HELMHOLTZ

RESEARCH FOR
GRAND CHALLENGES

Foundations of Research Software Publication

Enabling reproducibility in data science - learn why it matters and how you can do it, Online, 09.06.2022

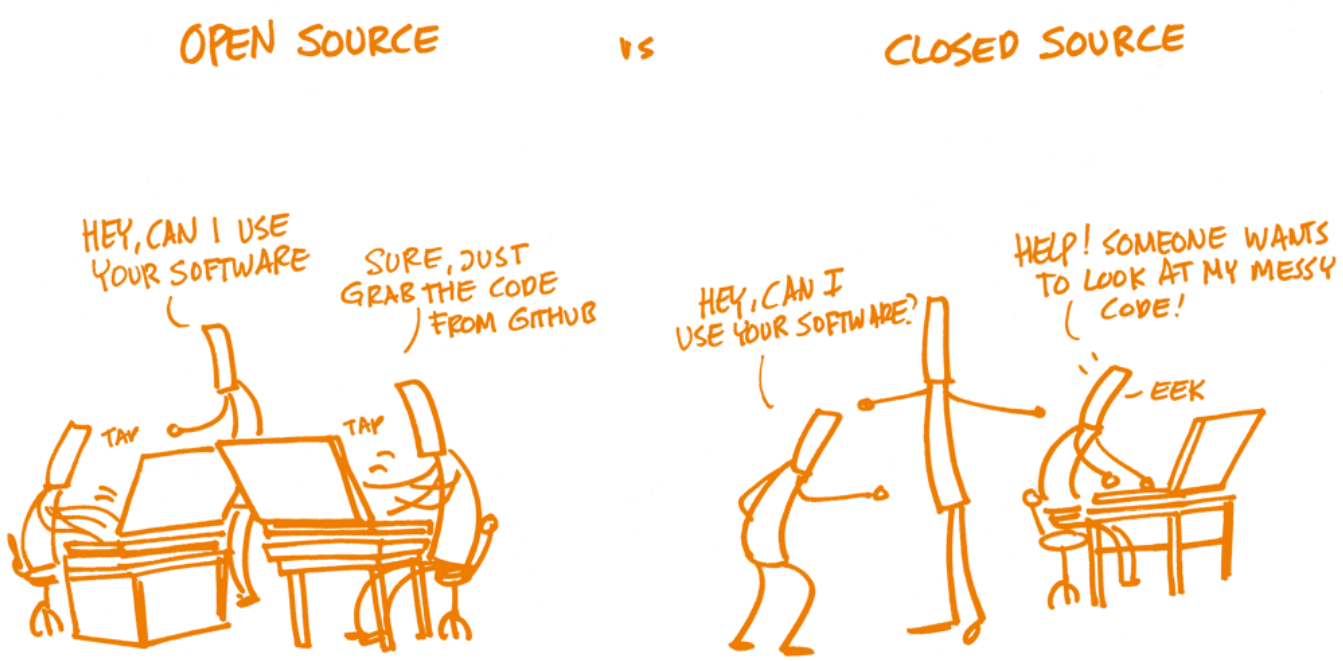
Tobias Schlauch <Tobias.Schlauch@DLR.de>
Institute for Software Technology
German Aerospace Center (DLR)
http://www.dlr.de/sc

Motivation

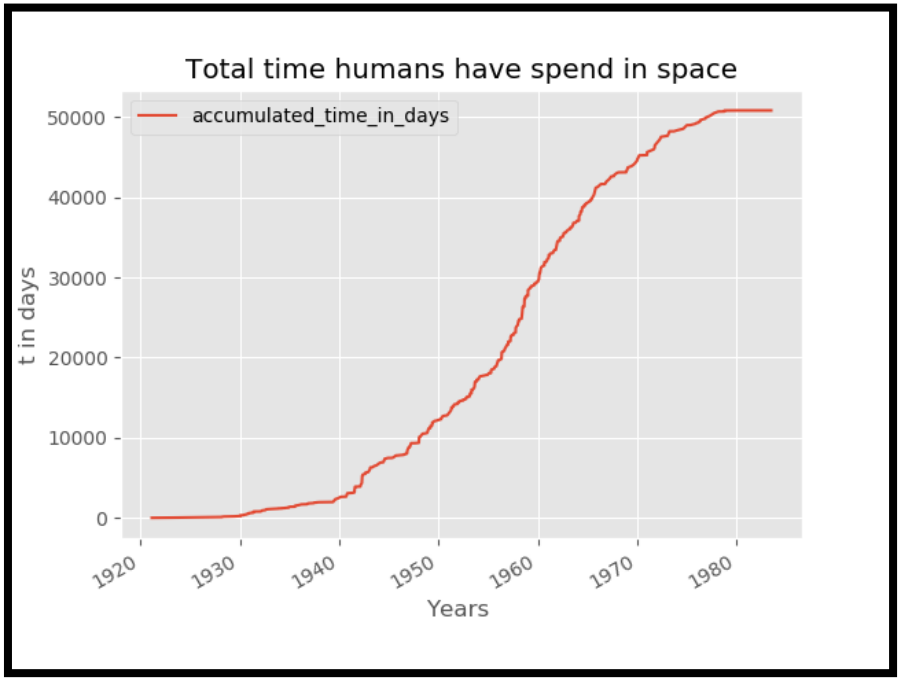
- Reproducibility of scientific results and enabling others to review them are vital in research.
- Making scientific results reproducible is **context-specific** and can be **hard in the concrete case**.
- Involved research software introduces a separate domain of potential reproducibility issues.

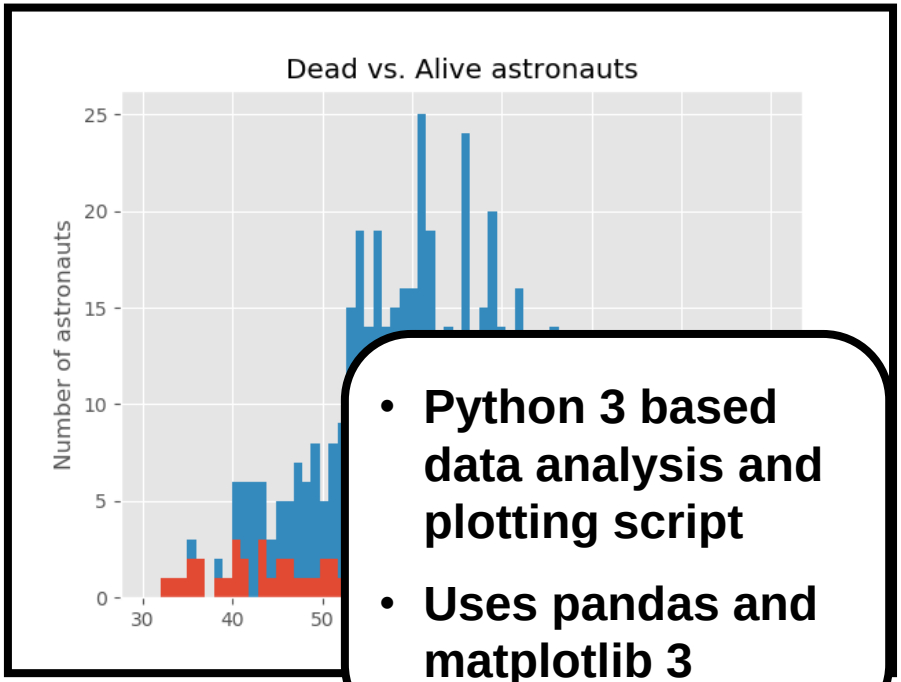
→ In the following we consider **minimum practices** when publishing research code to improve reproducibility of scientific results.

Making research code openly available helps. But is there more to consider?



Example: Astronaut Analysis





- **Python 3 based data analysis and plotting script**
- **Uses pandas and matplotlib 3**

How should I publish my analysis code to enable others to reproduce or check my results?

1. Put your code under version control
2. Make sure that your code is in a sharable state
3. Add essential documentation
4. Add a license file
5. Make your code citable
6. Release your code

1. Put your code under version control

HIFIS

Key points

- Version control helps you to prepare the code for sharing.
- Make sure to put all relevant artifacts into the repository.
- **.gitignore** helps you to specify things that you do not want to share.

2. Make sure that your code is in a sharable state

HIFIS

Key points

- Make sure that others can (re-)**use** your code
- Do not share internals and secrets with your code
- Strive for understandable and idiomatic code
- Think about introducing test automation

Changes:

- **PEP8 code style**
- **Code comments**
- **Relative file paths**
- **Functions**
- **Better directory structure**

3. Add essential documentation

Key points

- Provide documentation for relevant target groups
- Add a README file as a minimum documentation artifact to your repository

4. Add a license

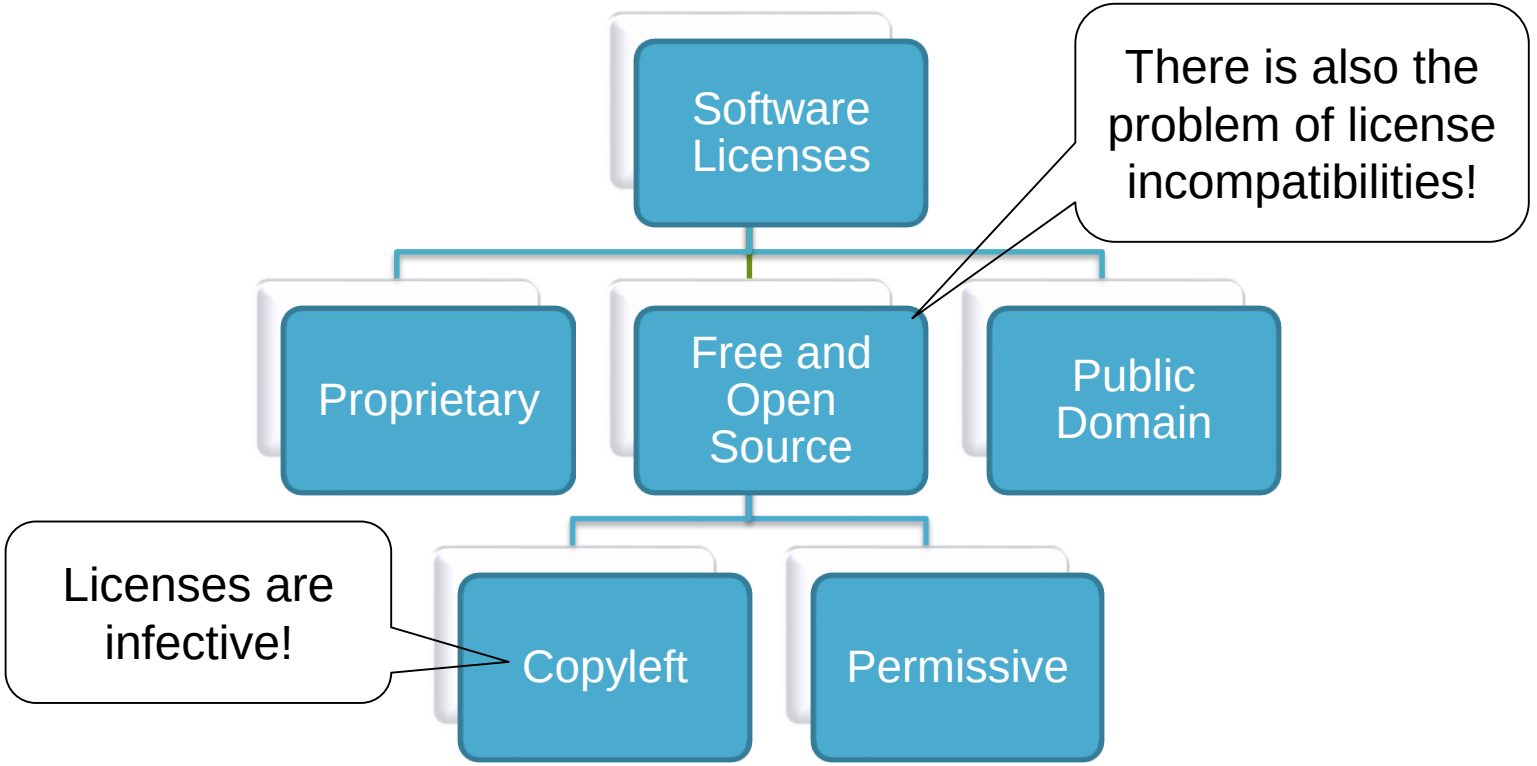
- Excursus on (open source) licenses…

Copyright

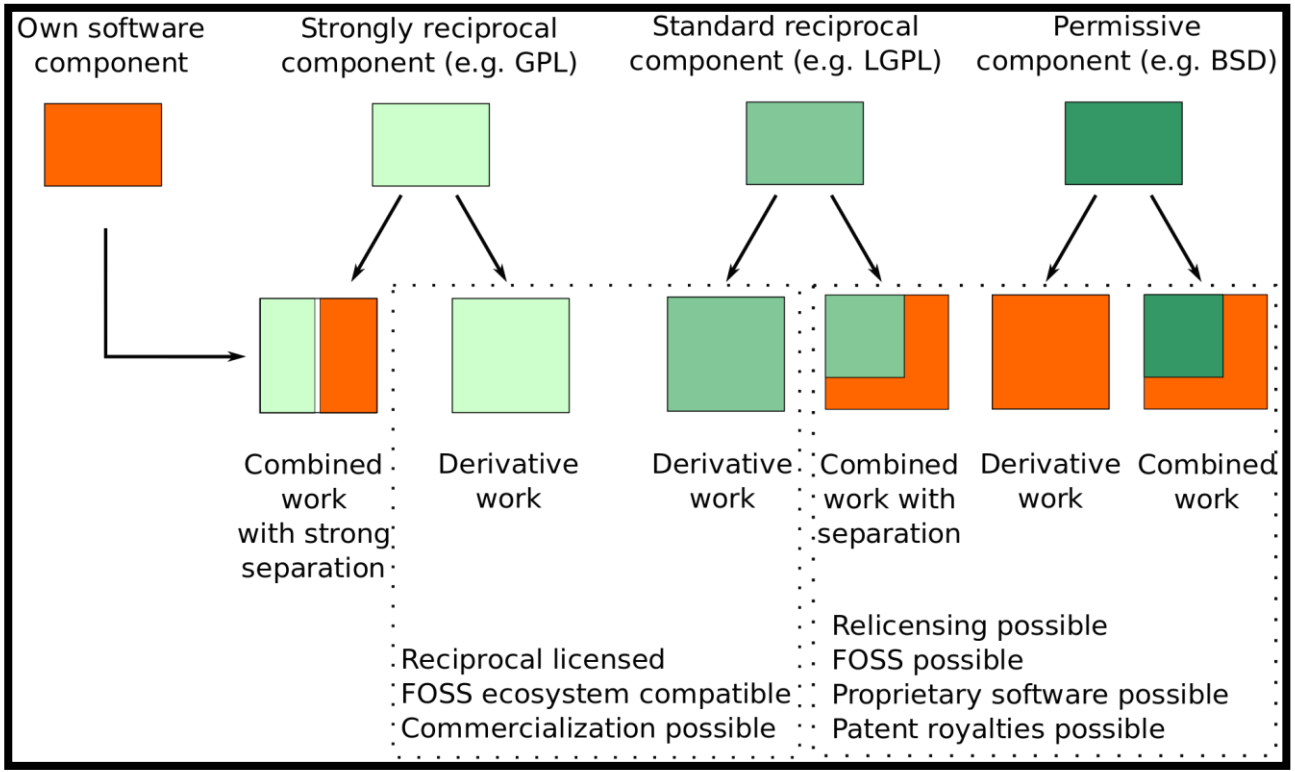
- Copyright:
  - protects the expression of an idea
  - grants exclusive rights to the copyright holder

- Everyone contributing to a software is considered a copyright holder of the software
  - All contributors are considered as copyright holders
  - A company paying an employed developer obtains most of the exclusive rights

Software Licenses



Combining Modules under Different Licenses



Take care when combining code under different licenses!

Minimal License Checklist

1. **Choose a license**
   - Analyze / comply with licenses of third-party dependencies
2. **Ask your boss for permission**
3. **Prepare your code**
   - Minimum: Add a license file and state the copyright holder(s)
   - Recommended: Follow REUSE Specification

Ask for legal advice if you are unsure!

Add the License File

Adapt the copyright hint

```
MIT License Copyright (c) 2013 German Aerospace Center (DLR)

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is furnished
to do so, subject to the following conditions:

The above copyright notice and this permission notice (including the next
paragraph) shall be included in all copies or substantial portions of the
Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS
OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF
OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```

Do not change the license text!

Document the Licenses of your Dependencies

Install

The script requires Python >= 3.6 and uses the libraries pandas (*BSD 3-Clause License*) as well as matplotlib (*Matplotlib License*).

Please clone this repository and install the required dependencies. We suggest that you install them in a virtual environment.

> The script has been used on CentOS 7 and Windows 7 with Python 3.6.6.

requirements.txt 32 Bytes

```
1  matplotlib==3.0.0
2  pandas==0.23.4
```

Key Points

- **Minimum:** Add a license file and state the copyright holder(s)
- **Recommended:** Follow REUSE Specification
- Consider third-party licenses from the very beginning

5. Make your code citable

- Excursus in software citation…

How to cite Software?
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

- **Authors**
- **Software Name**
- **Software itself**
- **Source code version**
- **Exact version**
- **Publication date**
- **PID**

How to make your Software citable?

- **Basic steps:**
  - Provide citation metadata
  - Obtain a persistent identifier
  - Add a citation hint
- **Practical approaches:**
  - Archive release in an (open) digital repository (e.g., Zenodo)
  - Add machine-readable citation metadata and archive via Software Heritage

Example 1: Archive via Zenodo

Installation

- Install Python >= 3.6
- Install the dependencies via: `pip install -r requirements.txt`

Run the Jupyter notebook(s)

- Start Jupyter: `jupyter notebook`
- Your browser should start and you can select the corresponding analysis notebook.

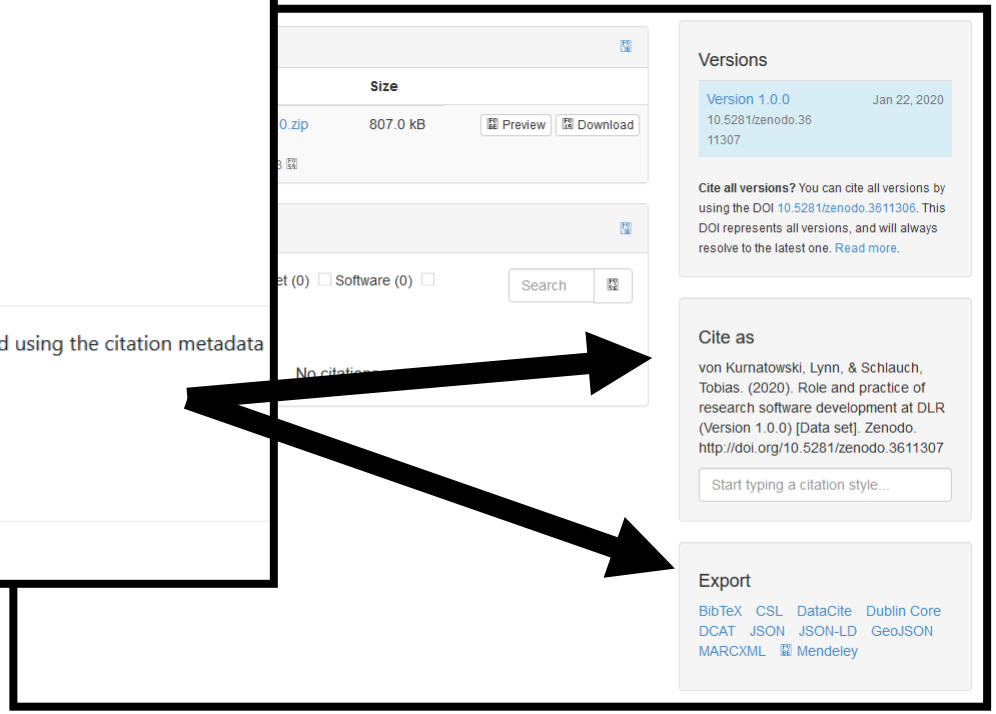
Citation

If you use this work in a research publication, please cite the specific version that you used using the citation metadata Zenodo DOI 10.5281/zenodo.3611306.

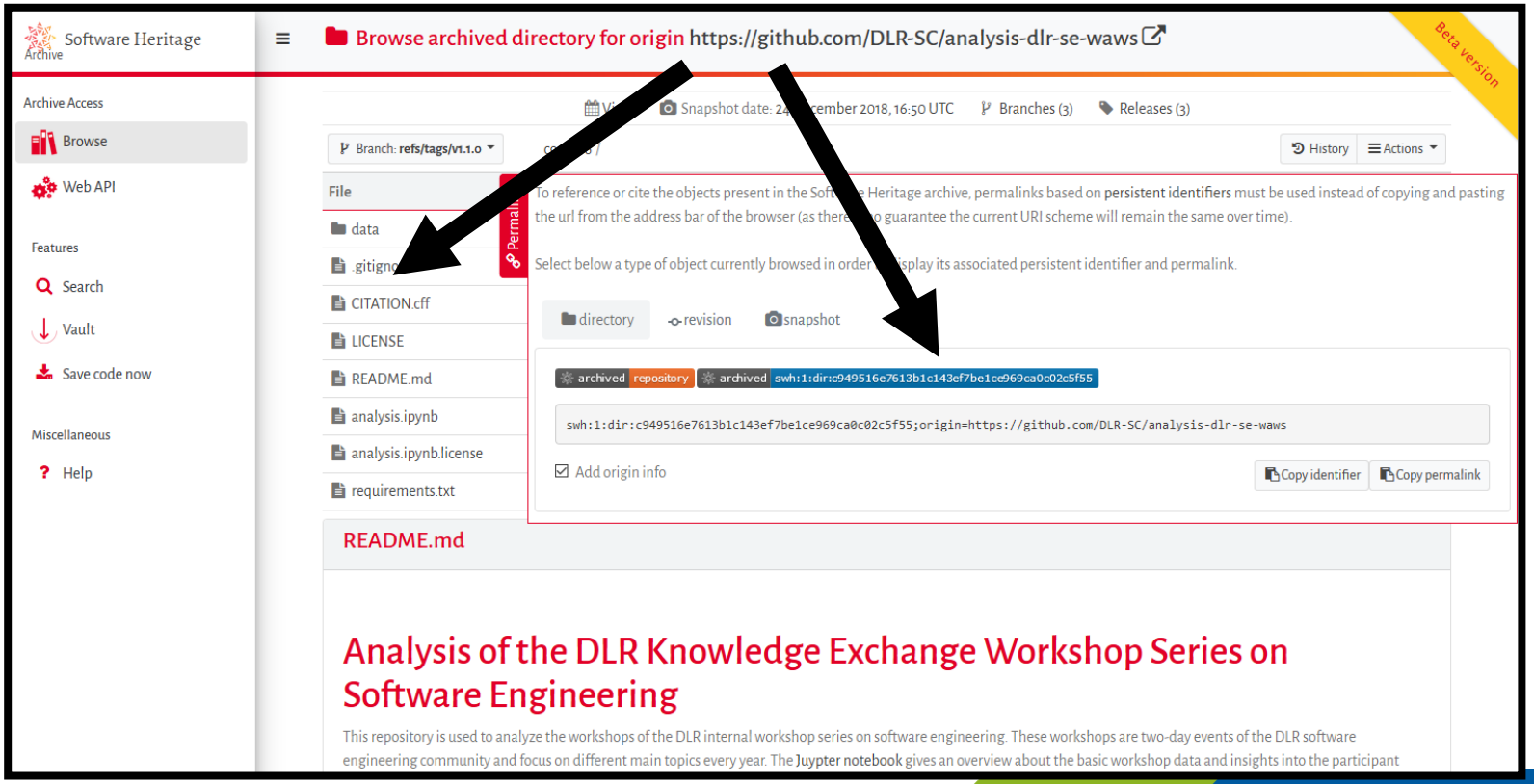
You can find an overview about the different versions of the material in the changelog.

License

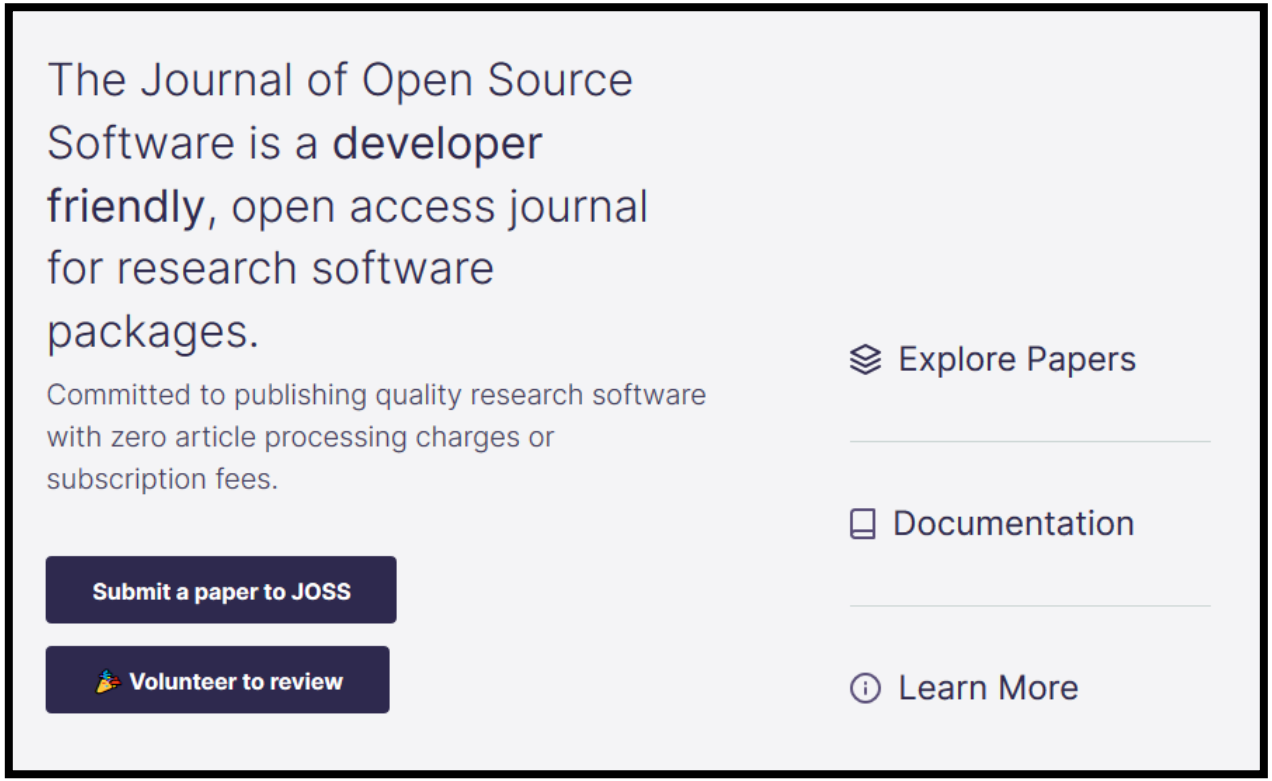
Copyright © 2020 German Aerospace Center (DLR)



Example 2: Archive via Software Heritage



Increase Chances to receive Credit? Publish in a Software Journal



Recommended options:

- Journal of Open Source Software (JOSS)
- Journal of Open Research Software (JORS)

Key Points

- Cite all relevant software packages as good as possible in your academic work
- Provide relevant information and encourage citation

6. Release your code

- Excursus on release management…

Software Release Basics

- A **release** is a specific working software version.
- The **release number** uniquely identifies the release (e.g., 1.0.1 or 2022-03-17)
- A user uses the **release package** to install and use the released software:
  - Contains code + documentation
  - Simplest form: snapshot of your source code repository packaged as Zip file

Minimal Release Checklist

1. **Prepare your code for release**
   a) Define the release number
   b) Update the documentation and citation metadata
2. **Check your code**
3. **Publish and archive the release**
   a) Mark the release in the source code repository using a tag
   b) Create the release package and make it available
   c) Archive the release package

Key Points

- Mark used, working software versions using release numbers
- Document important changes in a changelog
- Archive the release package

Summary

1. Put your code under version control
2. Make sure that your code is in a sharable state
3. Add essential documentation
4. Add a license file
5. Make your code citable
6. Release your code

Image Credits

- All content is licensed under Attribution 4.0 International (CC BY 4.0) with the following exceptions:
  - HIFIS logos / slide layout, © Helmholtz Association of German Research Centres. All rights reserved.
  - Open Source vs. Closed Source, slide 3, image by Patrick Hochstenbach, CC0 1.0 Universal (CC0 1.0), source: https://raw.githubusercontent.com/OpenScienceMOOC/Module-5-Open-Research-Software-and-Open-Source/master/content_development/images/open_research_software_open_source.png
  - Copyright logo, slide 10, public domain, source: https://commons.wikimedia.org/wiki/File:Copyright.svg
  - License compatibility, slide 12, image by Mikko Välimäki, public domain, source: https://commons.wikimedia.org/wiki/File:Software-license-compatiblity-graph.svg
  - Philae landing on comet 67 P/Churyumov-Gerasimenko, slide 30: DLR, CC BY 3.0

Thank you!

Questions?

Tobias.Schlauch@dlr.de

www.DLR.de/sc | Twitter: @TobiasSchlauch

DLR