METHOD ARTICLE

REVISED Recognizing the value of software: a software citation guide [version 2; peer review: 2 approved]

Previously titled: "The importance of software citation"

Daniel S. Katz[1], Neil P. Chue Hong[2], Tim Clark[3], August Muench[4], Shelley Stall[5], Daina Bouquin[6], Matthew Cannon[7], Scott Edmunds[8], Telli Faez[9], Patricia Feeney[10], Martin Fenner[11], Michael Friedman[12], Gerry Grenier[13], Melissa Harrison[14], Joerg Heber[15], Adam Leary[16], Catriona MacCallum[17], Hollydawn Murray[18], Erika Pastrana[19], Katherine Perry[20], Douglas Schuster[21], Martina Stockhause[22], Jake Yeston[23]

[1]University of Illinois at Urbana-Champaign, Urbana, IL, USA
[2]EPCC, University of Edinburgh, Edinburgh, UK
[3]University of Virginia, Charlottesville, VA, USA
[4]American Astronomical Society, Washington, DC, USA
[5]American Geophysical Union, Washington, DC, USA
[6]Harvard-Smithsonian Center for Astrophysics, Cambridge, MA, USA
[7]Taylor & Francis Group, Oxford, UK
[8]GigaScience Press, BGI Hong Kong, Hong Kong, Hong Kong
[9]Elsevier, Amsterdam, The Netherlands
[10]Crossref, Lynnfield, MA, USA
[11]DataCite, Hannover, Germany
[12]American Meteorological Society, Boston, MA, USA
[13]Publishing Technology, IEEE, Piscataway, NJ, USA
[14]Production, eLife, Cambridge, UK
[15]PLOS, San Francisco, CA, USA
[16]Oxford University Press, Oxford, UK
[17]Open Science, Hindawi, London, UK
[18]F1000Research, London, UK
[19]Springer Nature, New York, NY, USA
[20]Product Management, Wiley, Boston, MA, USA
[21]National Center for Atmospheric Research, Boulder, CO, USA
[22]German Climate Computing Center (DKRZ), Hamburg, Germany
[23]AAAS, Washington, DC, USA

v2 **First published:** 19 Oct 2020, **9**:1257
https://doi.org/10.12688/f1000research.26932.1

Latest published: 12 Jan 2021, **9**:1257
https://doi.org/10.12688/f1000research.26932.2

Open Peer Review

Approval Status ✓ ✓

	1	2
version 2 (revision)	✓ view	

Abstract

Software is as integral as a research paper, monograph, or dataset in terms of facilitating the full understanding and dissemination of research. This article provides broadly applicable guidance on software citation for the communities and institutions publishing academic journals and conference proceedings. We expect those communities and institutions to produce versions of this document with software examples and citation styles that are appropriate for their intended audience. This article (and those community-specific versions) are aimed at authors citing software, including software developed by the authors or by others. We also include brief instructions on how software can be made citable, directing readers to more comprehensive guidance published elsewhere. The guidance presented in this article helps to support proper attribution and credit, reproducibility, collaboration and reuse, and encourages building on the work of others to further research.

Keywords
Software citation, publishing, scholarly communication, guidelines, bibliometrics

	1	2
12 Jan 2021	|	
version 1 19 Oct 2020	? view	✓ view

1. **Ludo Waltman**, Leiden University, Leiden, The Netherlands
2. **Gianmaria Silvello**, University of Padua, Padua, Italy

Any reports and responses or comments on the article can be found at the end of the article.

This article is included in the Research on Research, Policy & Culture gateway.

Corresponding author: Daniel S. Katz (d.katz@ieee.org)

Author roles: Katz DS: Conceptualization, Methodology, Project Administration, Supervision, Writing – Original Draft Preparation, Writing – Review & Editing; **Chue Hong NP**: Conceptualization, Methodology, Project Administration, Writing – Original Draft Preparation, Writing – Review & Editing; **Clark T**: Conceptualization, Methodology, Project Administration, Writing – Original Draft Preparation, Writing – Review & Editing; **Muench A**: Conceptualization, Methodology, Project Administration, Writing – Original Draft Preparation, Writing – Review & Editing; **Stall S**: Conceptualization, Methodology, Project Administration, Writing – Original Draft Preparation, Writing – Review & Editing; **Bouquin D**: Conceptualization, Writing – Review & Editing; **Cannon M**: Conceptualization, Writing – Review & Editing; **Edmunds S**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Faez T**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Feeney P**: Writing – Review & Editing; **Fenner M**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Friedman M**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Grenier G**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Harrison M**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Heber J**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Leary A**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **MacCallum C**: Conceptualization, Writing – Review & Editing; **Murray H**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Pastrana E**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Perry K**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Schuster D**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Stockhause M**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Yeston J**: Conceptualization, Writing – Original Draft Preparation, Writing – Review & Editing

Competing interests: Authors include representative and editors from: Journal of Open Source Software (Open Journals), Journal of Open Research Software (Ubiquity Press), AAS Journals, AGU Journals, Taylor & Francis, GigaScience Press, Elsevier, DataCite, American Meteorological Society, IEEE Publications, eLife, PLOS, Oxford University Press, Hindawi, F1000Research, Springer Nature, Wiley, Science Magazine. Hollydawn Murray is the Head of Data and Software Publishing at F1000 Research, but was not involved in the processing of this article.

Grant information: The author(s) declared that no grants were involved in supporting this work.

How to cite this article: Katz DS, Chue Hong NP, Clark T *et al.* **Recognizing the value of software: a software citation guide [version 2; peer review: 2 approved]** F1000Research 2021, **9**:1257 https://doi.org/10.12688/f1000research.26932.2

First published: 19 Oct 2020, **9**:1257 https://doi.org/10.12688/f1000research.26932.1

REVISED Amendments from Version 1

In response to reviewer feedback, and an additional comment from a reader, we have made the following changes to this article:

- A new title to better reflect the content and purpose
- At the end of the first section, an added sentence and two references to recognize previous work in data citation and the differences between software and data
- In the software citation essentials section, updated text on software versions and the software concept (the set of all versions).
- Also in that section, added text to explain the software publication date.
- Also in that section, updated text to emphasize citing the software itself citing an article about the software.
- The usage note about hardware requirements has been removed as confusing and beyond the scope of the article.

Any further responses from the reviewers can be found at the end of the article

Software is as integral as a research paper, monograph, or dataset in terms of facilitating the full understanding and dissemination of research. Books and journal articles have long benefited from an infrastructure that makes them easy to cite, a key element in the process of research and academic discourse in all disciplines. We believe that software (including computational code, scripts, models, notebooks and libraries) should be cited in the same way that other sources of information, such as articles and books, are cited.

Citing software helps further research and provides the means for other researchers to access software in order to:

- support proper attribution and credit (similar to that of papers, data, etc.);
- enable peer-review, validation, and reproducibility of findings;
- support collaboration and reuse; and
- encourage building on the work of others.

Software citation elevates software to the level of a first-class object in the digital scholarly ecosystem, consistent with its immense actual present-day significance.

FORCE11 has been developing guidance for software citation. The Software Citation Principles (Smith *et al.*, 2016) were written to encourage broad adoption of a consistent policy for software citation across disciplines and venues. The Software Citation Checklist for Authors (Chue Hong *et al.*, 2019a) and Software Citation Checklist for Developers (Chue Hong *et al.*, 2019b) provide more practical information for those seeking to improve their practice. This work has been influenced by prior work on Data Citation (Data Citation Synthesis Group, 2014), while recognizing that software is not the same as data in the context of citation (Katz *et al.*, 2016).

Software citation essentials

This article is aimed at authors citing software. This includes software developed by others, as well as software developed by any or all of the authors. Making software citable is a critical developer-led step, which is briefly detailed in the next subsection, “Making Software Citable”.

The use of persistent identifiers (PIDs) and core descriptive metadata are essential elements of software citation. This is because they are the mechanism used to index and track citations. We recognise that the challenges associated with software deposit and publication vary across disciplines, and we encourage research communities to develop citation systems that work well for them. We also recognise that the citation style formats used vary between disciplines and journals. Independent of the style of any citation, we recommend certain essential metadata elements should always be captured.

There are multiple use cases for citing software. These include referring to the software used in deriving the results of an article or discussing algorithms, general features, or concepts provided by a piece of software. If you used the software directly in the research described in your article (e.g., in the Methods section), then we recommend citing the specific version used (and the authors and publication date for that version). When discussing software more broadly, we recommend citing the software as a concept (project).

Our recommended format for software citation is to ensure the following information is provided as part of the reference:

- *Creator(s):* the authors or project that developed the software.
- *Title:* the name of the software.
- *Publication venue:* the publication venue of the software, preferentially, an archive or repository that provides persistent identifiers.
- *Date:* the date the software was published. This is the date associated with a release or version of the software, or “n.d.” if the date is unknown.
- *Identifier:* a resolvable pointer to the software, preferentially, a PID that resolves to a landing page containing descriptive metadata about the software, similar to how a Digital Object Identifier (DOI) for a paper that points to a page about the paper rather than directly to a representation of the paper, such as the PDF. DOIs are preferable, and other examples of PIDs include Handles, RRIDs, ASCL IDs, swMath IDs, Software Heritage IDs, ARKs, etc. If there is no PID for the software, a URL to where the software exists may be the best identifier available.

It may also be desirable, and depending upon the publisher, may be required, to include information about two optional properties (as appropriate):

- *Version:* the identifier for the version of the software being referenced. If the version is unidentified or unknown, the date of access should be used.

- *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.

Making software citable

Authors should consult the Software Citation Checklist for Developers (Chue Hong *et al.*, 2019b) for information on how to obtain a PID or choose a software license for software they have developed. That document contains a set of steps that developers can take to ensure that they are following good practices. We strongly recommend that journals provide such information to their authors, either by referring to that document, or using text from it or similar text. Example guidance would include instructing authors to version their software, choose a license for their software, perhaps by linking to the information at choosealicense.org, record metadata about the software as part of the repository, deposit their software in a preservation repository that provides a PID, and advertise the recommended citation in the repository. In particular, guidance should explicitly mention that Creative Commons licenses (including CC-BY) must not be used for software, and an open source license should be used.

Software citation examples

The following examples show how software can be cited in one common citation style, APA. The general format for downloaded software, from Section 10.10 of (2020) Publication Manual of the American Psychological Association (Seventh Edition) is:

- Developer, A. A., Developer, B. B., & Developer, C. C. (yyyy)[1]. *Title of the software: Subtitle* (Version #.#)[2] [Computer software][3]. Publisher[4], https://URL[5]

If no version number or version string exists, we (the FORCE11 Software Citation Implementation Working Group) modify this to:

- Developer, A. A., Developer, B. B., & Developer, C. C. (yyyy). *Title of the software: Subtitle* [Computer software]. Archive Name. Retrieved Month dd, yyyy, from https://URL

The following are examples of software citations.

Ideal citations to the specific version of the software, where all recommended information is present (the first demonstrates a large author list; the second demonstrates a project team as the author):

- 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
- Lab For Exosphere And Near Space Environment Studies. (2019, March 20). *lenses-lab/LYAO_RT-2018JA026426: Original Release* (Version 1.0.0) [Computer software]. Zenodo. https://doi.org/10.5281/zenodo.2598836

Citation referencing software that is preserved in a software archive (e.g. Software Heritage)[6]:

- Delebecque, F., Gomez, C., Goursat, M., Nikoukhah, R., Steer, S., & Chancelier, J.-P. (1994). *Scilab* (Version 1.1) [Computer software]. Software Heritage, swh:1:dir:1ba0b67b5d0c8f10961d878d91ae9d6e499d746a;origin=https://hal.archives-ouvertes.fr/hal-02090402
- Di Cosmo, R. & Danelutto, M. (2020). *The Parmap library: Core mapping routine* (Version 1.1.1) [Computer software]. Software Heritage, swh:1:cnt:43a6b232768017b03da934ba22d9cc3f2726a6c5;lines=192-228;origin=https://github.com/rdicosmo/parmap

A citation for software that does not have a PID but does have a version and identifier (URL), where authorship is assigned to the project as a whole:

- Dataverse Project (2020). *Dataverse* (Version 4.20) [Computer software] https://github.com/IQSS/dataverse/releases/tag/v4.20

A citation for software where there is no version identified and where the publishing date is unknown:

- Thomas, J. & Daujotas, G.[7] (n.d.). *is-thirteen* [Computer software]. GitHub. Retrieved June 17, 2020 from https://github.com/jezen/is-thirteen

A citation for a software concept (all versions):

- BLAS team (n.d.), *BLAS (Basic Linear Algebra Subprograms)* [Computer software]. Netlib. http://www.netlib.org/blas/

[1] The year is required, or "n.d." if not identifiable.

[2] The version is optional but preferred. Note that the version may be a token/string that is not a semantic version (https://semver.org/) and that must be exactly preserved, such as a commit hash (e.g., a149dbc00fe8b0e8260f7c2d39c77692683e7fa4), a semi-numeric tagged release (e.g., v0.4-alpha01), or date string (e.g., 2020-02-20).

[3] APA style includes additional information that is helpful for software citation (e.g. it requires the [Computer software] bracketed description). Although this is not part of our guidance above, we recommend following APA style and including these elements. Other styles may not use this extra information.

[4] If the software is downloaded or if the developer is the same as the publisher, the publisher name is omitted.

[5] In APA style, the URL is used for both URLs and DOIs or other PIDs, e.g., a DOI is expressed as https://doi.org/DOI.

[6] This example is analogous to citing the preserved version of a webpage on archive.org, rather than the webpage directly.

[7] The README for the *is-thirteen* software says "A helpful tool by Jezen Thomas with helpful help from Gytis Daujotas and many fine folk."; therefore our citation tries to take the developers intentions around authorship into account.

A citation for software where little information is available, perhaps where only the executable program is available. For commercial software, a link to information about availability for purchase is helpful, as shown in the example below.

- IBM Corp. (2017). *IBM SPSS Statistics for Windows* (Version 25.0) [Computer software]. IBM Corp. https://www.ibm.com/products/spss-statistics

In-text referencing

Two examples of how the citations above would be referenced in the text of a paper according to APA style[8], the first in the methodology section and the second in a related work 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.
- In the field of bibliometrics, a different approach is taken by BLAS (BLAS team, n.d).

Usage note

This document provides generic guidance about software citation for the communities and institutions publishing academic journals and conference proceedings. We expect those communities and institutions to produce different versions of this document with software examples and citation styles that are appropriate for their intended audience. We request that those documents refer back to (or cite) this one. This document can be cited (in APA 7th Ed. style) as:

- Katz, D. S., Chue Hong, N. P., Clark T., Muench, A., Stall, S., Bouquin, D., Cannon, M., Edmunds, S., Faez, T., Farmer, R., Feeney, P., Fenner, M., Friedman, M., Grenier, G., Harrison, M., Heber, J., Leary, A., MacCallum, C., Murray, H., … Yeston, J. (2020) *Recognizing the value of software: a software citation guide.* F1000 Research. https://doi.org/10.12688/f1000research.26932.2

Data availability

No data is associated with the article.

Acknowledgements

This article is based in part on data citation guidance published by DataCite (Datacite), and on related publications from FORCE11 working groups (Cousijn *et al.*, 2018; Fenner *et al.*, 2019). It was initially drafted by Neil Chue Hong, and further developed by Daniel S. Katz, Neil Chue Hong, Tim Clark, August Muench, and Shelley Stall, along with many participants in the FORCE11 Software Citation Implementation Working Group's Journals Task Force. We also acknowledge useful advice from Kevin Swanson, Taylor & Francis.

[8] American Psychological Association. (2020). *Publication manual of the American Psychological Association* (7th ed.). American Psychological Association. https://doi.org/10.1037/0000165-000

References

Chue Hong NP, Allen A, Gonzalez-Beltran A, *et al.*: **Software Citation Checklist for Authors (Version 0.9.0).** *Zenodo.* 2019a.
Publisher Full Text

Chue Hong NP, Allen A, Gonzalez-Beltran A, *et al.*: **Software Citation Checklist for Developers (Version 0.9.0).** *Zenodo.* 2019b.
Publisher Full Text

Cousijn H, Kenall A, Ganley E, *et al.*: **A data citation roadmap for scientific publishers.** *Sci Data.* 2018; **5**: 180259.
PubMed Abstract | **Publisher Full Text** | **Free Full Text**

Data Citation Synthesis Group: **Joint Declaration of Data Citation Principles.** Future of Research Communication and e-Scholarship (FORCE11). 2014.
Publisher Full Text

DataCite: **DataCite - Cite Your Data.**
Reference Source

Fenner M, Crosas M, Grethe JS, *et al.*: **A data citation roadmap for scholarly data repositories.** *Sci Data.* 2019; **6**(1): 28.
PubMed Abstract | **Publisher Full Text** | **Free Full Text**

Katz DS, Niemeyer KE, Smith AM, *et al.*: **Software vs. data in the context of citation.** *PeerJ Prepr.* 2016; **4**: e2630v1.
Publisher Full Text

Smith AM, Katz DS, Niemeyer KE, *et al.*: **Software Citation Principles.** *PeerJ Comput Sci.* 2016; **2**: e86.
Publisher Full Text

Open Peer Review

Current Peer Review Status: ✓ ✓

Version 2

Reviewer Report 13 January 2021

https://doi.org/10.5256/f1000research.47168.r77167

✓ **Ludo Waltman**

Centre for Science and Technology Studies (CWTS), Leiden University, Leiden, The Netherlands

I am happy with the revised version of this article. I don't have any requests for further changes.

Competing Interests: I am working together with Catriona MacCallum in the Initiative for Open Abstracts (I4OA; https://i4oa.org/) and in a research project of the Research on Research Institute (http://researchonresearch.org/). In both cases I feel this has not affected my impartiality.

Reviewer Expertise: Scientometrics, quantitative science studies, open science

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Version 1

Reviewer Report 07 December 2020

https://doi.org/10.5256/f1000research.29749.r75320

✓ **Gianmaria Silvello**

Department of Information Engineering, University of Padua, Padua, Italy

This paper presents an overview of software citation detailing state of the art and providing some indications about how software should be cited in different contexts.

There is no innovative method presented, but rather this is a set of community-driven guidelines that can be really useful as a starting point to provide adequate software citations.
I am not sure this paper is a good fit as a method article for this journal, but it is a good fit for the journal. To me, it can be something in-between a method article and an opinion paper.
What can be considered missing from this paper are considerations or references to transitive citations, which are central for software citation. Nevertheless, this topic may be out of scope for a contribution like this one.
Anyway, the paper is well-written, and to me, it can be published as-is provided that it is clear that no major innovative contribution is described or new insight is presented.

Is the rationale for developing the new method (or application) clearly explained?
Yes

Is the description of the method technically sound?
Yes

Are sufficient details provided to allow replication of the method development and its use by others?
Yes

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?
No source data required

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?
Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Databases, Data citation, Information retrieval and Digital Libraries,

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Author Response 04 Jan 2021

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

Thank you for your review and suggestions. In our newly submitted revision, we considered your point

- What can be considered missing from this paper are considerations or references to transitive citations, which are central for software citation. Nevertheless, this topic may be out of scope for a contribution like this one.

but we feel this is indeed out of scope for this paper.

Competing Interests: No competing interests were disclosed.

Reviewer Report 17 November 2020

https://doi.org/10.5256/f1000research.29749.r73368

?

Ludo Waltman
Centre for Science and Technology Studies (CWTS), Leiden University, Leiden, The Netherlands

This is a very useful contribution. I have some minor comments.

Discussions about software citation and data citation are closely related. I would therefore find it helpful to read something about the way in which the guidance on software citation provided in this document relates to standards for data citation. It seems important that standards for software citation and data citation are consistent as much as possible.

The title of the contribution ("The importance of software citation") suggests that the contribution focuses on arguing for the importance of software citation. However, as explained in the abstract, the focus in fact is on providing "broadly applicable guidance on software citation". My suggestion therefore is to revise the title. An alternative title for instance could be "How to cite software?".

"We recommend citing the specific version used (and the authors and publication date for that version) if you used it directly in the research described in your publication (e.g., the Methods section). We recommend citing the software concept (project) if you are referencing the software elsewhere in your paper.": I don't fully understand the distinction that is made in these two sentences. The authors seem to have in mind a distinction between citing software because it is used directly in a research project and citing software for other reasons. I would like to know more about what other reasons for citing software the authors have in mind and why they believe citations should be made in different ways in the two situations they distinguish.

"If a published article exists that describes the software, it should be cited as an additional reference.": The motivation for this recommendation is not clear to me. The authors seem to give special treatment to published articles, by which I assume they have in mind articles published in scholarly journals. I find this questionable. Suppose we have two pieces of software. Software A is documented in a two-page article published in a scholarly journal. Software B is documented in a comprehensive report made available in GitHub. Why should the article documenting software A be cited, while the report documenting software B does not need to be cited? Note that the article documenting software A probably cannot be updated, and the article is therefore likely to provide an outdated description of the software. The report documenting software B can be updated and therefore is likely to offer an up-to-date description of the software.

“Hardware is important, but we have initially chosen not to overload software citations with hardware requirements directly. This might be better done through linkage between DOIs.”: I don’t understand these two sentences. Some additional explanation would be helpful.

Is the rationale for developing the new method (or application) clearly explained?
Yes

Is the description of the method technically sound?
Yes

Are sufficient details provided to allow replication of the method development and its use by others?
Yes

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?
Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?
Yes

Competing Interests: I am working together with Catriona MacCallum in the Initiative for Open Abstracts (I4OA; https://i4oa.org/). I am working together with Joerg Heber and Catriona MacCallum in a research project of the Research on Research Institute (http://researchonresearch.org/). In both cases I feel this has not affected my impartiality.

Reviewer Expertise: Scientometrics, quantitative science studies, open science

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Reader Comment 04 Dec 2020

Teresa Gomez-Diaz, CNRS/LIGM, France

In order to contribute to this interesting scientific discussion, we would like to point out some other aspects that could be considered.

This is an interesting work, both the article and the report, as it contributes to a sounder installation of best practices to be adopted by scientific communities in order to reference and cite research outputs other than articles.

The section "Software citation essentials" mentions:

- *The use of persistent identifiers (PIDs) and core descriptive metadata are essential* *elements of software citation. This is because they are the mechanism used to index and track citations.*

We would like to ask for further explanation and references in order to better understand this important mechanism.

- *Date: the date the software was published.*

This date point is an interesting issue, as much of the research software used and produced in the scientific communities has not gone through a thorough publication process involving some review procedure. They have been disseminated in a web page, a forge or a deposit like Zenodo or GitHub, usually associating a date to the disseminated version of the software. On the other hand, recent journals in the scientific publishing world have been created to publish software papers, which establishes a publication date. The authors could provide further explanation on the kind of date of publication that is considered in their recommendations.

Furthermore, the following recent work:

Gomez-Diaz T. and Recio T.
On the evaluation of research software: the CDUR procedure [version 2; peer review: 2 approved].
F1000Research 2019, 8:1353 (https://doi.org/10.12688/f1000research.19994.2)

studies referencing and citation issues in the context of research software (see the section 2.5 of "On the evaluation of research software: the CDUR procedure") and it could be interesting for the authors of the document we are commenting here to compare both approaches.

These comments have been prepared with the second author of this publication (T. Recio).

Competing Interests: I have collaborated with one of the authors of the commented document in the last four years (grant application). I have also participated in the assessment of a project involving another of the authors. I do not feel that these collaborations could have affected my impartiality.

Author Response 04 Jan 2021

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

Thank you for your comments.

We have just submitted a revised version that adds some additional description to explain the item about the software's publication date, as you requested.

Regarding your second point, suggesting that discuss the recent CDUR work, we believe this paper has a fairly narrow focus, and that a future expanded or follow-on paper would be a better place to compare with that work, as well as much other related work.

Competing Interests: No competing interests were disclosed.

Author Response 04 Jan 2021

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

Thank you very much for your careful reading and useful comments and suggestions. We have just submitted a revised version of the paper, which has the following changes made in response:

- Discussions about software citation and data citation are closely related. I would therefore find it helpful to read something about the way in which the guidance on software citation provided in this document relates to standards for data citation. It seems important that standards for software citation and data citation are consistent as much as possible.

We've added a sentence at the end of this paragraph to recognize the connection to work on data citation, and to point readers to references for more information.

- The title of the contribution ("The importance of software citation") suggests that the contribution focuses on arguing for the importance of software citation. However, as explained in the abstract, the focus in fact is on providing "broadly applicable guidance on software citation". My suggestion therefore is to revise the title. An alternative title for instance could be "How to cite software?".

We agree with this comment, and have changed the title in response.

- "We recommend citing the specific version used (and the authors and publication date for that version) if you used it directly in the research described in your publication (e.g., the Methods section). We recommend citing the software concept (project) if you are referencing the software elsewhere in your paper.": I don't fully understand the distinction that is made in these two sentences. The authors seem to have in mind a distinction between citing software because it is used directly in a research project and citing software for other reasons. I would like to know more about what other reasons for citing software the authors have in mind and why they believe citations should be made in different ways in the two situations they distinguish.

We agree that this was not clear as written, and have rewritten these sentences.

- "If a published article exists that describes the software, it should be cited as an additional reference.": The motivation for this recommendation is not clear to me. The authors seem to give special treatment to published articles, by which I assume they have in mind articles published in scholarly journals. I find this questionable. Suppose we have two pieces of software. Software A is documented in a two-page article published in a scholarly journal. Software B is documented in a comprehensive report made available in GitHub. Why should the article documenting software A be cited, while the report documenting software B does not need to be cited? Note that the article documenting software A probably cannot be updated, and the article is therefore likely to provide an outdated description of the software. The report documenting software B can be updated and therefore is likely to offer an up-to-date description of the software.

We have removed "published", as this was not an important part of the point we were trying to make, and have adjusted the text to make our point more clearly.

- “Hardware is important, but we have initially chosen not to overload software citations with hardware requirements directly. This might be better done through linkage between DOIs.”: I don’t understand these two sentences. Some additional explanation would be helpful

This text made sense in a much earlier version of the paper, but now we agree that this point was confusing as written and also feel it is not important to the article, so we have removed it.

Competing Interests: No competing interests were disclosed.

Comments on this article

Version 2

Reader Comment 15 Jan 2021

Oliver Bertuch, Forschungszentrum Jülich GmbH, Jülich, Germany

A very well written overview about how to make research software more important to research and researchers. If my comment below is out of scope for this doc, please feel free to disregard.

In many conversations with RDM people, there seem to form two "schools" of depositing software and making it citable. One is fostering bundles of software and data together in one package/container, the other encourages making seperate deposits. There are pros and cons in both approaches.

In my humble opinion it would help scientists to make an informed decision when this aspect is picked up in articles and guides like this. Maybe this is better suited for the checklist etc to explain all pesky details, yet a hint about taking care and where to find more information would be perfect.

Competing Interests: No competing interests were disclosed.