5 Tips to Citing Your Research Software and Improving Discovery

Chris Erdmann, AGU Assistant Director, Data Stewardship

cerdmann@agu.org   @libcce

https://orcid.org/0000-0003-2554-180X

Shelley Stall, AGU Sr. Director, Data Leadership

sstall@agu.org   @ShelleyStall

https://orcid.org/0000-0003-2926-8353

Overview - Software Citation Tips

1. Write a Software Management Plan
2. Choose a Repository
3. Prepare Your Availability Statement
4. Include Software in the References
5. Add Instructions to Your Development Platform on How to Cite Your Software

What are some of the software challenges that researchers are facing?

Citing the Software

Crediting the Software

Lisa DeBruine 🏳️‍🌈
@LisaDeBruine

Replying to @andrewang91

I have the opportunity to suggest things to the APA editors in my role in the APA Open Science committee. This is one of the issues I think is important. It's often ECRs creating research software and the lack of credit discourages investment in this.

2:14 PM · Apr 5, 2021 · Twitter Web App

Searching for Software

> **Konrad Förstner** 💻 🧬
> @konradfoerstner
>
> One of the core conclusions of the "finding and publishing research software" session is that there is a painful lack of search engines for research software.
> #oscibar #OpenScience #ResearchSoftware
>
> 8:02 AM · Mar 12, 2018 · Twitter Web Client

The software, code, workflow, model that is integral to your research.

1. Write a Software Management Plan

"A software management plan can help to formalise a set of structures and goals that ensure your software is accessible and reusable in the short, medium and long term."

The Software Sustainability Institute. Software Management Plan. Available online: http://www.software.ac.uk/software-management-plans.

Choose an open source license

An open source license protects contributors and users. Businesses and savvy developers won't touch a project without this protection.

{ Which of the following best describes your situation? }

I need to work in a community.

Use the **license preferred by the community** you're contributing to or depending on. Your project will fit right in.

If you have a dependency that doesn't have a license, ask its maintainers to **add a license**.

I want it simple and permissive.

The **MIT License** is short and to the point. It lets people do almost anything they want with your project, like making and distributing closed source versions.

Babel, **.NET Core**, and **Rails** use the MIT License.

I care about sharing improvements.

The **GNU GPLv3** also lets people do almost anything they want with your project, *except* distributing closed source versions.

Ansible, **Bash**, and **GIMP** use the GNU GPLv3.

{ What if none of these work for me? }

My project isn't software.

There are licenses for that.

I want more choices.

More licenses are available.

I don't want to choose a license.

Here's what happens if you don't.

AGU
ADVANCING EARTH
AND SPACE SCIENCE

2. Choose a Repository

Collaborative software development platforms (e.g. GitHub) are helpful for developing your software in the near term but **are not long-term solutions**.

Choose a repository as a long-term solution:

- **Institutional repository supporting software preservation** that provides registration services for a digital object identifier/persistent identifier (e.g. your institution's repository).

- **Software domain repository** (e.g. Computational Infrastructure for Geodynamics (CIG), Hydroshare),

- **General repository supporting software preservation** that provides registration services for a digital object identifier (e.g. Zenodo).

GitHub is NOT a Preservation Solution!
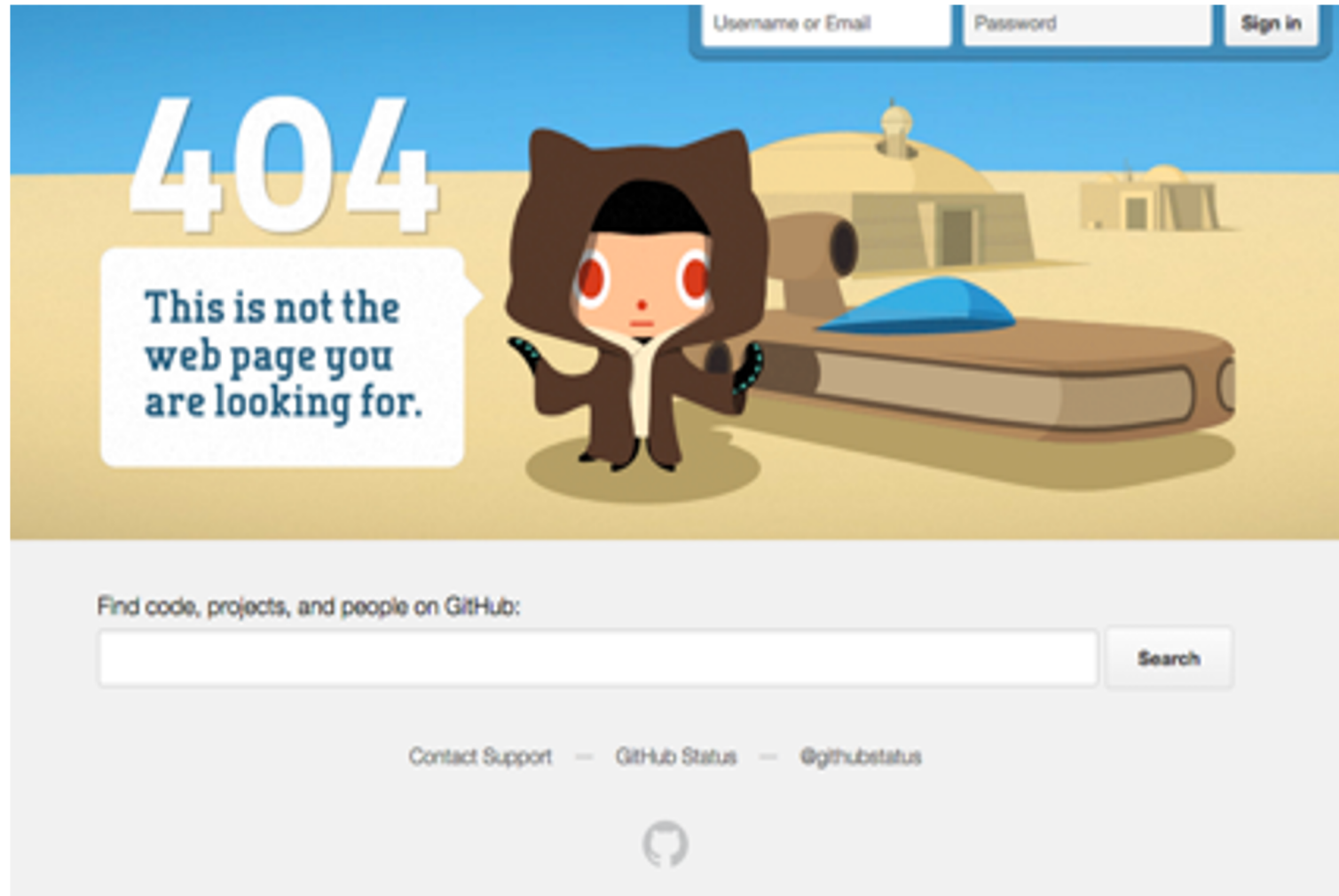
Making Your Code Citable

🕐 10 minute read

Digital Object Identifiers (DOI) are the backbone of the academic reference and metrics system. If you're a researcher writing software, this guide will show you how to make the work you share on GitHub citable by archiving one of your GitHub repositories and assigning a DOI with the data archiving tool Zenodo.

ProTip: This tutorial is aimed at researchers who want to cite GitHub repositories in academic literature. Provided you've already set up a GitHub repository, this tutorial can be completed without installing any special software. If you haven't yet created a project on GitHub, start first by uploading your work to a repository.

Intro

Choosing Your Repo

Login to Zenodo

Check Repo Settings

Create a New Release

Minting a DOI

Finishing up

Reminder

Check if the author name(s), title, version are specific.

Zenodo, a commonly used general repository for software, features a sandbox where you can test how your software will be deposited and described.
https://sandbox.zenodo.org/

3a. Prepare Your Availability Statement

You should include software that has a **significant impact** on the research outcome presented in your work, or on the way the research has been conducted. If the research you are presenting is **not reproducible without** a piece of software, then you should include the software.

In the Availability Statement section of your paper (also called Acknowledgements or Open Research), include:

1. Very brief description of what the software is used for
2. Repository DOI/PID for your preserved software (e.g. Zenodo)
3. Version of the software used and preserved
4. (Open source) license and other access conditions
5. Development platform link (e.g. GitHub repository)

In the methodology section of your paper:

1. Describe how your software works as it pertains to your research

AGU
ADVANCING EARTH
AND SPACE SCIENCE

3b. Prepare Your Availability Statement (Template)

The statement template would be:
[Version #] of the data cleaning and analysis software used for this research is preserved at [DOI/PID link], available via [license type, access conditions] and developed openly at [software development platform link].

4. Include Software in the References (Examples)

Yong Zhou. (2020, May 23). yzhou1989/FAPPWF master (Version master). Zenodo. http://doi.org/10.5281/zenodo.3840905

Lizzy Trower. (2020, January 8). lizzytrower/giantooids: Giant ooid equilibrium size model, 2nd update (Version 1.2). Zenodo. http://doi.org/10.5281/zenodo.3601507

Shaoqian Hu. (2019, December 25). Direct surface wave radial anisotropy tomography package (Version 1.0). Zenodo. http://doi.org/10.5281/zenodo.3592528

5. Add Instructions to Your Development Platform on How to Cite Your Software

Citation File Format (CFF)   ⦿ About   ⟲ Events   📄 Documentation

🕐 3 minute read

What is a `CITATION.cff` file?

`CITATION.cff` files are plain text files with human- and machine-readable citation information for software. Code developers can include them in their repositories to let others know how to correctly cite their software.

A `CITATION.cff` file looks like this:

```
cff-version: 1.1.0
message: If you use this software, please cite it as below.
authors:
  - family-names: Druskat
    given-names: Stephan
    orcid: https://orcid.org/0000-0003-4925-7248
title: "My Research Software"
version: 2.0.4
doi: 10.5281/zenodo.1234
date-released: 2017-12-18
```

📄 ON THIS PAGE

WHAT IS A `CITATION.CFF` FILE?

WHY YOU SHOULD ADD A `CITATION.CFF` FILE TO YOUR REPOSITORY!

CREATE A `CITATION.CFF` FILE NOW!

TOOLS FOR WORKING WITH `CITATION.CFF` FILES

EVENTS

HAVE AN IDEA? FOUND A PROBLEM?

https://citation-file-format.github.io/#/what-is-a-citation-cff-file

AGU
ADVANCING EARTH
AND SPACE SCIENCE

Benefits of Citing Your Software

- A **persistent copy** of your software that supports the reproducibility of your research
  - Cannot be confused with other versions/copies
- Increased **discoverability**, awareness of your work
- Trackable citation method that gives you (and your co-authors) **credit** as the creator

Thank you.

Chris Erdmann, AGU Assistant Director, Data Stewardship

cerdmann@agu.org @libcce

https://orcid.org/0000-0003-2554-180X

Shelley Stall, AGU Senior Director, Data Leadership

sstall@agu.org   @ShelleyStall

https://orcid.org/0000-0003-2926-8353