

□□ Lizenzierung und Urheberrecht

Software, deren Code veröffentlicht wird, ist grundsätzlich per Default, auch ohne Lizenz urheberrechtlich geschützt und darf nicht kopiert, verändert oder verbreitet werden (§ 69a u. § 69c UrhG) . Erst durch eine Open-Source-Lizenzierung werden anderen diese Rechte gewährt. Damit es sich bei einem neuen Projekt wirklich um Open-Source-Software handelt und andere von ihr profitieren sowie eigenständig an der Entwicklung mitwirken können, ist es deshalb wichtig, Code von Beginn an unter einer entsprechenden Lizenz zu veröffentlichen.

Lizenzen in der Entwicklung von Open-Source-Software

Lizenzen

Eine Softwarelizenz legt die Bedingungen fest, unter denen die Nutzung und Weitergabe der Software erlaubt bzw. verboten ist. Für Open-Source-Software existiert eine Vielzahl von Lizenzen. Sie alle haben gemeinsam, dass sie erlauben, den lizenzierten Quellcode unentgeltlich zu nutzen, zu teilen, zu ändern und weiterzuverbreiten. Unterschiede bestehen zwischen ihnen in Bezug darauf, unter welchen Bedingungen die Nutzung und Verbreitung des Codes erlaubt ist:

- **Permissive/Freizügige Lizenzen:** Für die Verbreitung von Kopien des Codes und darauf aufbauende Weiterentwicklungen dürfen Softwarelizenzen frei gewählt werden; selbst die Veröffentlichung unter proprietärer Lizenz ist erlaubt. Beispiele für diese Lizenzform sind MIT, BSD und Apache.
- **Copyleft-Lizenzen:** Die Verbreitung von Codekopien und von darauf aufbauenden Weiterentwicklungen ist erlaubt, muss aber unter derselben Lizenz erfolgen, unter der der Code ursprünglich veröffentlicht wurde. Unterschiede bestehen in Bezug darauf, wie weit die Veröffentlichungspflicht verstanden wird.
 - Schwaches Copyleft: Bei Lizenzen mit sogenanntem schwachem Copyleft ist es erlaubt, nur den veränderten Code unter derselben Lizenz zu veröffentlichen und diesen durch Verlinkung in proprietäre oder anders lizenzierte Software einzubinden. Beispiele sind die GNU Lesser General Public License (LGPL) und die Mozilla Public License (MPL).
 - Starkes Copyleft: Lizenzen mit starkem Copyleft erfordern die Veröffentlichung des gesamten Codes, in dem Codekopien bzw. Weiterentwicklungen genutzt werden. Ein Beispiel hierfür ist die GNU General Public License (GPL). Besonders weit geht die

GNU Affero General Public License (AGPL). Sie fordert auch dann die vollständige Veröffentlichung unter derselben Lizenz, wenn die ursprünglich unter AGPL lizenzierte Software bzw. eine darauf aufbauende Weiterentwicklung als Dienstleistung gehostet wird, ohne dass es zu einer Verbreitung des Codes selbst kommt (Software as a Service).

Lizenzauswahl

Welche Lizenz die richtige ist, hängt von einer Reihe von Faktoren ab. Dazu gehören z. B.:

- **Einfachheit und Bekanntheit:** Generell gilt, je einfacher und je bekannter eine Lizenz ist, desto leichter fällt es anderen zu verstehen, wie sie die Software nutzen und dazu beitragen können. Eine einfache und bekannte Lizenz kann so dazu beitragen, dass eine Community aus Nutzenden und Beitragenden um ein Open-Source-Projekt entsteht.
- **Beliebtheit in der Community:** Wenn ein Projekt an Open-Source-Software anknüpft, um die bereits eine aktive Community besteht, ergibt es oft Sinn, sich an der Lizenzierungspraxis dieser Community zu orientieren. Dahinter können Werte stehen, die in dieser Community als besonders wichtig erachtet werden und einen Einfluss darauf haben, ob Mitglieder sich an der Entwicklung eines neuen Projekts beteiligen. Zusätzlich kann die Verwendung derselben Lizenz eine Voraussetzung dafür sein, Weiterentwicklungen in bestehende Projekte integrieren.
- **Kompatibilität mit anderen Lizenzen:** Nicht alle Open-Source-Lizenzen sind miteinander kompatibel. Open-Source-Software mit inkompatiblen Lizenzen darf nicht kombiniert werden. Deshalb hat die Kompatibilität der gewählten Lizenz einen Einfluss auf die Verbreitung von Software. Copyleft-Lizenzen, die strengere Anforderungen enthalten, haben tendenziell eine geringere Kompatibilität.
- **Eignung für eine Kommerzialisierung:** Wenn aufbauend auf einem Open-Source-Projekt ein Geschäftsmodell entwickelt werden soll, spielen Wettbewerbsvorteile eine Rolle. Copyleft-Lizenzen untersagen beispielsweise, dass Wettbewerber den so veröffentlichten Code für Weiterentwicklungen in proprietärer Software nutzen. Unter permissiver Lizenz veröffentlichte Software hat dagegen unter Umständen bessere Chancen von Unternehmen genutzt zu werden, die Sorge davor haben, den eigenen Code veröffentlichen zu müssen.

Lizenzierung

Sobald die richtige Lizenz für ein Projekt gefunden wurde, ist das Lizenzieren einfach. Es genügt, im Repository, in dem der Code veröffentlicht ist, ein Dokument mit dem Namen LICENSE anzulegen und darin den Text der ausgewählten Lizenz zu kopieren bzw. aus einem im Repository verfügbaren Template auszuwählen.

Lizenzänderungen

Grundsätzlich ist es möglich, die Lizenz eines Open-Source-Projekts zu ändern. Voraussetzung dafür ist jedoch, dass alle Urheber der Änderung ausdrücklich zustimmen. Das sind in der Regel alle Personen, die zur Entwicklung der Software beigetragen haben. Wurde Code von Angestellten

geschrieben, kann das Nutzungsrecht daran allerdings auch bei deren Arbeitgeber*innen liegen (§ 69b UrhG). Alternativ zur Lizenzänderung kann für die Weiterentwicklung auch eine andere, mit der bisherigen kompatible Lizenz verwendet werden. Bei permissiven Lizenzen ist es außerdem möglich, eine Kopie des Projekts unter anderer Lizenz weiterzuentwickeln.

Lizenzverstöße

In Deutschland kann jede*r Urheber*in, der oder die an einem Open-Source-Projekt mitgewirkt hat, juristisch gegen Lizenzverstöße vorgehen (§ 8 Abs. 2, Satz 3 UrhG). Mögliche Konsequenzen eines Urheberrechtsverstoß reichen vom Unterlassungsanspruch gegen die Verbreitung der Software bis hin zu Schadensersatzansprüchen. Auch in Open-Source-Projekten selbst kann es zu Lizenzverstößen kommen, wenn Code beigetragen wird, der nicht unter der Lizenz des Projekts veröffentlicht werden darf. Insbesondere in Projekten, deren Software aktiv kommerziell oder auch nicht-kommerziell vertrieben wird, wird mit Beitragenden deshalb manchmal ein [Contributor License Agreement \(CLA\)](#) oder ein [Developers Certificate of Origin \(DCO\)](#) abgeschlossen, um Urheberrechtskonformität sicherzustellen.

Relevante Gesetze

[Urheberrechtsgesetz \(UrhG\)](#): Darin ist festgelegt, welche Rechte Personen haben, die als Urheber*innen Software entwickelt haben, wie sie diese durchsetzen können und welche Konsequenzen sich aus einem Urheberrechtsverstoß ergeben. Spezifische Regelungen für Software finden sich in §§ 69a ff. UrhG.

Wichtige Organisationen

[ifrOSS, Institut für Rechtsfragen der Freien und Open Source Software](#): ifrOSS ist ein privates Institut, dessen Team zu rechtswissenschaftlichen Fragen rund um Open-Source-Software forscht, veröffentlicht und Vorträge hält. Ein Hauptfokus liegt dabei auf Open-Source-Lizenzen.

[Open Source Initiative \(OSI\)](#): Die US-amerikanische Organisation gibt eine breit akzeptierte Open-Source-Definition heraus und zertifiziert anhand dieser Definition Softwarelizenzen.

Weiterführende Links

- Einführung in den Schutz von Computerprogrammen, Universität Bremen:
<https://www.uni-bremen.de/urheberrecht/wissensplattform/8-schutz-von-computerprogrammen>
- Überblick über verschiedene Open-Source-Lizenzen, GitHub:
<https://choosealicense.com/appendix/>

- Von der Open Source Initiative zertifizierte Open-Source-Lizenzen, Open Source Initiative:
<https://opensource.org/licenses>
 - Übersicht dazu, welche Lizenzen Projekte des Prototype Fund nutzen, Prototype Fund:
<https://prototypefund.de/projects/?filter=dataviz>
 - FAQ zu Lizenzierungsfragen in Open-Source-Projekten, GitHub:
<https://opensource.guide/legal/>
 - Vortrag zu Open-Source-Lizenzen und -Geschäftsmodellen, Frank Karlitschek (Nextcloud):
https://archive.fosdem.org/2020/schedule/event/gpl_and_business/
 - Blogpost zu verschiedenen Lizenzen, ihrer Geschichte und Verbreitung, Prototype Fund:
<https://prototypefund.de/freie-open-source-lizenzen/>
 - Überblick über Urteile zu Open-Source-Software-Lizenzen, ifrOSS:
https://ifrOSS.github.io/ifrOSS/Pages/oss_cases/de#urheberrecht-und-wettbewerbsrecht
 - Argumente gegen Contributor License Agreements, Ben Balter (GitHub):
<https://ben.balter.com/2018/01/02/why-you-probably-shouldnt-add-a-cla-to-your-open-source-project/>
-

Revision #7

Created 19 December 2024 13:51:59 by Sophia

Updated 26 February 2025 10:32:37 by Sophia