

Developer best practices for expressing license and copyright information in Free Software projects (Draft)

Jonas Öberg, Matthias Kirschner

Contents

1	Background	3
1.1	Best practices	3
1.1.1	Provide the exact text of each license used	3
1.1.2	Include a copyright notice and license in each file	4
1.1.3	Provide an inventory for included software	6
1.2	Further recommendations	7
1.3	Future development	7

1 Background

The FSFE has launched a project to develop and raise awareness of best practices for the expression of license and copyright information in free and open source projects. The project is intended to facilitate management of source code by making licensing and copyright information more consistent in how it is added to source code in ways which allow for automating many of the processes involved.

This is a concern for any free and open source software developer, user or distributor. Incorrect license and copyright information can lead to situations where software developers are not credited for the work they have done, where the license or copyright information is misunderstood or where using a software project becomes a burdensome process due to the inability to use automated tools.

The best practices we've identified and worked to describe in this text are based on existing sources for best practices, but we have identified and revised the practices which can facilitate automation.

We've also sought to identify practices which will be easy for software developers to adopt and will work more on this to develop material suitable for anyone wishing to help us make copyrights and licensing information computer readable.

1.1 Best practices

1.1.1 Provide the exact text of each license used

Free and open source software licenses are standardised and have standard texts. Regardless of which license you use, you should include the license text in your project. You should also include the license text of any code which may be under a different license, and it's important you do not change the license text on other software you include.

If the license you use is included in the SPDX License List¹, you should download and copy the text representation of the license directly from the SPDX repository². Doing so ensures that unless you modify the license text, the checksum of the license file will be identical across all projects using the same license.

¹<https://spdx.org/licenses/>

²<https://github.com/spdx/license-list>

If your project only includes code licensed under a single license, you may provide the text of this license in a file in the top level directory of your repository with the name “LICENSE”.

Since many projects include code under different licenses, it’s often not feasible to include all licenses in the top level, in which case you should create a directory at the top level called “LICENSES” within which you include the license text of each license used.

You must include all licenses which are used in your project, and you must never change any license texts even if they are very similar to existing ones. Some licenses, like the BSD 2 Clause license, must be adapted to include the name of the copyright holder in the license text. Your project may end up including multiple versions of the same BSD 2 Clause license because some parts may be written by Alice and others by Bob, resulting in two different license files, even if the only difference is the copyright holder.

This is how your source code tree can end up looking regarding the license files included:

```
LICENSES/  
    GPL-2.0.txt  
    BSD-2-Clause_Alice.txt  
    BSD-2-Clause_Bob.txt
```

Keep in mind:

- Don’t change any license texts, use the verbatim form of the license text
- Don’t remove any license texts, include the license texts of all software
- Keep the filename of the licenses consistent, so you can refer to it from the source code files (see the following practice)

1.1.2 Include a copyright notice and license in each file

You should ensure all files in your project have a header file, and that all header files have the same format. Even if your project has a header file which looks different from other projects, it helps to have a consistent style to the header.

Source code files are often reused across multiple projects, taken from their origin and repurposed, or otherwise end up in repositories where they are separate from its origin. Each file should, therefore, contain enough information in itself to convey copyright information and it’s important you do not remove information in existing copyright headers.

You may record information about authorship by relying on the underlying version control system you’re using. If you do, there’s no need to include copyright information in each file, but you must take care accurate information can be found through the version control system, and provide a link back to that location.

If you do use a version control system to carry information about copyright, you must take care to make sure correct information is retained when project maintainers commit code contributed to the project on behalf of others. You must also make sure the version control system is publicly accessible and will remain so.

For a project using the version control system to convey information about copyright, it is recommended each commit message include a copyright notice and that this copyright notice is enforced through a pre-commit hook or similar:

```
Fixing bugs #1 and #2.  
Copyright: (c) 2017 Alice Commit <alice@example.com>
```

An appropriate header would be:

```
/*  
 * This file is part of project X. It's copyrighted by the contributors  
 * recorded in the version control history of the file, available from  
 * its original location http://git.example.com/X/filename.c  
 *  
 * SPDX-License-Identifier: BSD-2-Clause  
 * License-Filename: LICENSE/BSD-2-Clause_Charlie.txt  
*/
```

If the project is not using the version control system to convey copyright information, the same copyright information should be included in the source code file. Copyright notices should have a consistent format and be sorted by year.

```
/*  
 * Copyright (c) 2017 Alice Commit <alice@example.com>  
 * Copyright (c) 2009-2016 Bob Denver <bob@example.com>  
 * Copyright (c) 2007 Charlie Example <charlie@example.com>  
 *  
 * SPDX-License-Identifier: BSD-2-Clause  
 * License-Filename: LICENSE/BSD-2-Clause_Charlie.txt  
*/
```

The “License-Filename” tag shall be a persistent URL or a filename in the repository where the actual license text is available. This is more accurate than the SPDX license identifier and ensures the full license text is always referenced from the individual source file. The tag can be repeated if multiple license files are relevant.

You should include information about your project’s practices in the README or similar file.

If your project includes binaries or source code files in which comments can not be placed, you should provide a separate file, one for each file, through which you provide a link to the license file. If your project includes a binary file “mydata.jpg”, it should then also contain the text file “mydata.jpg.license” which includes a copyright header according to

the format you customarily use for headers.

If you wish to be explicit about the license of an output file, which does not exist in the repository but which will be created at build time, you may include the license file without the corresponding binary file.

Keep in mind:

- Use a consistent style of your headers throughout the project
- Don't remove existing headers, but only add to them
- Do consider using version control systems to keep a record of copyright holders
- Do keep your version control system public if you use it
- Make references to the license text and the SPDX identifier from each source code file
- Include license and copyright information also for files which can not include a proper header by including this information in a separate file with the ending ".license"

1.1.3 Provide an inventory for included software

Aside from the license files included in the project, and the file level copyright information, you may include a bill of material for your project, but you should only do so if this is generated automatically.

A bill of material can be very complicated and lengthy, making it difficult to maintain. If you do not generate it automatically, it's very likely someone will forget to update it when making changes. In these cases, it's best not to have a bill of material, but to rely only on the information coupled with the source code files.

If you do have a way of automatically creating a bill of material, and if you choose to do so, you should generate it automatically from the most reliable information you have about each file in your project. This includes copyright information kept in version control systems and licenses on files which can not include a standard header, or which includes the header in a separate license file.

You may also choose to include in the bill of material your output files generated when compiling the project, such that you can signal through the bill of material, which license is relevant for the output files depending on the included source code.

The bill of material should be conformant to the SPDX specification and included in a file in the top level directory of your repository called "LICENSE.spdx".

Keep in mind:

- Don't create a bill of material if you can't generate it automatically
- If you generate one automatically, it's helpful to include one
- Make your bill of material conformant to the SPDX specification

1.2 Further recommendations

Aside from following the best practices above, we recommend that as a next step, you consider using ScanCode or FOSSology on your project, to verify that these tools can read and understand the licenses of your software.

1.3 Future development

These best practices hint at the need for license stewards to make available authoritative text versions of their licenses including their published checksum. When and if this happens, we would recommend using the license stewards' version, rather than the one from the SPDX repository.

While we have worked on some practices to include copyright information in the version control system, more practical experience from this is needed by projects to ascertain the best practices about it.

We also see a need to further develop the tools needed by projects to maintain and keep consistent copyright and license information. We have tried to keep relevant license information as close to the original files as possible, in ways that make it as easy as possible to manage them. Still, we believe a greater amount of automation would be beneficial even on a project level, for instance by having an appropriate “lint” checker for copyright information as part of the build process.