CiteAs.org: Discovering and Improving software requests for citation

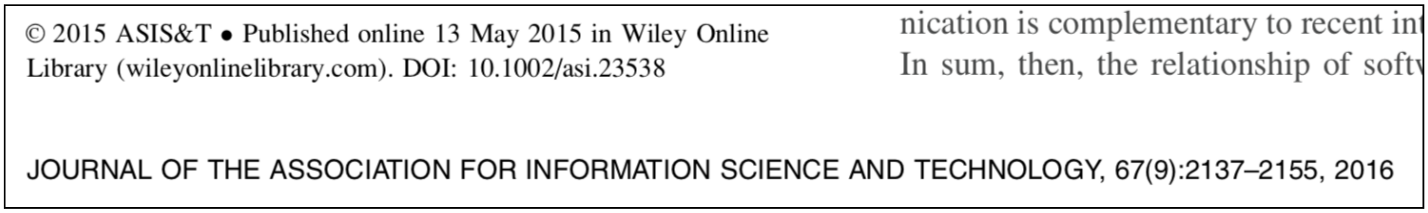
James Howison (University of Texas at Austin), Heather Piwowar and Jason Priem (Impact Story)

October 1, 2018

CiteAs.org (https://citeas.org) links between pieces of software and their requested citations. It enables moving from the name of a piece of software, its webpage URL, or a DOI, directly to the machine-readable metadata (e.g., BibTex, Zotero auto-import) for the citation the author of the software package wants you to use. CiteAs.org is funded by the Digital Science program at the Sloan Foundation (Grant Number 8028), and conceived and developed by Heather Piwowar and Jason Priem at ImpactStory (https://impactstory.org), together with James Howison (http://james.howison.name) from the Information School at the University of Texas at Austin.

Great software work → Clear requests for citation → More visibility in publications → More credit → Better Software → Better Research

Getting credit for useful software in science is important, yet it is rarely clear how the authors of a piece of software would like to be cited. Howison and Bullard, 2013 (http://doi.org/10.1002/asi.23538) found that less than half of the times in which publications actually mentioned software were formal, traceable, citations. At least part of the reason is that software very rarely contains citation information, and even when software does, that information is certainly not as obvious as it is in a paper. With a paper, if you have the paper you have the metadata right there in front of you:



© 2015 ASIS&T • Published online 13 May 2015 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/asi.23538

nication is complementary to recent int
In sum, then, the relationship of softv

JOURNAL OF THE ASSOCIATION FOR INFORMATION SCIENCE AND TECHNOLOGY, 67(9):2137–2155, 2016

But if you have a piece of software, you very likely don't have the metadata needed to cite it. You know the name of the code, you might know the homepage of the project that creates it, but you don't know how to cite it. In part this is because with software there is no standard place to "write" the information, but the problem is compounded because authors sometimes want their users to cite something other than the piece of software directly. Examples include citing a paper that introduces the software (or demonstrated its potential), a published software manual or book, a "software paper" created specifically as a citation target, or a benchmarking paper.

Great work is being done to guide best practices (including the FORCE11 Working Group on Software Citation (https://www.force11.org/group/software-citation-working-group)) which recommends always including a direct citation to the software itself, including version numbers—something key for reproducibility—in addition to papers. We don't disagree, but we think it's important to let the authors decide how their contribution should be acknowledged and to link users with those requests.

One approach to making this link is to create a new standard format and location to make clear requests, such as including a free text request in a CITATION file (https://www.software.ac.uk/blog/2016-10-06-encouraging-citation-software-introducing-citation-files) or a machine readable set of requests in a CodeMeta file (https://codemeta.github.io/) or CITATION.cff file (https://github.com/citation-file-format/citation-file-format) file. These have the advantage of being easy to locate and read, but the disadvantage of requiring everyone to adopt a new practice before this approach can work.

We know that people already make requests for citation in a whole range of places, including requests on project web-pages that provide `bibtex` or DOIs, metadata associated with DOIs or repositories (such as Github and Gitlab), and in language specific formats (such as R's `citation()` method, which reads from a `DESCRIPTION` file).

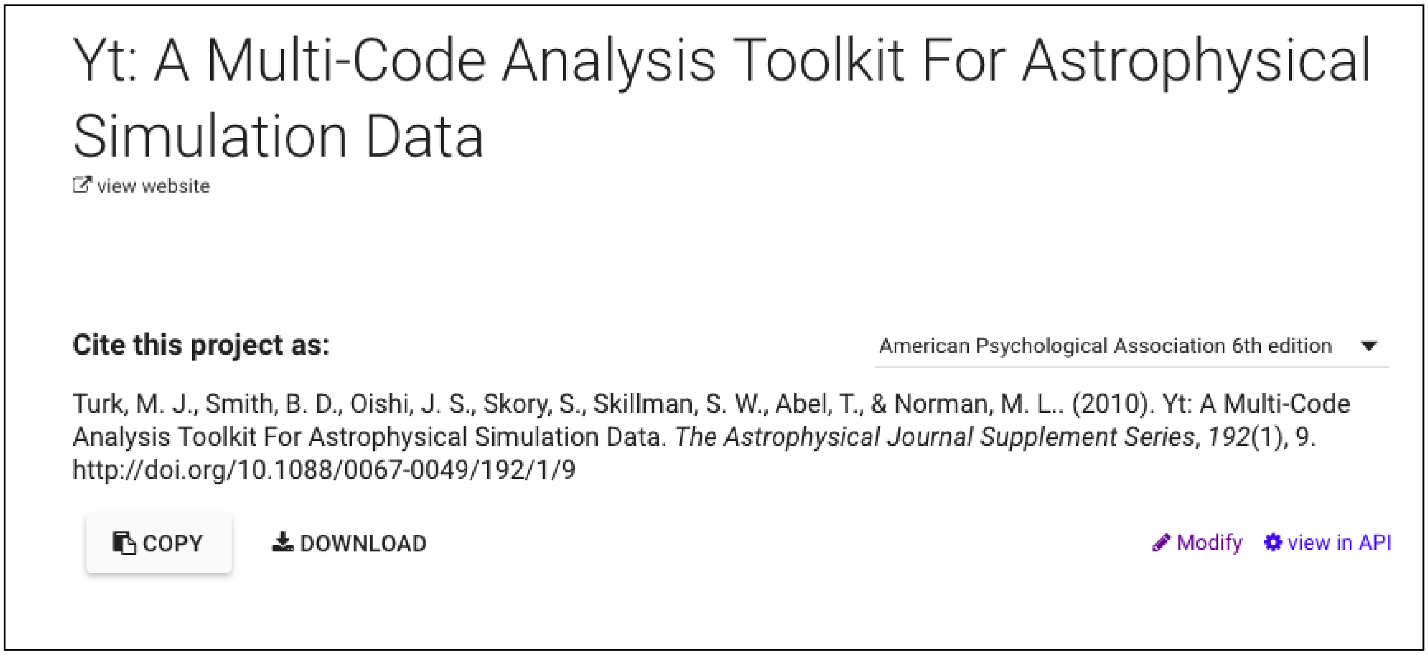
CiteAs includes a web-scraper that seeks out requests wherever they might be, following a set of logical rules based on how we've seen people ask for a citation. We ask users to start with something they know about the software, such as the project name, a project "landing page" (e.g., SciPy's requests for citations (https://www.scipy.org/citing.html)), or a project's repository URL. We then have plugins arranged in a sequence that obtain data from out on the web and seek the best citation request, prioritizing metadata by its imputed intentionality, such as `CITATION.cff` file, `CITATION` file, `citation()` calls, `DOAP` metadata, and metadata registered associated with a DOI (e.g., Zenodo's software DOIs (http://about.zenodo.org/principles/)). Of medium priority is metadata discovered through natural language requests on webpages (such as `bibtex` or other formats on landing pages). Finally we fall back to creating a simple citation to a repository or even web-page.

We want to discover and honor author's requests and simultaneously educate authors about how to make clearer or more specific citation requests, encouraging them to make use of more expressive formats. We do that by showing our discovery process and highlighting missing, higher intentionality, opportunities to make requests.

Examples

Example 1: YT

YT is a python package for analyzing and visualizing volumetric data. Entering the YT webpage URL into the CiteAs.org search field retrieves the correct citation in a variety of different common citation formats as well as in structured data that can be imported into Zotero in one click:



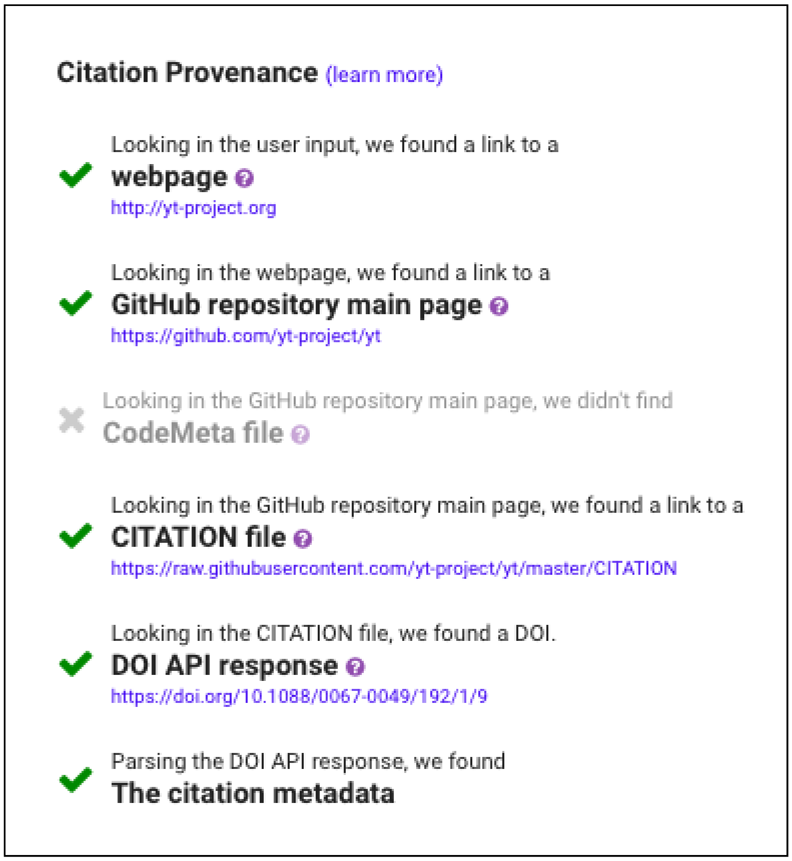
Yt: A Multi-Code Analysis Toolkit For Astrophysical Simulation Data

Cite this project as:

American Psychological Association 6th edition

Turk, M. J., Smith, B. D., Oishi, J. S., Skory, S., Skillman, S. W., Abel, T., & Norman, M. L.. (2010). Yt: A Multi-Code Analysis Toolkit For Astrophysical Simulation Data. *The Astrophysical Journal Supplement Series*, *192*(1), 9. http://doi.org/10.1088/0067-0049/192/1/9

Since a key goal of CiteAs is education, the **process used** to find this citation is also highlighted on the results page:



Citation Provenance (learn more)

✓ Looking in the user input, we found a link to a
webpage
http://yt-project.org

✓ Looking in the webpage, we found a link to a
GitHub repository main page
https://github.com/yt-project/yt

✗ Looking in the GitHub repository main page, we didn't find
CodeMeta file

✓ Looking in the GitHub repository main page, we found a link to a
CITATION file
https://raw.githubusercontent.com/yt-project/yt/master/CITATION

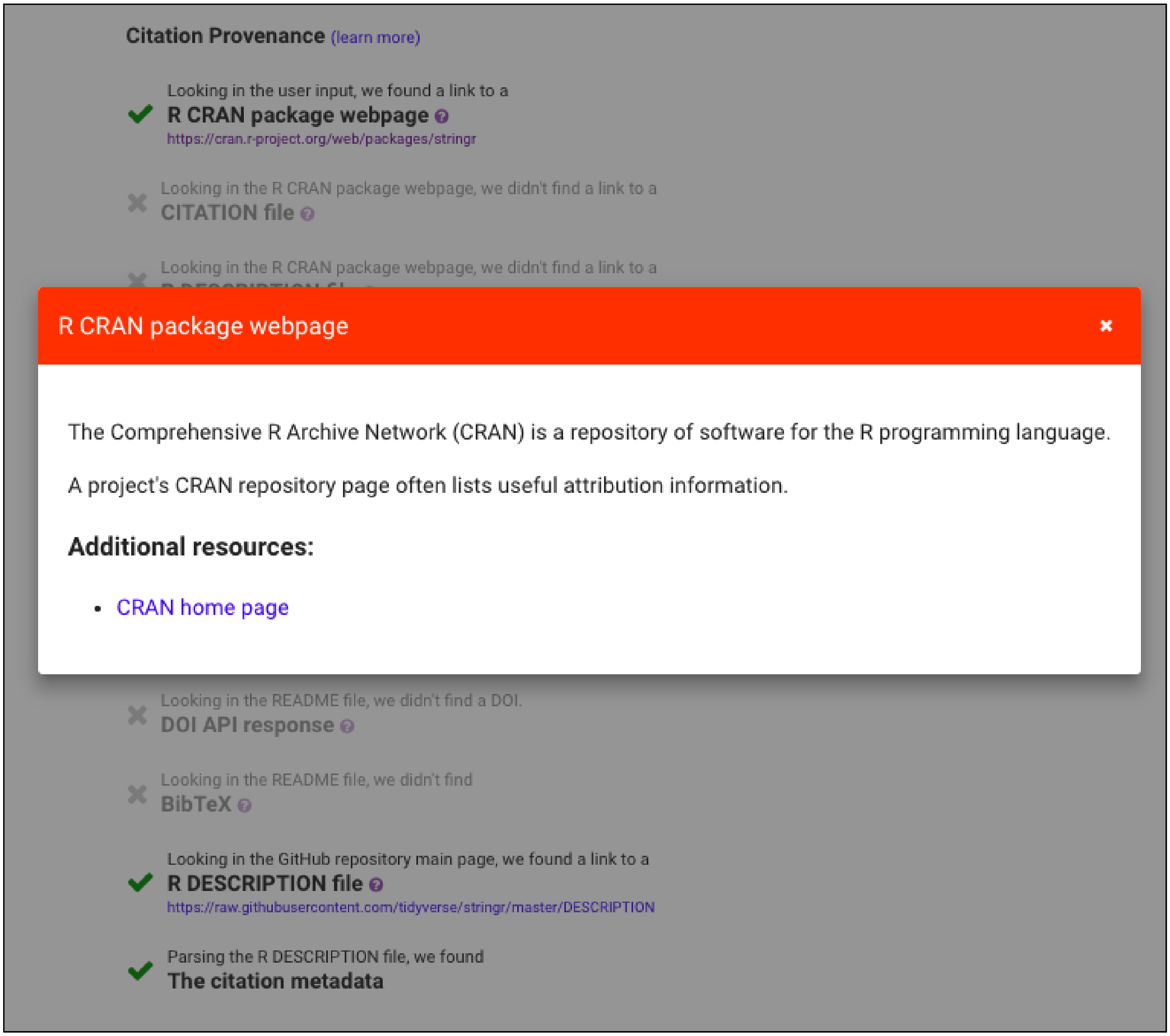
✓ Looking in the CITATION file, we found a DOI.
DOI API response
https://doi.org/10.1088/0067-0049/192/1/9

✓ Parsing the DOI API response, we found
The citation metadata

From this list we can see the steps the application took to find the citation—which both helps establish provenance and also educate users about better ways to register and discover software citation metadata.

Example 2: Stringr

The Stringr package provides wrappers for common string operations in R. Given the URL for the package's CRAN page, CiteAs finds and displays the correct citation for this package. CiteAs takes quite a few hops to finally figure out where the canonical citation metadata is, but it does eventually find it in the R DESCRIPTION file. Again, the display of all these steps helps users understand more about the possible approaches to software citation. The icon next to each step links users to additional documentation.



Citation Provenance (learn more)

✓ Looking in the user input, we found a link to a
R CRAN package webpage
https://cran.r-project.org/web/packages/stringr

✗ Looking in the R CRAN package webpage, we didn't find a link to a
CITATION file

✗ Looking in the R CRAN package webpage, we didn't find a link to a

R CRAN package webpage ×

The Comprehensive R Archive Network (CRAN) is a repository of software for the R programming language.

A project's CRAN repository page often lists useful attribution information.

Additional resources:

- CRAN home page

✗ Looking in the README file, we didn't find a DOI.
DOI API response

✗ Looking in the README file, we didn't find
BibTeX

✓ Looking in the GitHub repository main page, we found a link to a
R DESCRIPTION file
https://raw.githubusercontent.com/tidyverse/stringr/master/DESCRIPTION

✓ Parsing the R DESCRIPTION file, we found
The citation metadata

Challenges and next steps

Locating requests

Eventually we plan to incorporate an additional source: the manner in which packages are already being mentioned in publications. We plan to obtain this through machine learning of the literature (“entity recognition” for software). Towards this, we have trained content analytic coders labeling a randomly chosen set of publications and are making the labelled dataset available at Softcite Dataset (https://github.com/howisonlab/softcite-dataset). Using that system we plan to add “Here’s your current request and here’s how we see your software mentioned in the literature. If you’d like to change those practices you could start with a clear, standardized, machine-readable request.”

Presenting information

We have encountered plenty of challenges in designing the output of CiteAs to simultaneously realize our practical goals and our goal of educating users on clearer ways to make citations. We are still working towards better visualizations of the search process and ways of dealing with finding multiple different requests. We considered allowing users to “claim” their project and then to mark their preferred citation, but we want to improve existing infrastructure, rather than become centralized infrastructure ourselves. The system will therefore make recommendations about how to write clearer requests that everyone can read, rather than host those requests onsite.

Sustainability

CiteAs faces a key challenges that any grant-funded piece of software faces: how to continue after the end of the grant. Handling “bit rot” (or “software collapse” (http://blog.khinsen.net/posts/2017/01/13/sustainable-software-and-reproducible-research-dealing-with-software-collapse/)) will be a challenge, but in addition any web-hosted service has on-going financial needs for server space. Our by-no-means-perfect approach is to demonstrate the feasibility of the approach and seek partners to whom to pass off the service.

Please try CiteAs and report issues

We would love to hear your experiences with the CiteAs.org (http://citeas.org) service. We are especially interested in hearing about requests that CiteAs is not currently finding, as well as feedback on the presentation of the results, and the position of CiteAs within the ecosystem of related services. We are also very interested in efforts within software ecosystems or fields to provide requests for citation that we could collect. Report issues or opportunities on our GitHub issues page (https://github.com/Impactstory/citeas-webapp/issues).

Great software work → Clear requests for citation → More visibility in publications → More credit → Better Software → Better Research

URSSI (/)

(https://twitter.com/si2urssi) (https://github.com/si2-urssi)

About

Project Description (/about)

Blog (/blog)

Team (/about#team)

Resources

Discussion board (https://discuss.urssi.us/)

Survey (http://bit.ly/urssi-2018-survey)

Project Updates (/news)

Sign up for our mailing list (https://urssi.us17.list-manage.com/subscribe/post?u=34c9c3bb4d54665136bd03e49&id=f55b22de1

Newsletters (/newsletter)

Resources (/resources)

Contact (/contact)

Workshops

Berkeley workshop, April 2018 (/workshops/berkeley)

Chicago workshop, October 2018 (/workshops/chicago)

Software metrics workshop, January 2019 (/workshops/santa-barbara)

Incubator workshop, February 2019 (/workshops/college-park)

RSS Feeds

Blog Feed (http://urssi.us/blog/index.xml)

Workshop Feed (http://urssi.us/workshops/index.xml)

Final workshop, April 2019 (/workshops/chicago-final)

Contributions, corrections and suggestions for this website are welcome on the project repository (https://github.com/si2-urssi/website).