

Software Citation - Current Practice and Recent Developments

This workshop is part of the [Love your data? Make it reproducible!](#) event which is organized in context of the [International Love Data Week 2023](#).

The workshop bases on the materials of the HIFIS workshop [Foundations of Research Software Publication](#).



Agenda

- **Introduction and Overview** (60 min)
- **Exercise: Write a CITATION.cff file** (30 min)
- **Closing Discussion and Wrap Up** (30 min)

Introduction and Overview

Introduction Round

Please introduce yourself briefly:

- What is your background?
- What is your interest in the workshop?
- What are your experiences with referencing software in a research paper?
 - How did you reference it?
 - What problems did you have to reference another person's software?

Introductory Talk

Please add your questions below:

What's the issue with stack overflow and licenses that you mentioned?

- If no other license is mentioned, the text/code is licensed under [CC-BY-SA-3.0](#). This license makes it much harder to reuse the code snippet in your actual code base.
- However, it does not apply if the threshold of originality is too low. For example, a simple "Hello World" program usually does achieve that level of originality. But details depend on the concrete case.

Should we clearly distinguish between publishing SW in order to document the current status and to be cited on one side, and the maintenance of the SW on the other side? Not all SW developed and used for some articles is worth to maintain over long time.

- This is true. But also the "one-shot script" should be made citable. For example, you keep the code in your usual Git repository and only archive the specific snapshot of the script on Zenodo.
- For software that is longer maintained, it would be worth to automate the archival step.

Why .cff and not the established .bib?

BibTeX is not suited well for software citation metadata as it is too generic and it misses options to express certain semantics such as required metadata fields. In particular, we also want to be able to define metadata related to the FAIR principles while BibTeX is quite focused on the citation use case only.

Can we use CFF also in gitlab repositories with the nice link to zenodo?

- The badges should already work because they are provided by Zenodo.
- [The HERMES project is currently working on something similar/more generic](#) like the [GitHub Zenodo integration](#).

Are there requirements for publishing software/data on zenodo (since they're giving you the valuable DOI...)?

- No, currently there are no requirements.
- You can create an account and directly start using it.

Further Readings

- [Helmholtz Open Science Policy](#)
- [Software Citation Notes of the HIFIS Workshop "Foundations of Research Software Publication"](#)
- [Software Citation Principles](#)
- [Citation File Format](#)
- [The HERMES Project](#)

Exercise: Write a CITATION.cff file

Example

"The data sets and the notebook containing the analysis details have been published separately [11]."

References

- ...
- [11] Schlauch, Tobias & Haupt, Carina. (2019). Analysis of the DLR Knowledge Exchange Workshop Series on Software Engineering (Version 1.2.0). Zenodo. <https://doi.org/10.5281/zenodo.3403991>

How to make your Software citable?

- **Step 1: Provide citation metadata for your software!**
 - Define the citation metadata for your software:
 - Manage citation metadata in your code repository using the Citation File Format (CFF)
- **Step 2: Publish your software in a way suitable for software citation!**
 - Publish software including citation metadata to a publication repository (standardized metadata, archived snapshots, persistent identifiers)
 - Publish every software release

More Examples

- ESMValCore: <https://doi.org/10.5281/zenodo.3387139>
- CosmoScout VR: <https://doi.org/10.5281/zenodo.3381953>
- gitLab2prov: <https://doi.org/10.5281/zenodo.3624166>

You can look up more citable software via the Research Software Directory of the Netherland's escience center: <https://research-software-directory.org/software>

[Task]: Please write a CITATION.cff file for a software or a data set.

1. Decide which software or data set you want to use. If in doubt, you can use the [data set of our workshop example published on Zenodo](#).
2. Determine the relevant citation metadata for a specific software/data set release. This includes:
 - Software name
 - Authors
 - Identifier for the exact software version:
 - Recommended: Persistent identifier such as the DOI
 - Repository URL + exact version (tag name, Git commit ID)
 - Version number (e.g., 1.2.0)
 - Publication date
3. Create an initial CITATION.cff using the online editor [cffinit](#).

[Discussion]: Please reflect about your experiences creating the CITATION.cff file.

What worked well?

- Basically easy +2
- Good explanation of single fields (especially authors)

Which problems did you encounter?

- Too many licenses more guidance on this topic
- Multiple licenses are not supported well => see the [open issue](#)

Closing Discussion and Wrap Up

[Discussion]: Required support in research centers for the publication of citable software

What kind of support do research centers have to provide to enable their researchers to publish citable software?

- Clarify which licenses are preferred
- Provide guidance on publishing closed code / software
- Missing best practices for authorship in software
- Coding guidelines, support to ensure a certain code quality

What about the support in your organization? What is already available? What are you missing?

- Own gitlab server is available +1

- Agenda
- Introduction and Over...
- Exercise: Write a CITATI...
- Closing Discussion and...

Expand all
 Back to top
 Go to bottom