

Open Source License Obligations Checklists

A project to create and disseminate generally accepted rules to fulfill the obligations when distributing software that is licensed under commonly used Open Source licenses

by Carsten Emde¹ and Till Jaeger²

Version 5 of September 3, 2017

Rationale

Whenever Open Source software is copied and distributed which typically is permitted by every type of Open Source license, a number of obligations and prohibitions are imposed on the distributor. As a very common situation of such software, the receivers of the software will recursively redistribute it in such a way that a chain of distributors and receivers is created – all of them having to fulfill the same license obligations. However, for the time being, there is no common understanding how these obligations are to be fulfilled in detail which regularly leads to misunderstandings, conflicts and sometimes even to court cases. A solution would be to create generally accepted checklists of the obligations of commonly used Open Source software that are accepted by distributors and copyright holders and trusted by all members of the distribution chain.

Goal

The goal of this project is to create checklists for the most frequently used and the most important Open Source licenses and to provide assistance tools for the determination of differences between them. The checklists shall be agreed upon by a sufficiently large number of lawyers and other legal specialists from different legislations. In addition, the checklists should be disseminated in such a way that authors and users of particular license texts are also reached and given the opportunity to confirm or to modify the checklists. The checklists shall be made available not only in text form for reading by humans but also in digital form in a defined data exchange format. The latter is intended, for example, to give scanner manufacturers the option to provide an interface for electronically reading the checklists and supplying reliable information on license obligations along with detected licenses as an add-on to their software. A Rights Expression Language (REL) such as for example the Open Digital Rights Language (ODRL) (<https://www.w3.org/community/odrl/>) could be used for this purpose.

1 Open Source Automation Development Lab (OSADL) eG, Heidelberg, Germany

2 JBB Rechtsanwälte, Berlin, Germany

Procedures

To reach the goal of creating checklists for a number of different Open Source licenses, several separate steps are needed. First of all, a *vocabulary* must be created to use the same wording among different licenses. In a second preparatory step, a *license describing language* must be created to translate the wording of the licenses into a common set of instructions and to mark unclear provisions. This then creates the prerequisite for interested laymen, lawyers and other legal specialists who may *compile* every atomic content (permission, prohibition, condition, use cases etc.) of every license into the license describing language. This can best be developed and reviewed using a mailing list and a related Internet-based repository that are restricted to the participants of the project.

In cases where the wording of the license obligation leaves some room for interpretation, an attempt could be made to find an agreement among lawyers and other legal specialists. Such agreements or, if impossible to reach, at least descriptions of such situations, represent an additional part of the project – particularly in cases where a license clause may be interpreted either literally or constructed in the spirit of the license. GPL-2.0, for example, contains the license obligation to mark modified source code files: "You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change." If literally followed, a header line such as "This file was changed on August 23, 2017." would need to be added on top of modified files which, after some continuous development, would become difficult to handle because of the many useless header lines. Today's practice would translate the above given license obligation into: "You must provide a machine-readable context diff of modified files along with the date of the change and the name and the email address of the responsible author." It is no coincidence that this information typically is provided by state-of-the-art software revision and release systems such as *git*. But although this way to fulfill the obligation is much more comprehensive and very probably better represents the spirit of the license, it does not literally follow the text and, thus, may constitute a copyright infringement – at least in the eyes of a formally arguing rightholder. However, whether interpretation topics such as this one should already be included in the first phase of the project or better in a later one remains to be discussed.

Finally, possible additional parts of the project may be to semantically compare licenses to each other in order to establish *differences* between them and to develop a concept to describe license *compatibility*.

Elements

1 Vocabulary

License terms may be described using different wording in different license texts such as "limitation of liability"/"disclaimer of warranty". A list of all relevant license terms must be created, grouped into those with identical and those with different meaning, and a common term be introduced for those with identical meaning. The same applies to actions that are imposed or prohibited by a license.

2 License describing language

On the base level, a license contains obligations (“YOU MUST”) and prohibitions (“YOU MUST NOT”). In order to allow for the extraction of license obligation and prohibition in atomic granularity, “YOU MUST” and “YOU MUST NOT” atoms may be repeated as often as required. The atoms are understood as logical “and” elements, i.e. all of them must be followed to reach license compliance. The arguments of the atoms, i.e. what must or what must not be done, are described with words that are selected from a vocabulary. This vocabulary along with a distinct description of every term and action used to describe the atoms will be created and reviewed in the course of the project.

Sometimes the license obligation atoms may allow the distributor to freely select between a number of optional use cases; therefore the “USE CASE” element is introduced which, in turn, may again be followed by “YOU MUST” and “YOU MUST NOT” atoms depending on the selected option. In addition, the license may contain certain obligations that only are applicable, in case a particular unalterable condition is present; to encode such a situation the “YOU MUST” and “YOU MUST NOT” atoms may be preceded by the “IF” element along with a condition. Finally, a term may have an “ATTRIBUTE”.

Examples

To provide an idea of how checklists and the related tools may look like the following examples of parts of a GPL-2.0 and GPL-3.0 checklist are given. Please note that these examples only serve to elucidate the principle and certainly are far from complete. In addition, project participants will need to discuss how far this first version will go, e.g. the inclusion of patent-related license obligations may be too difficult and better be left for a future version of the checklists.

(a) Vocabulary

The following terms and actions are used in this example. Here only the terms and actions are given, but the final project manual will contain a dictionary with a description of every term and action as well as a list of synonymous expressions, if applicable.

Terms

Copyright notice, License text, Warranty disclaimer, Source code, Customary medium, Machine-readable, Interactive, License announcement, Patent holder, Patent license, Source code modification, Copyright license, Original license, File name, Change date, Binary-only delivery, Installed, Tool chain information, Installation scripts, Installation material, Delayed source code delivery, Written offer, Duration, No profit, No charges, Usage

Actions

Forward, Provide, Display, Grant, Use, Permit, Restrict

(b) GPL-2.0 checklist (provisional and incomplete)

YOU MUST *Forward* **Copyright notices**

YOU MUST *Forward* **License text**

YOU MUST *Forward* **Warranty disclaimer**

YOU MUST *Provide* **Source code**

ATTRIBUTE **Customary medium**

ATTRIBUTE **Machine-readable**

IF **Interactive**

YOU MUST *Display* **License announcement**

IF **Patent holder**

YOU MUST *Grant* **Patent license**

IF **Source code modification**

YOU MUST *Grant* **Copyright license**

YOU MUST *Use* **Original license**

YOU MUST *Provide* **File name**

YOU MUST *Provide* **Change date**

USE CASE **Binary-only delivery**

YOU MUST *Provide* **Tool chain information**

USE CASE **Delayed source code delivery**

YOU MUST *Provide* **Written offer**

ATTRIBUTE **Duration** 3 years

ATTRIBUTE **No profit**

USE CASE **Installed**

YOU MUST *Provide* **Installation scripts**

YOU MUST *Provide* **Installation material**

YOU MUST *Permit* **Installation**

YOU MUST NOT *Restrict* **Usage**

(c) GPL-3.0 checklist (provisional and incomplete)

YOU MUST *Forward* **Copyright notices**

YOU MUST *Forward* **License text**

YOU MUST *Forward* **Warranty disclaimer**

YOU MUST *Provide* **Source code**

ATTRIBUTE **Customary medium**

ATTRIBUTE **Machine-readable**

IF **Interactive**

YOU MUST *Display* **License announcement**

IF **Patent holder**

YOU MUST *Grant* **Patent license**

IF **Source code modification**

YOU MUST *Grant* **Copyright license**

YOU MUST *Use* **Original license**

YOU MUST *Provide* **File name**

YOU MUST *Provide* **Change date**

USE CASE **Binary-only delivery**

YOU MUST *Provide* **Tool chain information**

USE CASE **Delayed source code delivery**

YOU MUST *Provide* **Written offer**

ATTRIBUTE **Duration** At least 3 years

ATTRIBUTE **Duration** As long as product is supported

ATTRIBUTE **No profit**

USE CASE **Via Internet**

ATTRIBUTE **No charges**

USE CASE **Installed**

YOU MUST *Provide* **Installation scripts**

YOU MUST *Provide* **Installation material**

YOU MUST *Permit* **Installation**

YOU MUST NOT *Restrict* **Usage**

(d) Example of an analysis of differences (using above GPL-2.0 and GPL-3.0)

The following example analysis of differences is not intended as an automated test that can be carried out by a layman, but as kind of a “computer-aided license difference evaluation” to assist lawyers and legal specialists when doing their professional work. Please also note that the following two examples are incomplete and only intended to describe the principle. Tables of differences between licenses are intended to facilitate the quality assessment of software products in such a way that the license obligations of similar licenses, for example, may be grouped together and a common set of obligations used for them.

Difference	GPL-2.0	GPL-3.0
#1	YOU MUST Provide Written offer ATTRIBUTE Duration 3 years	YOU MUST Provide Written offer ATTRIBUTE Duration At least 3 years ATTRIBUTE Duration As long as product is supported
#2	YOU MUST Provide Written offer	YOU MUST Provide Written offer USE CASE Via Internet ATTRIBUTE No charges

(e) License compatibility

In addition to the tables of license differences, tables of license compatibility shall be created. Main topics such as licenses with copyleft and those without copyleft will be used for this purpose and extended by explicitly ruled compatibility information derived from particular licenses. Finally, often used combinations of licenses will be evaluated individually.

3 Compilation

After the license describing language will be established and successfully tested on a number of licenses, the selected project licenses will be compiled by an international group of experts using distributed development via mailing lists and Internet-based repository.

4 License differences and license compatibility

If the primary goal of the project can be reached and approved checklists of the selected licenses are available, these data may be used to create tables of license differences. Further work may be necessary to develop a two-dimensional table of the investigated licenses to establish the compatibility between pairs of them as probable, improbable or unsure.

Collaboration, licensing and liability

The initial steps of preparing and discussing the project already were done in close collaboration with the OpenChain initiative of the Linux Foundation. The OpenChain initiative offered support to disseminate

the project description, to recruit volunteers and contributors, to supervise the progress of the project and to distribute the resulting documents that will be licensed under CC-0. Although the checklists and the related tools will be prepared and provided with maximum diligence and care, any liability with respect to recommendations which license obligations must be fulfilled and, more importantly, which license obligations do not need to be fulfilled is rejected as far as permissible by applicable law. To make clear that the checklists are not intended as static but are seen as work under progress, the checklists will contain a user section where individual items may be added to the lists or existing items removed from them. Lawyers may use this feature to adapt the checklists to the needs of a particular situation they are responsible for.

Project management and funding

A number of various parties (machine builders, software manufacturers etc.) have signaled interest to contribute to the project with funding and/or personnel. Lawyers and other legal specialists have confirmed to act as service providers. At least one scanner manufacturer is willing to contribute and test the digital version of the checklists and its data interface.

Duration of the project is estimated to take about six month and can be started in September 2017 or at any later date when the funding threshold of the equivalent of 20,000 euros is reached. Companies who would like to join in and contribute to the project are asked to sign a letter of intent where they state the amount of funds or personnel they are willing to contribute. Funding by providing personnel is based on a per-hour rate of 100 euros. Thus, the starting threshold of 20,000 euros will, for example, be reached when a total of 10,000 euros of funds and 100 work hours of personnel are committed. The example 100 hours of personnel would be approximately 10% of the normal work time during a six-month period, i.e. four hours per week. Interested parties may use the attached letter of intent, scan it and submit it to the Open Source Automation Development Lab (OSADL) via email to office@osadl.org.

Letter of Intent

The company

wishes to support the above described project “Open Source License Obligations Checklists” and herewith declares the intent to make a

- monetary contribution in the amount of _____ euros
- work hour contribution in the amount of _____ hours per week during the project duration from September 1, 2017 to March 31, 2018. The contributing person to be added to the mailing list will be

(Name)

(Email address)

(Please tick and complete at least one of the above or both)

- I agree that the resulting documents will be made available under the Common Creative CC-0 license.
(Must be ticked)

- Please send an overview about the results obtained so far at least once per month until the end of the project to the address:
(Tick and complete the email address below, if desired)

(Email address)

(Location)

(Date)

(Printed name of a signatory of the company)

(Signature)