

# Good Practices for Documenting Copyright and License Information in your Software

Talk in the Seminar Series, 23.08.2022,  
Max Planck Institute for Meteorology, Hamburg

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



Knowledge for Tomorrow



# Research Software Development at DLR

## Some context

- About 10.000 employees, ~20% of DLR employees involved in software development
- Variety of fields, maturity, and technologies:  
<https://doi.org/10.1145/3387940.3392244>

## Brief history of DLR`s Software Engineering Initiative

- Activities started in 2005 as part of DLR`s quality assurance program
- Since 2017 focus moved more and more on research software development aspects
- Work is driven by the DLR institute for Software Technology and funded by DLR IT



# Copyright

- **Copyright**

- Software is protected by copyright.
- Copyright protects the expression of an idea.
- Copyright grants exclusive rights to the copyright holder.



# Copyright

- **Copyright**

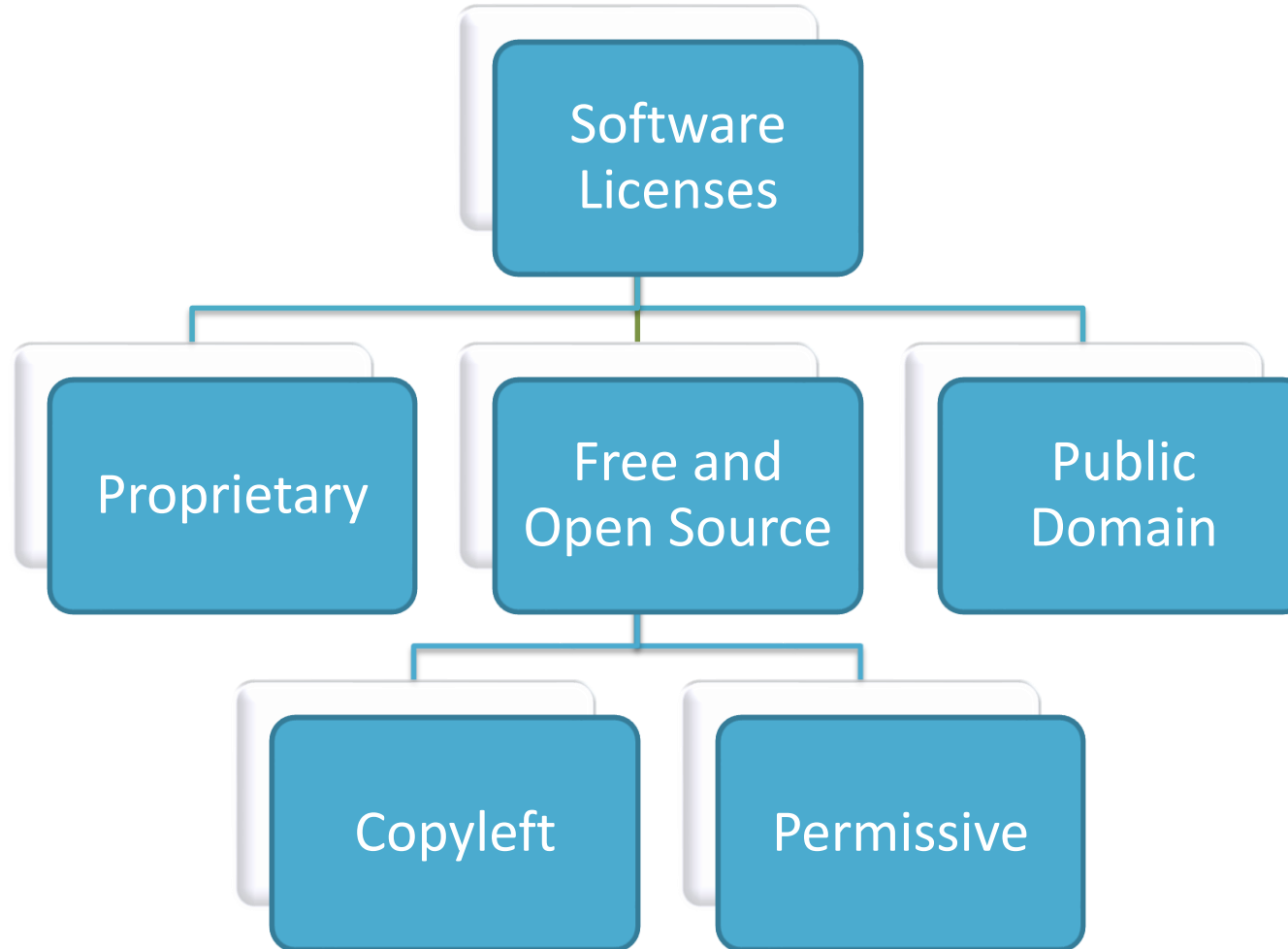
- Software is protected by copyright.
- Copyright protects the expression of an idea.
- Copyright grants exclusive rights to the copyright holder.

- **Who is the copyright holder of a software?**

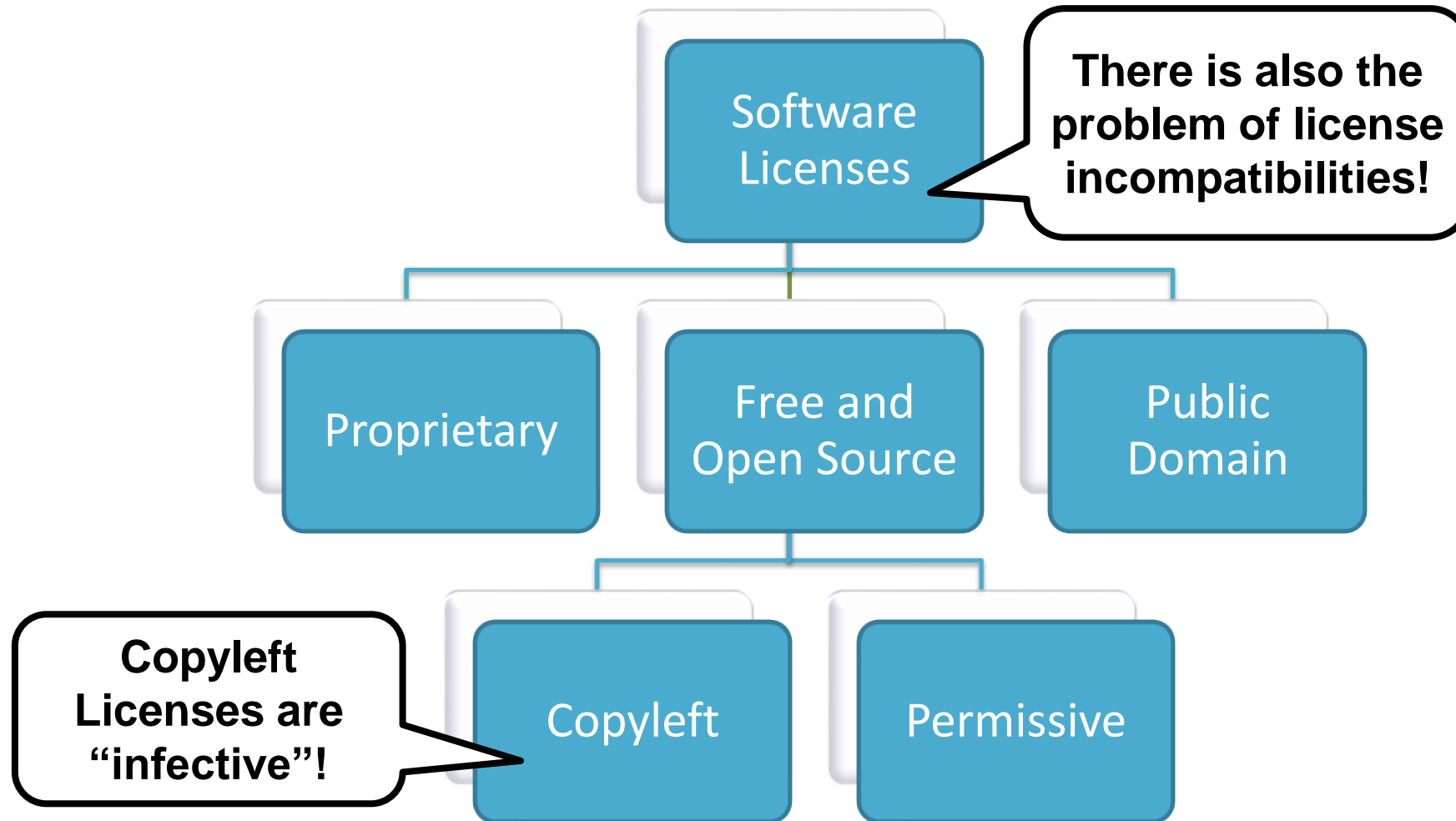
- All contributors are considered as copyright holders and jointly exercise the rights granted by copyright.
- A company paying an employed developer obtains most of the exclusive rights.



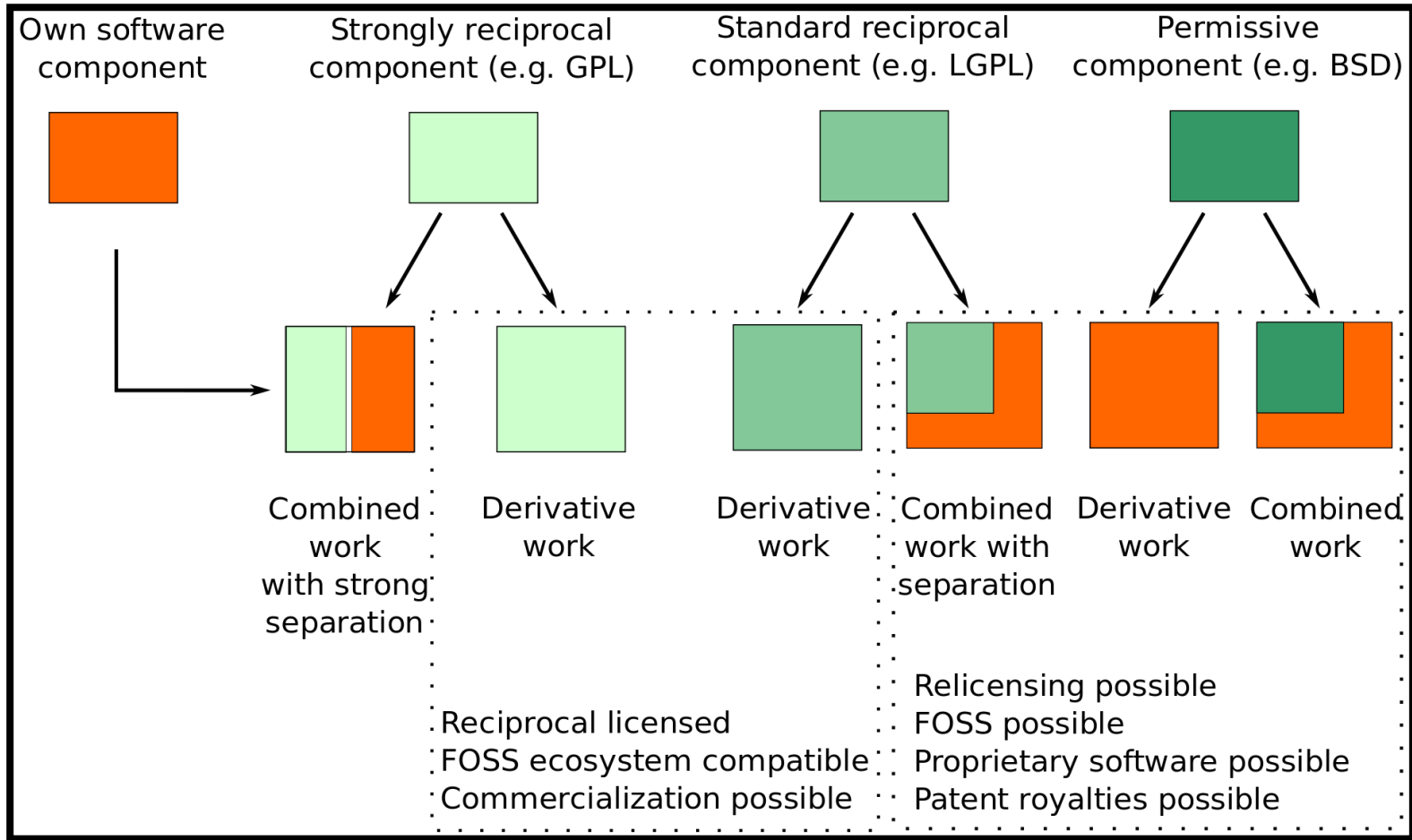
# Software Licenses



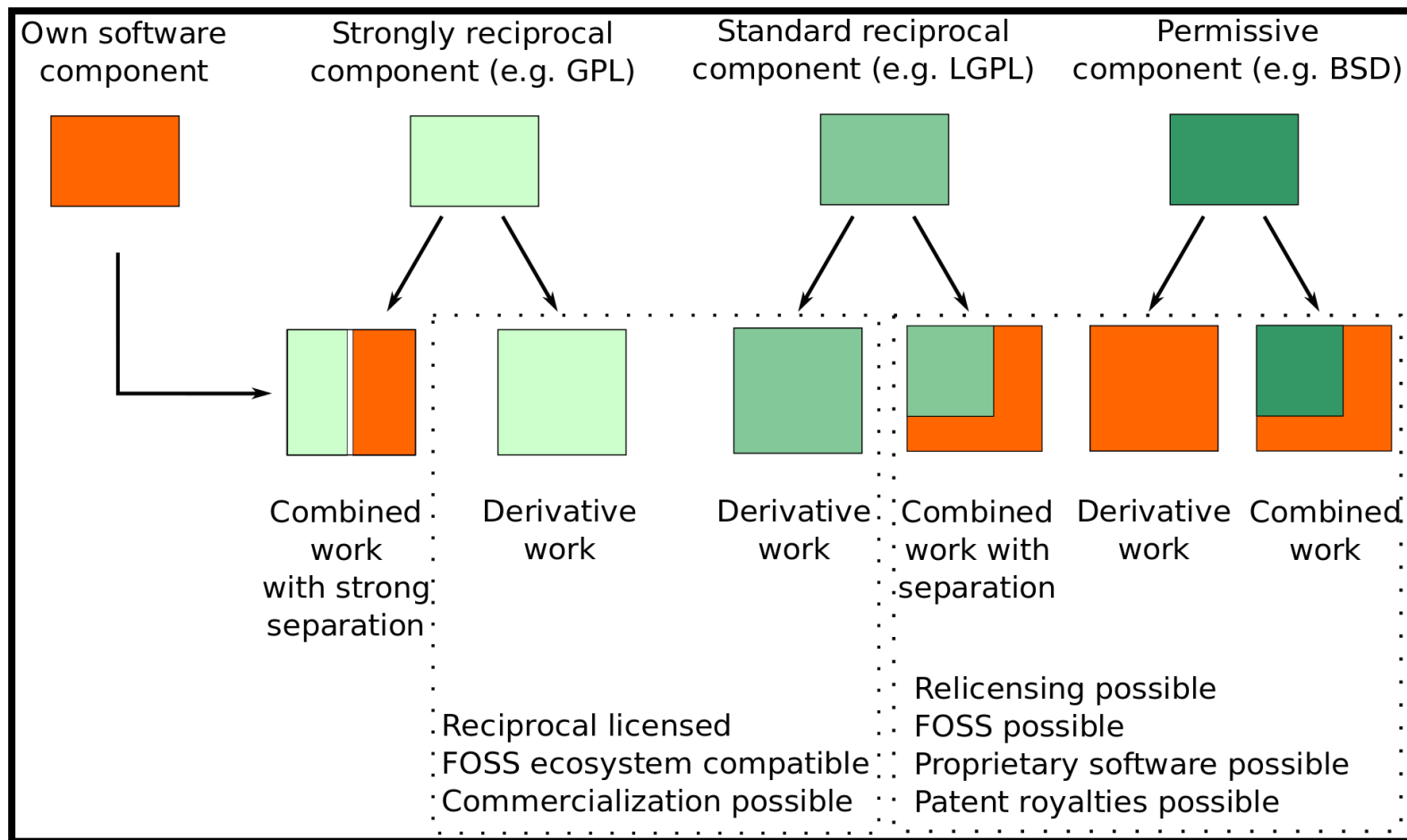
# Software Licenses



# Combining Modules under Different Software Licenses



# Combining Modules under Different Software Licenses



**Make sure that every code that you use is covered by a license!**

**Take care when combining code under different licenses!**





# Minimal License Checklist

## 1. Choose a license

- Consider strategical implications
- Comply with licenses of third-party dependencies

## 2. Ask your boss for permission

## 3. Add copyright holder and license information to your code repository

### • Assumptions:

- You created the software as part of your job.
- You created the software completely on your own.



# Minimal License Checklist

## 1. Choose a license

- Consider strategical implications
- Comply with licenses of third-party dependencies

## 2. Ask your boss for permission

## 3. Add copyright holder and license information to your code repository

### • Assumptions:

- You created the software as part of your job.
- You created the software completely on your own.

**Find out about  
your  
organizational  
processes!**

**Ask for legal  
advice if you are  
unsure!**



# How to best add Copyright Holder and Licensing Information?

- **Typical questions:**

- How and where do I indicate the copyright holders?
- How and where do I provide license information?
- Do I really need the long legal texts in my file headers?
- How to handle different copyright holders and licenses?
- And much more...



# How to best add Copyright Holder and Licensing Information?

- **Typical questions:**

- How and where do I indicate the copyright holders?
- How and where do I provide license information?
- Do I really need the long legal texts in my file headers?
- How to handle different copyright holders and licenses?
- And much more...

- **Let us try the REUSE Approach :)**

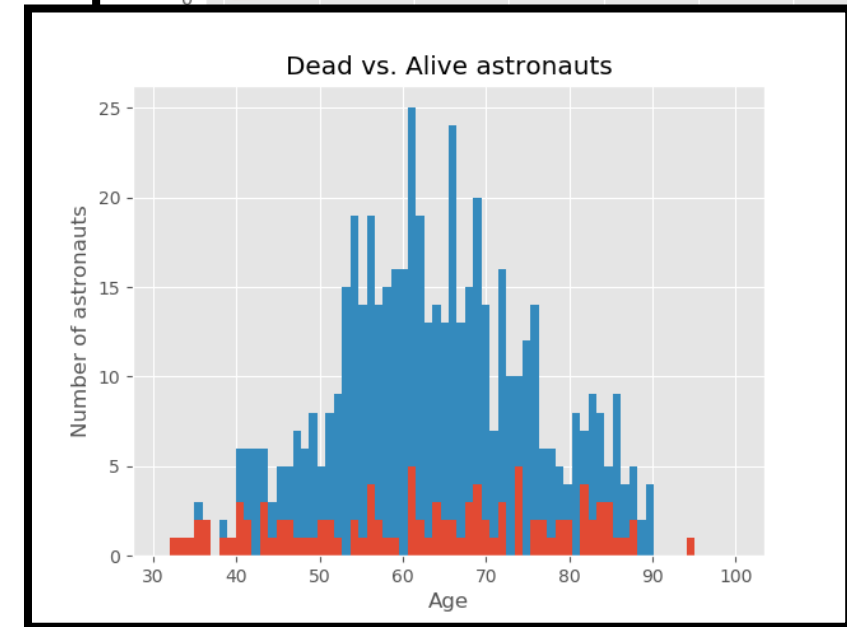
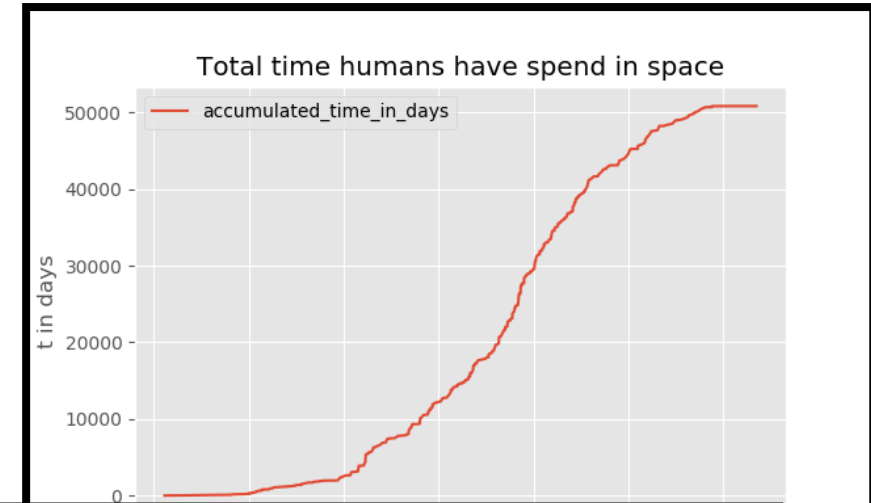
- **Goal:** Make it easy to determine license and copyright holders of a file for humans and machines!
- Heavily builds on SPDX: <https://spdx.dev/>
- For more information: [Tutorial](#), [FAQ](#), [Specification](#)



# REUSE in Action

## Step 0: Preparation

- **Astronaut Analysis:**
  - Data publication which we make ready for (open) publication
  - Code Repository: <https://gitlab.com/hifis/hifis-workshops/make-your-code-ready-for-publication/astronaut-analysis/-/tree/3-add-docs>
- **Copyright and license decisions:**
  - Copyright holder: German Aerospace Center
  - Source code: MIT
  - Data set: CC0-1.0
  - Documentation and plots: CC-BY-4.0
  - Insignificant files: CC0-1.0
- **Install the reuse tool:** <https://git.fsfe.org/reuse/tool>



# REUSE in Action

## Step 1: Add License Files

- Add all required license files to the `LICENSES` directory
- You can provide them manually or via `reuse download`:

```
$ reuse download MIT CC-BY-4.0 CC0-1.0  
Successfully downloaded LICENSES\MIT.txt.  
Successfully downloaded LICENSES\CC-BY-4.0.txt.  
Successfully downloaded LICENSES\CC0-1.0.txt.
```

- **Notes:**

- `reuse download` supports all licenses from the SPDX License List Data repository:  
<https://github.com/spdx/license-list-data>
- Other licenses can be added manually to the `LICENSES` directory
- `reuse download` can download all missing license files



# REUSE in Action

## Step 2: Add Copyright and Licensing Information

- Add copyright and licensing information via SPDX tags to all files in the code repository
- You can attach them manually or via `reuse addheader`:

```
$ reuse addheader \  
> --copyright="German Aerospace Center" \  
> --license="MIT" \  
> --year=2018 \  
> code/* .gitlab-ci.yml  
Successfully changed header of code\astronaut-analysis.py  
# ·SPDX-FileCopyrightText: ·2018 ·German ·Aerospace ·Center  
#  
# ·SPDX-License-Identifier: ·MIT
```

- **Notes:**

- `reuse addheader` handles comment styles and uncommentable files automatically
- You can have multiple copyright and license statements in each file
- You can use templates to provide additional text



# REUSE in Action

## Step 3: Check REUSE Software Compliance

- You can check whether everything is fine via `reuse lint`:

```
$ reuse lint
# SUMMARY

* Bad licenses:
* Deprecated licenses:
* Licenses without file extension:
* Missing licenses:
* Unused licenses:
* Used licenses: CC-BY-4.0, CC0-1.0, MIT
* Read errors: 0
* Files with copyright information: 12 / 12
* Files with license information: 12 / 12

Congratulations! Your project is compliant with version 3.0 of
the REUSE Specification :-)
```

- You can use the linter in pre-commit hooks or in your CI/CD workflow: <https://reuse.software/dev/>





# Summary

- Copyright protects software and grants exclusive rights to the copyright holder(s).
- Software licenses offer a “defined” way to grant rights to others but also require to fulfill some obligations:
  - Make sure that every code that you use is covered by a license!
  - Take care when combining code under different licenses!
- Software publication processes work differently in research organization:
  - Find out about your process!
  - Ask for legal advice if you are unsure with licensing and copyright aspects!
- When publishing software, REUSE helps you to attach copyright and licensing information properly.
  - Step 1: Add license files
  - Step 2: Add copyright and licensing information to all files
  - Step 3: Check REUSE compliance
- Currently, the Free Software Foundation puts a lot of effort in the REUSE project and its adoption!



# Copyright and License Information

- All content is licensed under [Attribution 4.0 International \(CC BY 4.0\)](https://creativecommons.org/licenses/by/4.0/) with the following exceptions:
  - DLR logo, slide layout, DLR locations map on slide 2, © German Aerospace Center. All rights reserved.
  - Copyright logo, slides 3/4, public domain, source: <https://commons.wikimedia.org/wiki/File:Copyright.svg>
  - License compatibility, slides 7/8, image by Mikko Välimäki, public domain, source: <https://commons.wikimedia.org/wiki/File:Software-license-compatibility-graph.svg>
  - Philae landing on comet 67 P/Churyumov-Gerasimenko, slide 19: German Aerospace Center, [CC BY 3.0](https://creativecommons.org/licenses/by/3.0/)



# Thank you!

Questions?

[Tobias.Schlauch@dlr.de](mailto:Tobias.Schlauch@dlr.de)

[www.DLR.de/sc](http://www.DLR.de/sc) | [@TobiasSchlauch](https://twitter.com/TobiasSchlauch)