable.
- . . . scientific software sustainable and FAIR.

Questions? Remarks? Suggestions?

Jens Saak, saak@mpi-magdeburg.mpg.de
As an author make your ... 

- ... CBEx replicable, reproducible, reusable.
- ... scientific software sustainable and FAIR.

As a reviewer/editor ask the authors to do so.
Wrap-up!

As an author make your . . .

- . . . CBEx replicable, reproducible, reusable.
- . . . scientific software sustainable and FAIR.

As a reviewer/editor ask the authors to do so.

Questions? Remarks? Suggestions?
Finding

"... Metadata and data should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of datasets and services, ..." persistent identifier, rich & clear metadata, searchable resource
FAIR principles

F indable
“...Metadata and data should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of datasets and services, ...” persistent identifier, rich & clear metadata, searchable resource

A ccessible
“Once the user finds the required data, she/he needs to know how can they be accessed, possibly including authentication and authorisation.”
open, free and universal protocol with authentication where necessary
FAIR principles

F indable

“... Metadata and data should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of datasets and services, ...” persistent identifier, rich & clear metadata, searchable resource

A ccessible

“Once the user finds the required data, she/he needs to know how can they be accessed, possibly including authentication and authorisation.”

open, free and universal protocol with authentication where necessary

I nteroperable

“The data usually need to be integrated with other data. In addition, the data need to interoperate with applications or workflows for analysis, storage, and processing.” (meta)data in common language and fair vocabulary with qualified cross-references
FAIR principles

F indable
“...Metadata and data should be easy to find for both humans and computers. Machine-readable metadata are essential for automatic discovery of datasets and services, ...” persistent identifier, rich & clear metadata, searchable resource

A ccessible
“Once the user finds the required data, she/he needs to know how can they be accessed, possibly including authentication and authorisation.”
open, free and universal protocol with authentication where necessary

I nteroperable
“The data usually need to be integrated with other data. In addition, the data need to interoperate with applications or workflows for analysis, storage, and processing.” (meta)data in common language and fair vocabulary with qualified cross-references

R eusable
“The ultimate goal of FAIR is to optimise the reuse of data. To achieve this, metadata and data should be well-described so that they can be replicated and/or combined in different settings.”
(meta)data in community standard representation follows clear and accessible license
Related Material

- Software deposit guidance for researchers [10]
  (The Software Sustainability Institute)

- Recommendations on the development, use and provision of Research Software [9]
  (Alliance of German Science Organizations)

- Criteria for Software Self-Assessment [6]
  (INRIA Evaluation Committee)

- Open Source Guides [5]
  (GitHub and friends)

- Code of Conduct
  (Your favorite research organization or funding agency)

- ...
Further Reading I


version 1.0.


Useful Minimal Information (MATLAB, Octave, Python, R, Julia):

- Runtime interpreter name and version
- Operating system name, version and architecture / word-width
- Processor name and exact identifier
- Required amount of random access memory
- BLAS / LAPACK library implementation name and version
Pitfalls:

- CPU time vs wall time
- Parallelization (implicit / explicit)
- Efficient memory access (NUMA)
- Overhead (actual compute-time)
- Statistics (i.e. means of repeated runs)
Numerical Results

...

Code Availability Section

The source code of the implementations used to compute the presented results can be obtained from:

https://my.stable.url

and is authored by: X. Y., A. B.

(if available use supplemental material!)
README  Read this to get started!
RUNME   Run this to get started!
CODE    Machine readable code meta-data
CITATION How to cite the software?
...
Standard Project Files

AUTHORS  Who wrote it
LICENSE  The license text
INSTALL  How to install
CHANGELOG  What changed
DEPENDENCIES  What are the dependencies
VERSION  The version number
TODO  Open problems
FAQ  Frequently Asked Questions

...