OSPO Explainer Transcript: Citing Research Software

URL: https://ospo.library.jhu.edu/ospo-explainers/

Slide 1: Hello, and welcome to the Johns Hopkins University Open Source Programs Office Explainer Series, a collection of bite-sized videos tackling a variety of open-source essentials. New topics are added regularly, check out ospo.library.jhu.edu to see what's coming up, or drop us a line at ospo@jhu.edu to make a request. Today's explainer? Citing Research Software.

Slide 2: Research software is the collection of tools, code, or code libraries that allow a researcher to generate new data or analyze and make meaning of existing data.[1]

Slide 3: Software is foundationally important to scientific and social progress; however, traditional acknowledgement of the use of others' work has not adapted in step with the rapid development and use of software in research.[2]

Slide 4: One way to improve acknowledgement of software is to create and use software citations.

Slide 5: Citations are great! But, when should you cite software? How do you create a citation for others to use?

Slide 6: From the Software Citation Checklist for Authors, "You should cite software that has a significant impact on the research outcome presented in your work, or on the way the research has been conducted. If the research you are presenting is not repeatable without a piece of software, then you should cite the software. Note that the license or copyright of the software has no bearing on whether you should cite it."[3]

Slide 7: In general, you should cite:

- Software (including scripts) you have written yourself to conduct the research presented.
- A software framework or platform upon which the software you wrote relies.
- Software packages, plugins, modules and libraries you used to conduct your research and that perform a critical role in your results.
- Software you have used to simulate or model phenomena or systems.
- Specialist software used to prepare, manage, analyse or visualise data.

---

[1] "Research Software Programs." Accessed March 14, 2024. https://researchsupport.harvard.edu/research-software-programs

[2] Bouquin, Daina, Ana Trisovic, Oliver Bertuch, and Elena Colón-Marrero. "Advancing Software Citation Implementation (Software Citation Workshop 2022)." arXiv, February 15, 2023. http://arxiv.org/abs/2302.07500.

[3] Chue Hong, Neil P., Alice Allen, Alejandra Gonzalez-Beltran, Anita de Waard, Arfon M. Smith, Carly Robinson, Catherine Jones, et al. "Software Citation Checklist for Authors." Zenodo, October 15, 2019. https://doi.org/10.5281/zenodo.3479199.

- Software being evaluated or compared as part of the research presented.
- Software that has produced analytic results or other outputs, especially if used through an interface.

Slide 8: In general, you do not need to cite software packages or libraries that are not fundamental to your work and that are a normal part of the computational and scientific environment used. These dependencies do not need to be cited outright but should be documented as part of the computational workflow for complete reproducibility.

You also do not need to cite software that was used during the course of the research but had no impact on research results, e.g. word processing software, backup software.

Slide 9: Next, how do you create a citation for others to use? The best way to create a citation for your work that others can find is to add a .cff file to your code repository.

CFFINIT is a Citation File Format webapp that can generate or update a validated citation file for you based on a few inputs. The link to CFFINIT is on this slide and in the transcript.

https://citation-file-format.github.io/cff-initializer-javascript/#/

If you have deposited your code into the Johns Hopkins Research Data Repository, a standard citation will be created based on your deposit's metadata.

Slide 10: If you add a CITATION.cff file to the default branch of a GitHub repository, a link will be added to the right sidebar of your repository landing page with the label "Cite this repository."[4]

Repository visitors can click on the link in the sidebar and copy an APA or BibTex-formatted citation.

Visit the OSPO website to see an example of a citation file and the sidebar link:

https://github.com/JH-OSPO/citation-examples

Slide 11: There are only four required elements in the standard citation schema: authors, cff-version, which is the version of the cff schema you are using, not the version of your software, message, and title.

Slide 12: That said, adding half a dozen more elements will make your citation much more useful. Recommended additions include an abstract, the date of the software release the citation refers to, any identifiers such as a DOI, keywords, the software's license, the URL for the repository, and the version of the software.

---

[4] GitHub Docs. "About CITATION Files." Accessed August 1, 2024.
https://docs.github.com/en/repositories/managing-your-repositorys-settings-and-features/customizing-your-repository/about-citation-files.

Slide 13: To view the complete cff schema with definitions, formatting requirements, and examples, visit the Citation File Format's GitHub repository:

https://github.com/citation-file-format/citation-file-format/blob/main/schema-guide.md

Slide 14: Finally, it's good practice to regularly review your citation files. The CFFINIT tool linked earlier can be used to create new .cff files if lots of things have changed, or to edit existing files if you just need to make a few tweaks.

Events that might kick off a citation file review or update include a new release with significant changes or improvements, new contributors joining the project, changes or modifications to the software's license, or changes in official project website or repository location.

Slide 15: Have any questions about what you've just learned? Ask the JHU Open Source Programs Office for more information. You can reach out to us online at ospo.library.jhu.edu or send an email to ospo@jhu.edu.