National Institutes of Health
Office of Data Science Strategy

Best Practices for Sharing Research Software

Frequently Asked Questions

Introduction

The National Institutes of Health (NIH) promotes broad dissemination of research products including NIH-funded and/or NIH-developed research software. These FAQs offer best practices for sharin research software and source code, developed under research grants in any stage of development, in a free and open format. Releasing the software source code in an "open" manner means that yo permit users to use, modify, and/or redistribute the code. The FAIR principles (findable, accessible, interoperable, and reusable) also provide a useful framework for better software management an sharing.

1. Why should I share software and code as "open source" software?

Openly sharing research software, as well as source and object code, provides transparency. It also serves the key objectives of rigor and reproducibility as part of responsible conduct of researc These objectives are consistent with existing NIH data sharing policies and guidance.

Sharing software and associated code allows for software sustainability as well as contributes to the advancement of science and fosters collaboration among researchers across institutions.

2. How do I make software source code "open"?

Making software and code "open" means sharing software and code in a way that allows researchers to use code, modify and redistribute it. This can be done by releasing software and code in a open-access and open-source format in an appropriate, unrestricted, publicly accessible repository with version control. Repositories that allow the provisioning of additional metadata and provi search tools and finding aids can enhance the FAIR-ness of software.

Software may be shared in multiple ways:

- as source code or the executable version
- as code libraries published to general or specific package distribution channels
- as workflows or containers
- as services and APIs

In many cases both the code and the packages or runtime may be shared via different repositories. Researchers should use funder specified/preferred repositories. In the absence of any guidan researchers are encouraged to select a repository that is appropriate for the software and code generated from the research project that has long-term sustainability via social coding repositories are open, revision-control, source-code management systems such as GitHub, GitLab, or Bitbucket.

Code may also be registered in a community repository to enhance discoverability. Specialized software packages may also be distributed via package management and library utilities such as C or Bioconductor. Containers and workflows may be shared through repositories such as Dockstore.

3. Why should I use a license when distributing code?

Open licenses facilitate and encourage the reuse of the code by clarifying and documenting the terms of how the software can be used, modified, and redistributed by others, and for what purp Open licenses also disclaim any liability for problems associated with the software after the code has been released to the public. For those reusing and building on the work of others, it is import review the license and usage requirements.

4. How do I choose a license under which to release software developed as part of an NIH award?

Follow guidance provided by your funder. If no specific guidance is provided, NIH suggests using one of the licenses approved by the Open Source Initiative (OSI). An OSI license makes using and contributing to research software easier because these licenses are well known and understood.

5. How can I make my software citable?

Providing a reference citation to the released software will allow others to easily cite your work. Minting persistent identifiers or easily accessible URLs in stable locations will ensure credit to you developer and make it easier for the research community at large to discover the software.

Consider strategies that allow your software to be citable such as Zenodo's ability to assign unique Digital Object Identifiers (DOIs) (one type of persistent identifiers) for source code released in GitHub or adding a citation.cff file to your GitHub repository. This may be especially important when specific software version references are required to reproduce a research result.

6. How should I acknowledge NIH as the funder?

When acknowledging NIH support, provide the same information as you would in a publication and refer to the NIH guidance on Communicating and Acknowledging Federal Funding. Be sure to include this information in the software documentation, the license agreement, and the repository website or the disk/drive from which people download the software. Make sure to identify all contributors to the research software.

7. Are there any restrictions I should consider in deciding whether to share the research software I develop?

NIH encourages inclusion of plans to share software developed under NIH grants as part of resource sharing plans. Release of research software, however may be restricted when:

- Patent protection for the software is under consideration.
- The software code contains or may be used to obtain personally identifiable information.
- The software is associated with a medical device.

NIH recommends consulting your institution's Technology Development and Transfer Office or Sponsored Projects Office as appropriate to determine whether you should restrict release of your research software.

8. Can research software I have developed be allowed for use in medical practice or clinical settings?

In most instances, software developed for research may not be permitted for clinical use or medical purposes. If the software is intended for use in a clinical or medical setting, the software may to undergo appropriate scientific and regulatory reviews by the U.S. Food and Drug Administration (FDA). NIH also recommends consulting your institution's Technology Development and Transfe Office to seek appropriate guidance for sharing such software.

9. Do I have to check software developed for security vulnerabilities prior to sharing it?

It is encouraged to check software for vulnerabilities prior to sharing it. The SANS institute has published a list of the top 25 software errors. These lists could help guide you in review of the softw source code. See http://www.sans.org/top25-software-errors/.

10. What metadata should be considered when sharing research software?

Providing rich metadata allows research software to be discoverable and reusable. Open metadata allows for exchanges between systems, and reusable metadata eliminates duplication of efforts Metadata to include with software vary by the use cases – for example keywords and descriptions allows for discoverability of software, credit for academic software requires citation metadata, a research replicability requires software versions – often one or more of these are required. Consider metadata that allow software and code to be linked to publications, data sources, funding sup and other digital objects related to research. NIH also suggests considering metadata recommendation provided by repositories specialized to manage and share software when developing and releasing software.

Metadata to consider include, but not limited to are:

# Title: [Name of software]

# Description: [Describe the purpose of the software]

# Persistent Identifier: [A unique persistent identified (PID) such as a digital object identifier (DOI) or accession number supports data discovery, reporting and assessment.]

# Software Language and Version/Standard: [For example, C++ ISO/IEC 14882:2020]

# Author(s): [Names of software developers and contributors]

# Grant Number: [In this format R01GM987654]

# Publications: [Persistent identifiers and citations for publications by your team related to this software or code.]

11. To what extent should I include documentation for the software?

Documenting your code with sufficient detail allows other programmers the opportunity to update, extend, or execute your software application. Documenting the design and purpose of your co provides others with a better understanding of your code. Providing compute platform-specific details for how your code can be installed and used would allow other programmers to test and de the code appropriately for reproducible usage.

Better Scientific Software is one of many information sources on that provides guidance on code documentation practices.

12. Does NIH have any requirements or benchmarks for research software quality before releasing it?

There are no NIH-wide standard requirements related to quality of research software. However, NIH encourages researchers to adopt best practices of research software engineering while develo software funded by NIH with the aim of enhancing the sustainability of the software. Consider checklists such as those developed by:

- The Linux Foundation's Core Infrastructure Initiative
- The Software Sustainability Institute

The US Research Software Sustainability Institute is one of many information sources for software sustainability practices.

Conclusion

Several resources are available for developing and sharing research grade software. Working groups such as Force 11, WSSPE, The Society for Research Software Engineering, RDA and projects such a CodeMeta and Metadata2020 all provide guidance, tools, and access to practicing communities. NIH recommends considering these when developing and releasing software.

For specific questions on sharing research software developed under your NIH award, contact your assigned Program Officer or your institution's Sponsored Projects or Technology Development and Transfer Offices.

This page last reviewed on April 3, 2023

NIH...Turning Discovery Into Health®
National Institutes of Health, 9000 Rockville Pike, Bethesda, Maryland 20892
U.S. Department of Health and Human Services