

Recognising the value of software: how libraries can help the adoption of software citation

Neil P. Chue Hong

*Software Sustainability Institute & University of Edinburgh
(@npch, N.ChueHong@epcc.ed.ac.uk)*

Jez Cope, The British Library

Patricia Herterich, University of Edinburgh

Daniel S. Katz, University of Illinois at Urbana-Champaign

Simon Worthington, TIB - German National Library of Science and Technology

*Acknowledging the work of the FORCE11 Software Citation Implementation WG
LIBER2021, 23-25 June 2021*



Software underpins research

- 69% of researchers at UK universities could not do their research without software ([Hettrick et al., 2014](#))
- Analysis of 40 Nature papers identified 32 which mention software: 211 mentions in total ([Nangia and Katz, 2017](#))

Software is indirectly referenced

Mention type	Count (n = 286)	Example
Citation to a related object	Cite to publication	105 <i>... was calculated using biosys (Swofford & Selander 1981).</i>
	Cite to users manual	6 <i>... as analyzed by the BIAevaluation software (Biacore, 1997).</i> <i>Reference List has: Biacore, I. (1997). BIAevaluation Software Handbook, version 3.0 (Uppsala, Sweden: Biacore, Inc)</i>
	Cite to project name or website	15 <i>... using the program Autodecay version 4.0.29 PPC (Eriksson 1998).</i> <i>Reference List has: ERIKSSON, T. 1998. Autodecay, vers. 4.0.29 Stockholm: Department of Botany.</i>
Difficult to assign credit	Instrument-like	53 <i>... calculated by t-test using the Prism 3.0 software (GraphPad Software, San Diego, CA, USA).</i>
	URL in text	13 <i>... freely available from http://www.cibiv.at/software/pda/.</i>
	In-text name mention only	90 <i>... were analyzed using MapQTL (4.0) software.</i>
	Not even name	4 <i>... was carried out using software implemented in the Java programming language.</i>

No PID

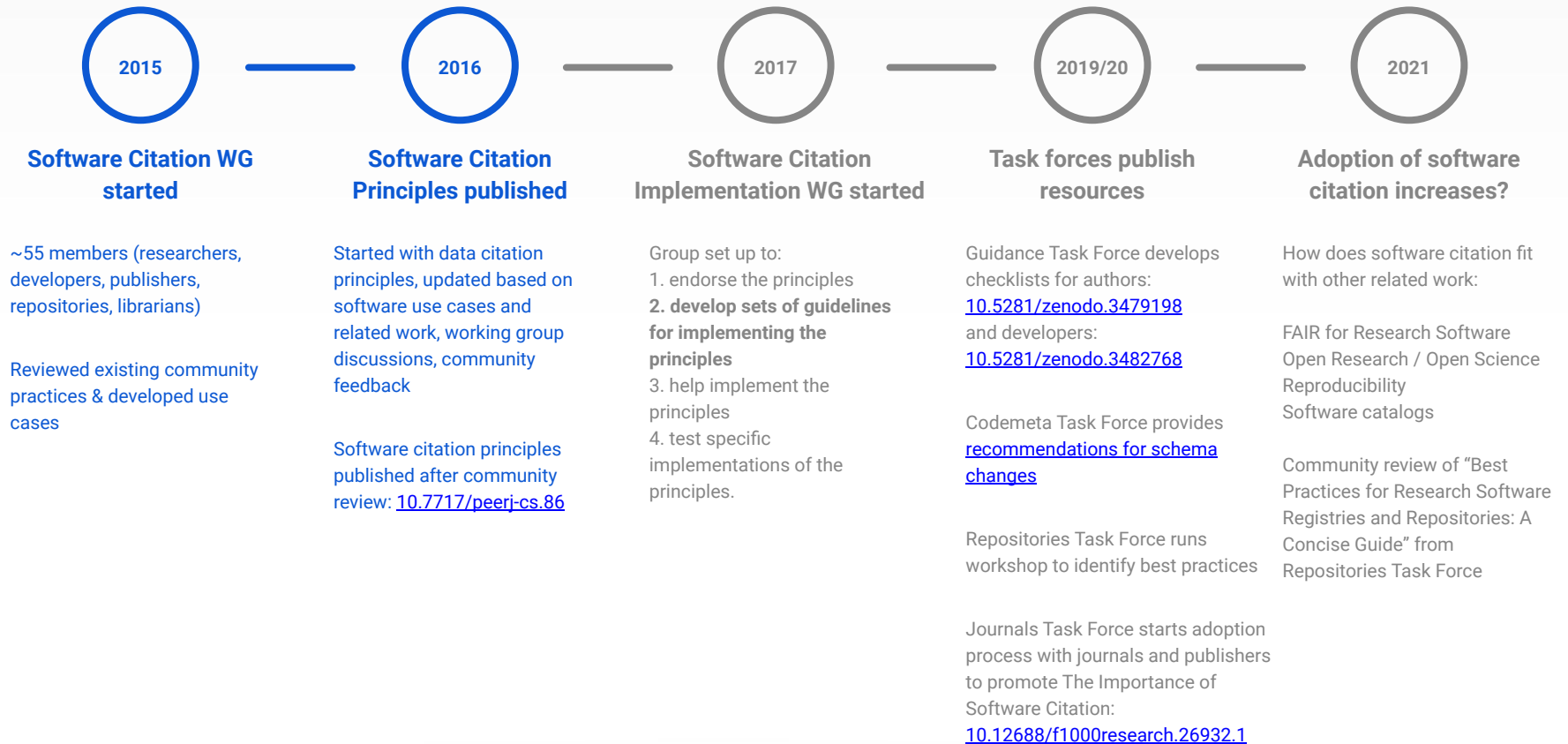
Table adapted from Table 6 in Howison and Bullard (2016). <https://doi.org/10.1002/asi.23538>

Value of citing software

- Supports proper attribution and credit
- Supports peer-review, validation, and reproducibility of findings
- Supports collaboration and reuse
- Encourages building on the work of others

But the typical “self-publication” model of software makes it harder to cite directly.

The journey so far...



1. Importance
2. Credit and Attribution
3. Unique Identification
4. Persistence
5. Accessibility
6. Specificity

SOFTWARE CITATION PRINCIPLES

IMPORTANCE

Software should be considered a legitimate and citable product of research. Software citations should be accorded the same importance in the scholarly record as citations of other research products; they should be included in the metadata of the citing work, such as a reference list. Software should be cited on the same basis as any other research product such as a paper or a book.

UNIQUE IDENTIFICATION

A software citation should include a method for identification that is machine actionable, globally unique, interoperable, and recognized by at least a community of the corresponding domain experts, and preferably by general public researchers.

ACCESSIBILITY

Software citations should facilitate access to the software itself and to its associated metadata, documentation, data, and other materials necessary for both humans and machines to make informed use of the referenced software.

PERSISTENCE

Unique identifiers and metadata describing the software and its disposition should persist—even beyond the lifespan of the software they describe.

CREDIT AND ATTRIBUTION

Software citations should facilitate giving scholarly credit and normative, legal attribution to all contributors to the software, recognizing that a single style or mechanism of attribution may not be applicable to all software.

SPECIFICITY

Software citations should facilitate identification of, and access to, the specific version of software that was used. Software identification should be as specific as necessary, such as using version numbers, revision numbers, or variants such as platforms.

Smith AM, Katz DS, Niemeyer KE, FORCE11 Software Citation Working Group.(2016) Software Citation Principles. PeerJ Computer Science 2:e86.
DOI: [10.7717/peerj-cs.86](https://doi.org/10.7717/peerj-cs.86) and <https://www.force11.org/software-citation-principles>

... yet there is little support for its acknowledgement and citation

From principles to implementation

Started Software Citation Implementation Working Group to:

- Write out the “small amount” of detail needed to implement the principles
- Work with communities to actually implement them
 - Publishers, conferences, repositories, indexers, funders, etc.
- Checklists are an effective way of ensuring consistency and completeness
 - See The Checklist Manifesto: How to Get Things Right by Atul Gawande for examples

Software Citation Guidelines

Recommended

Creator(s): authors or project that developed the software.

Title: the name of the software.

Publication venue: preferentially, an archive or repository that provides PIDs

Date: date (a release of version of) the software was published.

Identifier: a resolvable pointer to the software, preferentially, a PID that resolves to a landing page containing descriptive metadata about the software. If there is no PID for the software, a URL to where the software exists may be the best identifier available.

Optional

Version: the identifier for the version of the software being referenced.

Type: some citation styles (e.g., APA), require a bracketed description of the citation (e.g., Computer software) to be included.

If an article exists that describes the software, it should be cited as an additional reference, as well as citing the software itself

- Do not cite the article instead of the software

Katz et al., 2021.

DOI: [10.12688/f1000research.26932.2](https://doi.org/10.12688/f1000research.26932.2)

Guidance and Policies

Checklists:

- [Software Citation Checklist for Authors](#)
- [Software Citation Checklist for Developers](#)

CHORUS [Software Citation Policies Index](#) lists publisher policies on software citation

- AAAS, AAS, AGU, AMS, eLife, Elsevier, F1000Research, GigaScience Press, PLoS, Springer Nature

Examples (APA 7th edition)

Related Work section: *“In the field of numerical software, a different approach is taken by BLAS (BLAS team, n.d).”*

Methodology section: *“We used version 0.88 of Advanced Terrestrial Simulator (Coon et al., 2019) and version 25.0 of IBM SPSS Statistics for Windows (IBM Corp., 2017) to carry out the analysis of the data in this paper.”*

References

- BLAS team (n.d.), BLAS (Basic Linear Algebra Subprograms) [Computer software]. Netlib. <http://www.netlib.org/blas/>
- Coon, E., Berndt, M., Jan, A., Svyatsky, D., Atchley, A., Kikinzon, E., Harp, D., Manzini, G., Shelef, E., Lipnikov, K., Garimella, R., Xu, C., Moulton, D., Karra, S., Painter, S., Jafarov, E., & Molins, S. (2020, March 25). Advanced Terrestrial Simulator (ATS) v0.88 (Version 0.88) [Computer software]. Zenodo. <https://doi.org/10.5281/zenodo.3727209>
- IBM Corp. (2017). IBM SPSS Statistics for Windows (Version 25.0) [Computer software]. IBM Corp. <https://www.ibm.com/products/spss-statistics>

First is citation of a software concept, second an ideal citation with a long author list, third for commercial software where only the executable is available

The Role of Research Libraries

1. **Support and training:** to develop researchers' skills on the correct ways to include software in research output management plans, publish software and get PIDs, cite software and use reference managers
2. **Infrastructure:** to help support software citation and open access to software, such as digital repositories, software registries, identifiers and catalogues

Examples of practice

Support and Training

- MIT Libraries - [Citing and Publishing Software](#)
- TIB - [FAIR Data and Software Workshops](#)
- Library Carpentry - [Top 10 FAIR Data & Software Things - Thing 9: State how to cite your software](#)

Infrastructure

- [CaltechData Codemeta Integration](#)
- [Nine Best Practices for Research Software Registries and Repositories](#) (also include related best practice for research library infrastructure e.g. retention, scope policies)

Next steps

Publisher and repository support for software citation is rapidly progressing

- To facilitate adoption, it is essential that this guidance from research libraries is consistent

Opportunity for research libraries to collaborate with research software engineering / research computing groups at their institutions

- To provide broader support for open research, FAIR research objects, reproducibility and software preservation

Get Involved

- Join FORCE11 Software Citation Implementation WG
 - <https://www.force11.org/group/software-citation-implementation-working-group>
 - <https://github.com/force11/force11-sciwg>
- Contribute your experience / implementations
- Expect to see more journals publishing requirements for software citations