

Archiving and Citing Software

An Introduction for Novices

Daina Bouquin

Head Librarian

Harvard-Smithsonian Center for Astrophysics

daina.bouquin@cfa.harvard.edu

future scientific legacy relies on code

Your work will be the foundation on which the next generation must build an improved understanding of how the Universe works

**if we don't take care of code we create
holes in the scientific record**

An example:

Machine Learning

Software is inseparable from "the data"

ML frameworks trade off exact numeric determinism for
performance and often require remote computing resources

Even if you copy development steps there will be
tiny differences in the end results

This is the future (emergent reality) of scientific research

<https://petewarden.com/2018/03/19/the-machine-learning-reproducibility-crisis/>

Sharing research is challenging in new ways.

This is not a "problem"

We need to acknowledge that changes are needed though

FAIR Code

is good for the future of science

The screenshot shows a scientific article page from the journal *Scientific Data*. The title of the article is "The FAIR Guiding Principles for scientific data management and stewardship". The authors listed are Mark D. Wilkinson, Michel Dumontier, [...] Barend Mons. The article was published on 15 March 2016. The page includes metrics such as Altmetric: 1144 and Citations: 311, and links for Comment, OPEN, and Download Citation.



**Findable
Accessible
Interoperable
Reusable**

backup ≠ archive

Backup

- backups are just copies
- can be used to restore an original
- used for operational recoveries (e.g. recover an overwritten file)
- focus is on speed of recovery and integrity

Archive

- archives store a version of a file that's no longer changing, or shouldn't be changing
- searchability is more critical in archives
- importance is placed on the ability to scale data integrity and retention over long periods of time
- collection of historical records kept for long-term retention/used for future reference

★ You should not pit backup
against archiving ★
(use them together)

Archives have persistent identifiers

DOI

**"the backbone of the
academic reference and
metrics system"**

<https://guides.github.com/activities/citable-code/>

[Citation Needed]

- need for a complete record of the research process
- need to enable software discoverability
- importance of research reproducibility
- **give credit to academic researchers of all levels for the software that they develop**

Native data and software citation are vitally important

```
\software{Astropy \citep{http://dx.doi.org/10.1051/0004-6361/201322068},  
        Matplotlib \citep{http://dx.doi.org/10.1109/MCSE.2007.55}}
```

Citation File Format: CITATION.cff

If you want to make your software easily citable, you can put a file called `CITATION.cff` in the root of your repository. This file should provide at least the minimally necessary metadata to cite your software. For example:

```
cff-version: 1.0.3
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 Tool
    version: 1.0.4
    ★
    doi: 10.5281/zenodo.1234
    date-released: 2017-12-18
```

</>

Solution for most common software citation use cases:

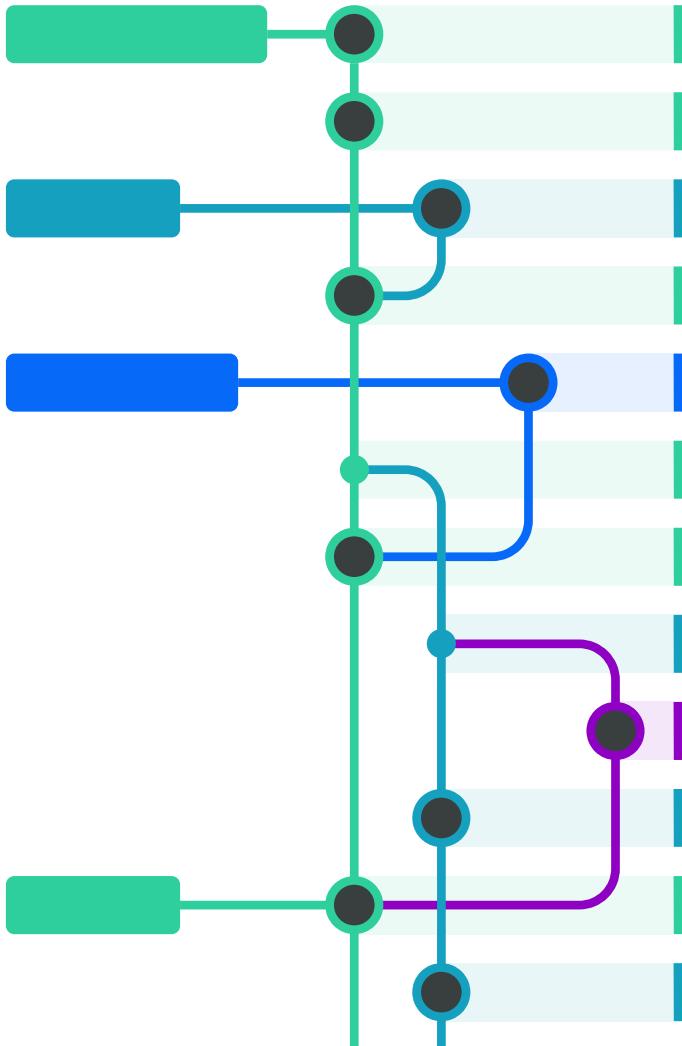
1. *Use software for a paper*
2. *Use software in/with new software*
3. *Store software entry*

Authorship

The astronomy community doesn't agree on how much someone should contribute to a code before that person is considered an author.

- astronomers will cite a paper rather than natively citing code whether or not the code they want to cite is the same version as the code discussed in the paper
 - **May contribute to software paper authors receiving disproportionate credit and current contributors not receiving any**
- Acknowledgement for contributions that might not fit a definition of "authorship"
 - **giving all contributors equal credit as authors may serve to dilute the perceived importance of authoring software**

Versioning



Complicates authorship issues and issues pertaining to dependencies and documentation (metadata)

How do we deal with multiple forks?

How should citations be calculated across different types of digital objects and versions of those objects?

Licenses

Putting things online doesn't make them "open"

Open

(very important)

- Allow software to be freely used, modified, and shared
- **You define how your code can be used**
- Many options

- Apache License 2.0
- BSD 3-Clause "New" or "Revised" license
- BSD 2-Clause "Simplified" or "FreeBSD" license
- GNU General Public License (GPL)
- GNU Library or "Lesser" General Public License (LGPL)
- MIT license
- Mozilla Public License 2.0
- Common Development and Distribution License
- Eclipse Public License

Proprietary

(e.g. IDL, IRAF, MATLAB, etc)

- **Restrictive licenses**
- older versions have no ongoing support
- Still determining "fair use" of proprietary research software

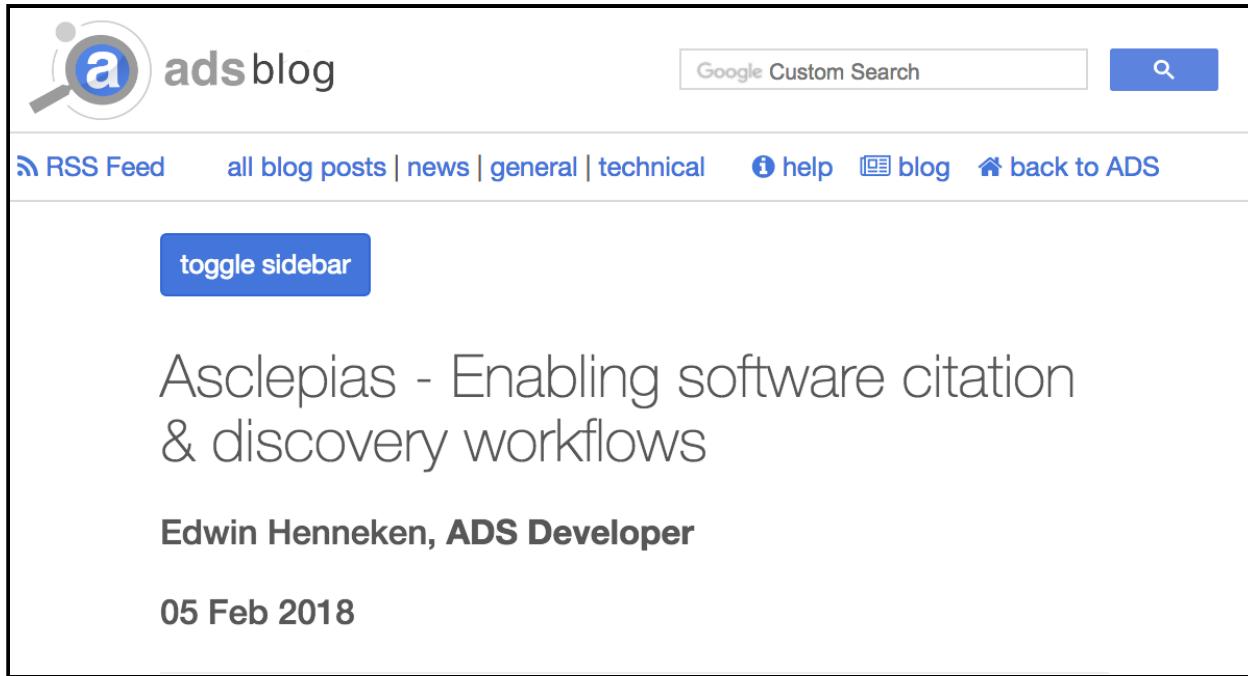
★ **Choose a license that fits your goals and project**

Considerations for how to share code

- Papers
 - Publishers have different citation/publishing policies (e.g. [AAS](#))
- GitHub/Zenodo integration
 - Native software citation
 - **Versioned** DOIs
- Journal of Open Source Software
 - Peer review of code
 - Currently establishing partnership with AAS Publishing
- ASCL
 - Index but no persistent IDs
 - Does not enable native software citation

Ongoing Projects

Asclepias



The screenshot shows a blog post titled "Asclepias - Enabling software citation & discovery workflows" by Edwin Henneken, ADS Developer, posted on 05 Feb 2018. The page includes a sidebar with links to RSS Feed, blog posts, news, general, technical, help, and back to ADS. It also features a "toggle sidebar" button and a Google Custom Search bar.

ads blog

Google Custom Search

RSS Feed all blog posts | news | general | technical help blog back to ADS

toggle sidebar

Asclepias - Enabling software citation & discovery workflows

Edwin Henneken, ADS Developer

05 Feb 2018

Gus Muench

Alberto Accomazzi, Sergi Blanco-Cuaresma, Edwin Henneken

Lars Holm Nielsen, Krzysztof Nowak, Alexander Ioannidis

Thomas Robitaille



<https://adsabs.github.io/blog/asclepias>



The CodeMeta Project

Metadata for Software

- **credit** for academic software
 - citation metadata
- **replicate** some analysis
 - versions and dependencies
- discover software you don't already know
 - **keywords** and descriptions

Zenodo.org is a data archive based at CERN which is popularly used to archive and provide DOIs to academic software from GitHub, as described in the official GitHub guide to [Making your code citable](#).

Property	Zenodo
codeRepository	relatedLink
applicationCategory	communities
author	creators
datePublished	date_published
funder	contributors.Funder
keywords	keywords
license	license
description	description/notes
identifier	id
name	title
affiliation	affiliation
identifier	ORCID
name	name



Software Heritage

THE GREAT LIBRARY OF SOURCE CODE



An essential infrastructure for science

[Home](#) / [Mission](#) / An essential infrastructure for science

A large part of the technical and scientific **knowledge** that is being developed today **resides in software**. The preservation of this universal body of knowledge has become as essential as preserving research articles and data sets.

As an extremely valuable **service to the research community**, we will search for, collect, organize, preserve and make easily available all the software.

US Research Software Sustainability Institute



URSSI

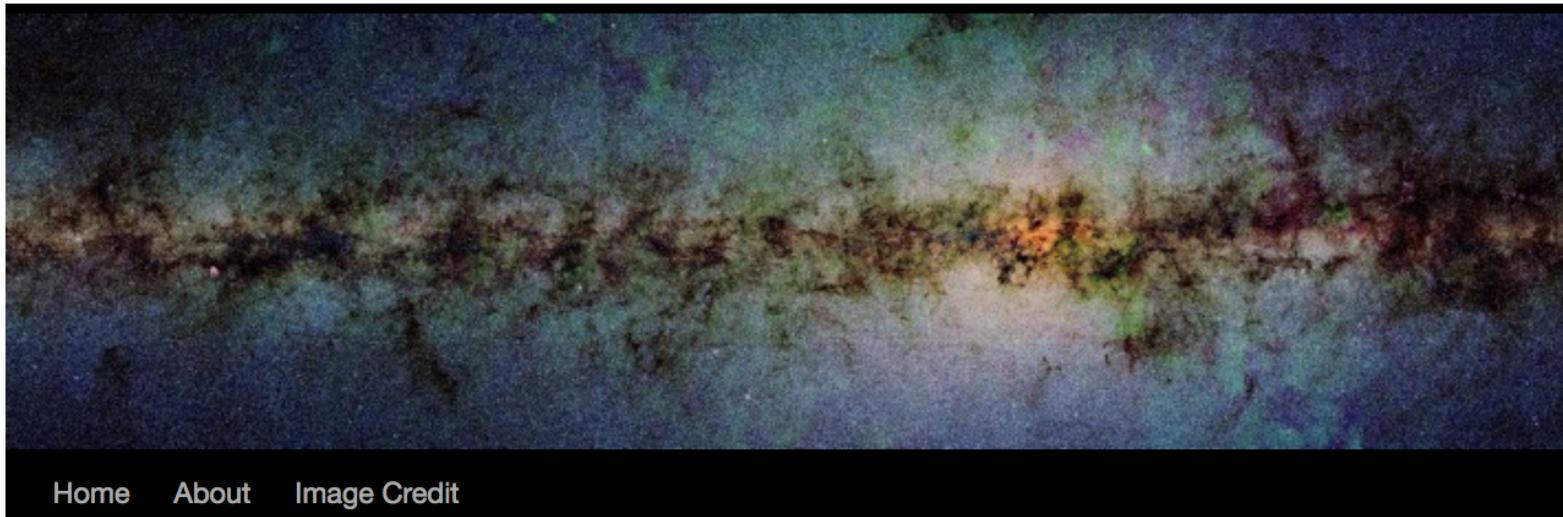
Developing a pathway to
research software
sustainability

<http://urssi.us/>

Let's do some stuff right now

- License your code openly
- Create persistent ID for yourself (ORCID)
- Make a DOI for your code using Zenodo
 - Create a Zenodo account if you don't already have one
- Create a CITATION.cff file
 - Software examples

Astronomy Computing Today



[Home](#) [About](#) [Image Credit](#)

← The Virtual Observatory Is Very Much Real!

Software Citation Implementation in Astronomy

Posted on [March 5, 2018](#)

This week's post is a repost of a summary of a meeting I attended, held at the 231st AAS meeting in National Harbor, MD, January 2018. The original summary, prepared by Daina Bouquin and Arfon Smith, is at https://github.com/CfA-Library/Cite_Astro_Software/blob/master/whitepaper.md, and is reproduced here with permission.

<https://astrocompute.wordpress.com/2018/03/05/software-citation-implementation-in-astronomy/>