

Acknowledging Scientific Software

Daina Bouquin

Harvard-Smithsonian Center for Astrophysics

daina.bouquin@cfa.harvard.edu

[@dainabouquin](https://twitter.com/dainabouquin)

**because you want
credit for your work**

**because you care about
your work**

**I don't want you to
have to think about
citation.**

**I want you to have a
scientific legacy.**

Your work will be the foundation on which future generations build new knowledge.

Your work will be their heritage.

Code is speech.

Daina,

I took the liberty of looking you up in the faculty directory. Thank you for looking into the code for the [REDACTED] computer program. The PI for the study was [REDACTED].

If you do find the code, I can arrange for it to be loaned to one of my colleagues at SAO.

Thank you, and if you need any more information from me, please let me know.

Hello,

At the end of the attached paper, there is a link to a computer code

[REDACTED]

The link does not work any more. Is it still possible to get the code?

Software Citation Principles

<https://doi.org/10.7717/peerj-cs.86>

- Software should be considered **important**
- Normative, legal **attribution and credit** should be given to software authors
- Software should be identified using machine actionable, **globally unique**, interoperable identifiers
- Identifiers should point to **specific versions** of software
- Software metadata should be **persistent**
- Citation should enable **access to the software itself**

How do we actually do this?

FORCE11: Software Citation **Implementation** Working Group

- publish software
- document software using structured metadata
 - Citation File Format
 - CodeMeta
 - Software Heritage

Publishing Software

published = permanently archiving it and creating a resolvable identifier

(e.g. by [Zenodo](#), figshare, institutional archival repositories)

unpublished = the software is made available by a hosting organization that does not commit to long term preservation (e.g. GitHub, personal website)

Best thing to cite



"published software"

means

"archived software"



oh god

Citation File Format

human- and machine-readable file format that provides citation metadata for software.

Example

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
```

CodeMeta

more than citation metadata

Creating A CodeMeta Instance File

A CodeMeta instance file describes the metadata associated with a software object using JSON's linked data (JSON-LD) notation. A codemeta file can contain any of the properties described on the [CodeMeta terms page](#).

When creating a CodeMeta document, note that they contain JSON name ("property" in linked-data), value pairs where the values can be simple values, arrays or JSON objects. A simple value is a number, string, or one the literal values *false*, *null*, *true*, for example:

```
"name" : "R Interface to the DataONE REST API"
```

A JSON array is surrounded by the characters [and], and can contain multiple values:

```
"keywords": [ "data sharing", "data repository", "DataONE" ]
```

Some properties, such as **author**, can refer to other JSON objects surrounded by curly braces and can contain other JSON values or objects, for example:

```
"author": {  
  "@id": "http://orcid.org/0000-0003-0077-4738",  
  "@type": "Person",  
  "email": "slaughter@nceas.ucsb.edu",  
  "givenName": "Peter",  
  "familyName": "Slaughter"  
}
```

The JSON-LD "@type" keyword associates a JSON value or object with a well known type, for example, the statement "**@type**:"**Person**" associates the **author** object with <http://schema.org/Person>.

It is good practice to always provide the **@type** for any property which specifies a node (JSON object). The [terms page](#) indicates these node types.

The "author" JSON object illustrates the use of the JSON-LD keyword "@id", which is used to associate an IRI with the JSON object. Any such node object can be assigned an **@id**, and we may use the **@id** to refer to this same object (the person, Peter), elsewhere in the document; e.g. we can indicate the same individual is also the **maintainer** by adding:

```
"maintainer": "http://orcid.org/0000-0003-0077-4738"
```



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.

**You need to care to do this
properly.**

It's not easy yet.

Issues from Astronomy Land

Authors are specifically requesting people cite something other than the code even when a software DOI for the code exists.

Acknowledging or Citing Astropy

In Publications

If you use Astropy for work/research presented in a publication (whether directly, or as a dependency to another package), we ask that you please cite the Astropy papers.

- [Astropy Paper II \(ADS - BibTeX\)](#)
- [Astropy Paper I \(ADS - BibTeX\)](#)

[Copy BibTeX to clipboard](#)

We provide the following LaTeX/BibTeX acknowledgment if there is no specific place to cite the papers:

```
This research made use of Astropy,\footnote{http://www.astropy.org} a community-developed core Python package for Astronomy \citep{astropy:2013, astropy:2018}.
```

**These things can be cited in addition to the code,
rather than as stand-ins for the code**

Software DOI doesn't guarantee a native software citation

```
<resourceType resourceTypeGeneral="Software"/>
```

zenodo Search Upload Communities Log in Sign up

April 3, 2018

Software **Open Access**

astropy/astropy-v2.0-paper: final draft

Adrian Price-Whelan; Steve Crawford; Brigitta Sipocz; Miguel de Val-Borro; Hans Moritz Günther; Adam Ginsburg; P. L. Lim; Thomas Robitaille; Erik Tollerud; Simon Conseil; Paul Sladen; Pauline Barmby; Jake Vanderplas; Igbouma; Yannick Copin; Derek Homeier; Nadia Dencheva; Hugo Buddelmeijer; Tim Jenness; Ole Streicher; mdmueller; David Shupe; David Pérez-Suárez; Benjamin Alan Weaver; Kelle Cruz; Jörg Dietrich; Juan Luis Cano Rodríguez; Gabor Kovacs; Demitri Muna; Aleksandr Bakanov

This is the final submitted draft of the paper

Preview

astropy-v2.0-paper-final_draft.zip

- astropy-astropy-v2.0-paper-dc3b6fe
 - .gitignore 66 Bytes
 - .travis.yml 343 Bytes
 - Makefile 935 Bytes
 - README.md 3.3 kB
 - aasjournal.bst 35.7 kB
 - aastex62.cls 203.7 kB
 - affiliated.py 973 Bytes
 - author.tex 16.3 kB
 - bib_mapping.json 932 Bytes
 - bibliography.bib 51.2 kB
 - build-paper-travis.sh 419 Bytes
 - figures
 - bayesian_blocks_hist.pdf 16.6 kB
 - commits.pdf 60.2 kB
 - commits_figure.py 4.2 kB
 - convolution_example.pdf 38.6 kB
 - convolution_figure.py 3.0 kB
 - coordinates-benchmark.pdf 18.3 kB

29 views 0 downloads

See more details...

Available in

GitHub








Publication date: April 3, 2018

DOI: DOI 10.5281/zenodo.1211397

Related identifiers: Supplement to: https://github.com/astropy/astropy-v2.0-paper/tree/final_draft

License (for files): [Other \(Open\)](#)

PlasmaPy: an open source community-developed Python package for plasma physics

PlasmaPy Community;  Murphy, Nicholas A.;  Leonard, Andrew J.;  Stańczak, Dominik;  Kozłowski, Pawel M.; Langendorf, Samuel J.; Haggerty, Colby C.; Beckers, Jasper P.;  Mumford, Stuart J.;  Parashar, Tulasi N.;  Huang, Yi-Min

BibTeX Export

```
@misc{plasmapy_community_2018_1238132,
  author      = {PlasmaPy Community and
                 Murphy, Nicholas A. and
                 Leonard, Andrew J. and
                 Stańczak, Dominik and
                 Kozłowski, Pawel M. and
                 Langendorf, Samuel J. and
                 Haggerty, Colby C. and
                 Beckers, Jasper P. and
                 Mumford, Stuart J. and
                 Parashar, Tulasi N. and
                 Huang, Yi-Min},
  title       = {{PlasmaPy: an open source community-developed
                 Python package for plasma physics}},
  month       = apr,
  year        = 2018,
  note        = {{This work was partially supported by the U.S.
                 Department of Energy.}},
  doi         = {10.5281/zenodo.1238132},
  url         = {https://doi.org/10.5281/zenodo.1238132}
}
```

Slide Deck



complicated / conflicting author instructions

ASCL Code Record

[[ascl:1109.015](#)] [WCSTools: Image Astrometry Toolkit](#)

Mink, Jessica

WCSTools is a package of programs and a library of utility subroutines for setting and using the world coordinate systems (WCS) in the headers of the most common astronomical image formats, FITS and IRAF .imh, to relate image pixels to sky coordinates. In addition to dealing with image WCS information, WCSTools has extensive catalog search, image header manipulation, and coordinate and time conversion tasks. This software is all written in very portable C, so it should compile and run on any computer with a C compiler.

Code site: <http://tdc-www.harvard.edu/software/wcstools/>

Appears in: <http://adsabs.harvard.edu/abs/1999ASPC..172..498M>

Bibcode: [2011ascl.soft09015M](#)

Preferred citation method:

Depends on usage; see <http://tdc-www.harvard.edu/software/wcstools/publications/> for information



References

If the WCSTools package has proven useful in your work, please reference at least one of the following papers. If you want to find out which is most applicable to your work, please contact the author.

Entire Package

As of 2018, the latest summary paper for WCSTools is being presented at ADASS XXVIII in College Park, MD and is entitled:

"Exploring Space, Time, and Data with WCSTools " and will be published in 2019 in Astronomical Data Analysis Software and Systems XXVIII

[[abstract](#)] [[PDF poster](#)] [[HTML of the mostly-text poster](#)]

In 2005, WCSTools was upgraded to the then-latest version of WCSLIB and a paper was presented at ADASS:

"WCSTools 4.0: Building Astrometry and Catalogs into Pipelines," Douglas J. Mink, 2005 in Astronomical Data Analysis Software and Systems XV,

[[abstract](#)] [[PDF poster](#)] [[HTML presentation](#)] [[PDF presentation](#)] [[PDF paper](#)] [[HTML paper](#)]

The [2001 ADASS paper](#) gives the current status of the package, with examples of many of its capabilities:

"WCSTools 3.0: More Tools for Image Astrometry and Catalog Searching", Douglas J. Mink, 2002, in Astronomical Data Analysis Software and Systems XI,

[[abstract](#)] [[full text](#)] [[PDF poster](#)]

The [1998 ADASS paper](#) is the first published description of the tools in the WCSTools package, including SAOimage when used for WCS work:

"WCSTools: An Image Astrometry Toolkit", Douglas J. Mink (1998), in Astronomical Data Analysis Software and Systems VIII, A.S.P. Conference Series, Vol. 100,

[[ADS abstract](#)]

Fitting World Coordinate Systems to Images

The [paper presented at ADASS in 1996](#) is the best published description of the IMWCS world coordinate system fitting program:

"WCSTools: Putting Image World Coordinate Systems to Use", Douglas J. Mink, 1997, in Astronomical Data Analysis Software and Systems VI, A.S.P. Conference Series, Vol. 100,

[[ADS Abstract](#)] [[local abstract](#)] [[proceedings full text](#)]

Using Source Catalogs

Our [2003 ADASS paper](#) tests the accuracy of the various catalogs one is likely to use with [imwcs](#), showing how well it works in a pipeline mode on 1728 images:

"A Comparison of Large All-Sky Catalogs" Douglas J. Mink, Warren R. Brown, and Michael J. Kurtz in Astronomical Data Analysis Software and Systems XV, p. 141.

[[abstract](#)] [[PDF full text](#)] [[Web presentation](#)]

Our [2003 AAS/DDA paper](#) has more of an astrometric slant comparing accuracies of the most commonly used catalogs against actual star positions in 1728 images:

"A Comparison of Large All-Sky Catalogs", Douglas J. Mink, Warren Brown, and Michael J. Kurtz, 2003.

[[abstract](#)] [[Web presentation](#)]

The [2002 ADASS paper](#) displays some of the uses of [scat](#) as part of the Virtual Observatory:

"Federating Catalogs and Interfacing Them with Archives: A VO Prototype" Douglas J. Mink and Michael J. Kurtz, 2002, in Astronomical Data Analysis Software and Systems VIII, A.S.P. Conference Series, Vol. 100, p. 169.

[[abstract](#)] [[PDF full text](#)] [[Web presentation](#)] [[Open/StarOffice presentation](#)]

software authors

- You control your metadata.
- You are your own cataloger.

article authors

- You need to cite software correctly.
- No one else will catch mistakes.
- You are your own copy editor.

Publishers

- You need policies that can be enforced.
- You need to provide examples.

**Things you can do
right now**

Software Authors

- Mint a software DOI
- Create a CFF file
- License your data and code explicitly
- **Update and check your metadata**
 - Check it again
- Link documentation to the source code directly
- Ensure your preferred citations/any instructions about attribution **enable native software citation**
- If you have many versions of software, decide who the authors are for each version*

* get a freaking ORCID

Article Authors

- Look for preferred citations
 - **Look everywhere**
- If you cannot find a preferred citation, follow the F11 guidance and make sure you're doing your best at native software citation
- Consider the version that you are citing
 - Who are you trying to give credit?
- Follow publisher policies, if there isn't one follow F11 principles
- **Put software citations in the references section**
- **Cite your own code in a software paper**
 - tells others how you want it cited

Publishers

- Make a software citation policy
- Provide **examples**
 - What to do
 - What not to do
- Make expectations clear as to how much editorial review will be dedicated to checking software citations
 - **Everyone assumes you will fix it**
- If you accept software papers recommend authors create metadata files and mint a DOI

Thank You