ECONSTOR
Make Your Publications Visible.

A Service of

ZBW Leibniz-Informationszentrum Wirtschaft
Leibniz Information Centre for Economics

Wilmer, Thomas

Article — Published Version

Rechtliche Fallstricke des Einsatzes von Open Source Software und freier Software – Hinweise für die Praxis

HMD Praxis der Wirtschaftsinformatik

Provided in Cooperation with:
Springer Nature

Suggested Citation: Wilmer, Thomas (2021) : Rechtliche Fallstricke des Einsatzes von Open Source Software und freier Software – Hinweise für die Praxis, HMD Praxis der Wirtschaftsinformatik, ISSN 2198-2775, Springer Fachmedien Wiesbaden, Wiesbaden, Vol. 58, Iss. 2, pp. 271-287, https://doi.org/10.1365/s40702-021-00705-3

This Version is available at:
https://hdl.handle.net/10419/287689

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

CC BY https://creativecommons.org/licenses/by/4.0/

WWW.ECONSTOR.EU

Mitglied der Leibniz-Gemeinschaft

HMD (2021) 58:271–287
https://doi.org/10.1365/s40702-021-00705-3

SCHWERPUNKT

Rechtliche Fallstricke des Einsatzes von Open Source Software und freier Software – Hinweise für die Praxis

Thomas Wilmer

Eingegangen: 1. Dezember 2020 / Angenommen: 22. Januar 2021 / Online publiziert: 9. Februar 2021
© Der/die Autor(en) 2021

Zusammenfassung Der folgende Artikel bietet einen Überblick über die häufigsten Anwendungsprobleme beim Einsatz und der Integration von freier und Open Source Software in der Praxis. Ein Schwerpunkt liegt dabei auf den Herausforderungen für Softwarehersteller, welche freie Software in ihre Produkte integrieren und vertreiben. Hingewiesen wird auf die unterschiedlichen freien Lizenzarten, Best Practice-Lösungsmöglichkeiten und Compliancefragen sowie die Grenzen der Auslegung der Lizenzen.

Schlüsselwörter Open Source Software · Freeware · Softwarelizenzen · Lizenzmodelle · GPL · Viraler Effekt · Nutzungsrechte · Softwarerecht

Legal Pitfalls of the Use of Open Source Software and Free Software—Best Practice Advice

Abstract The following article provides an overview of the most common application problems when using and integrating free and open source software in practice. One focus is on the challenges for software producers who integrate and distribute free software in their products. Reference is made to the different types of free licences, best practice solutions and compliance issues as well as the limits of the interpretation of the licences.

Keywords Open source software · Freeware · Software licences · Licence models · GPL · Viral effect · Rights of use · Software law

T. Wilmer (✉)
Hochschule Darmstadt University of Applied Sciences, Haardtring 100, 64295 Darmstadt, Deutschland
E-Mail: thomas.wilmer@h-da.de

1 Grundlegende Fragestellungen freier Lizenzen

Als freie Softwarelizenzen sind alle Lizenzen zu betrachten, welche ohne Vergütung von ihren Urhebern zur Verfügung gestellt werden. Wird der Programmcode offen zur Verfügung und Bearbeitung gestellt, spricht man von „Open Source Software“ („OSS“ mit offenem Quellcode). In der Folge soll betrachtet werden, welche rechtlichen Fragen beim Einsatz freier Software zu beachten sind und welcher „Best-Practice“-Umgang sich mit den Lizenzen empfiehlt.

1.1 Grundsätze der Lizenzfreiheit

Urheber freier Software[1] können sich ihre Lizenz frei wählen. Sie können sich beispielsweise einer bestimmten, bereits existierenden Lizenz – wie etwa den Open Source Lizenzen, z. B. der noch zu behandelnden „General Public Licence“ („GPL“) – anschließen und diese für ihr Programm zur Anwendung bringen. Wie die Praxis zeigt, kann dies zu den kuriosesten Lizenzmodellen führen, wie etwa der Free-Beer[2]- oder Free-Chocolate[3]-Lizenzen.

Nach deutschem Recht sind diese Lizenzen zwar nicht immer mit den Prinzipen der Urheberrechts und den zugehörigen restriktiven Vorgaben für die klare Definition von Nutzungs- und Verwertungsrechten vereinbar[4]. Es existiert zum Bereich der freien Lizenzen leider noch nicht sehr viel Rechtsprechung[5], es ist jedoch auch vor dem Hintergrund weniger Urteile anerkannt, dass durch den Download freier Software die Lizenzbedingungen des Urhebers rechtswirksam akzeptiert werden. Werden diese Bedingungen durch die Anwender nicht beachtet, kann dies zum Erlöschen der Nutzungs- und Verbreitungsberechtigung führen[6]. Die Einhaltung der Lizenzbedingungen stellt dementsprechend eine sogenannte „auflösende Bedingung“ des Vertrags nach § 158 Abs. 1 BGB dar[7]. Wird die Lizenzbedingung verletzt, verliert der Nutzer seine Berechtigung zur Verwendung der Software[8].

[1] Zur Abgrenzung der Begrifflichkeiten der freien und Open Source Software: Jaeger/Metzger, Open Source Software Rechtliche Rahmenbedingungen der Free Software, 5. A. 2020, 1D, Rn. 28.

[2] Siehe https://people.freebsd.org/~phk/.

[3] Siehe https://www.chocolatesoftware.com/sbrelay/Readme.txt.

[4] Wobei lizenz- und versionsabhängig nicht immer von einer Auslegung nach deutschem Recht ausgegangen werden kann, siehe zur Anknüpfung an ausländisches Recht Auer-Reinsdorff/Conrad, Handbuch IT- und Datenschutzrecht, 3. A. 2019, Fn. 68; Redeker, IT-Recht, 7. Auflage 2020, A II.1h, Rn. 100.

[5] Siehe dazu Czychowski, Der „Urheberrechts-Troll“ – Wichtige Rechtsfragen von Open Source-Lizenzen?; GRUR-RR 2018, 1.

[6] Redeker, Helmut; IT-Recht, 7. A. 2020, A II.1h, Rn. 98.

[7] LG München I, GRUR-RR 2004, 350, siehe dazu Hoeren, in: Westphalen, Graf von, Friedrich/Thüsing, Gregor; Vertragsrecht und AGB-Klauselwerke, Werkstand 45, März 2020, IT-Verträge, X.2, Rn. 212.

[8] Zum entsprechenden Rückfall der Nutzungsrechte Dreier in: Dreier/Schulze/Specht; Urheberrechtsgesetz, 6. A. 2018, § 69c, Rn. 38.

1.2 Konsequenzen der Verletzung von Lizenzbedingungen

Die Verletzung von Lizenzrechten kann verschiedene Rechtsfolgen nach sich ziehen.

Zum einen führt dies zum bereits genannten Erlöschen der Lizenz, welches einen Anspruch des Urhebers auf Unterlassung der Nutzung der Software zur Folge hat[9]. Grundsätzlich erlischt jede Lizenz, wenn die Lizenzbedingungen – auch bei kostenloser Software – nicht erfüllt werden. Nutzt der Verletzer die Lizenz nur zu eigenen Zwecken, muss er die entsprechende Nutzung umgehend einstellen. Handelt es sich um ein unternehmenswichtiges Programm – oder den Teil eines solchen Programms bei Integration freier Software in eigene Systeme – kann dies entsprechend gravierende Konsequenzen zur Folge haben.

Zum Anderen kommen neben dem – auch im einstweiligen Rechtsschutz durchsetzbaren – Unterlassungsanspruch auch Schadensersatzansprüche für die Vergangenheit in Betracht[10]. Zur Aufklärung des Einsatzes und des Vertriebs der Software kann der Urheber außerdem Auskunftsansprüche geltend machen. Dies gilt auch dann, wenn die OSS lediglich zu Testzwecken in die eigene Software integriert wurde und keine produktive Funktion erfüllt[11].

Besonders problematisch sind in der Praxis daher Fälle, in welchen die freie Software in eigene, proprietäre Software integriert wurde[12] und an diesem Gesamtprodukt Rechte an Partner und/oder Kunden vergeben wurde. Diese haben einen Nacherfüllungsanspruch nicht nur auf Freiheit der Software von Sachmängeln (wie Programmierfehlern), sondern auch auf Freiheit von Rechtsmängeln. Wird Software unter Verletzung von Rechten der Open Source Programmierer vom Softwarehersteller geliefert, handelt es sich um einen solchen Rechtsmangel. Der Kunde hat bei einer dauerhaften Überlassung der Software dann das Recht, Nacherfüllung nach § 439 BGB, das heißt den Nachkauf der Rechte zu verlangen. Anders als beim Schadensersatz für die Vergangenheit kann der Urheber jeden Preis dafür verlangen, dass er nachträglich Rechte zur Weiterübertragung einräumt. Außerdem kann er den Softwarehersteller durch Drohung mit der Durchsetzung von Unterlassungsansprüchen in eine problematische Lage bringen. In den letzten Jahren hat hier vor allem der Linux-Programmierer McHardy für Schlagzeilen gesorgt, indem er Unternehmen entsprechend auf Lizenzverstöße hingewiesen und Unterlassungsverfahren angestrengt hat[13]. Wird ein solcher Unterlassungsanspruch geltend gemacht, muss der Programmierer allerdings neben der Urheberrechtsfähigkeit seines Beitrags nachweisen, dass sein Programmieranteil (zu den meist in sukzessiver Kooperation zahlloser Programmierer entstandenen Open Source Software-Produkten wie Linux)

[9] Schneider, Handbuch EDV-Recht, 5. Aufl. 2017, X., Rn. 158.

[10] Noch nicht höchstrichterlich geklärt ist, ob auch bei kostenloser freier Software ein Schadensersatz in Betracht kommt (zuletzt ablehnend: OLG Hamm, GRUR-RR 2017, 421) und ob der Zwang zur kostenlosen Verbreitung kartellrechtlich zulässig ist, siehe Czychowski, S. 5.

[11] LG Bochum, CR 2011, 289.

[12] Siehe Redeker, IT-Recht A II.1h, Rn. 99.

[13] Siehe dazu von Welser, Kampf um Linux: ist die Freiheit der Open-Source-Software bedroht?, GRUR-Prax 2018, 164; zu Details der Mc Hardy-Verfahren Czychowski, S. 2 ff.

bereits vom ursprünglichen Willen des ersten Urhebers vorausgedacht und umfasst war[14] und er somit als Miturheber für eine Klage aktivlegitimiert ist[15].

Neben den zivilrechtlichen Ansprüchen ist für den Softwarehersteller außerdem zu beachten, dass ein Verstoß gegen Lizenzbedingungen bei einer Verbreitung der Software nach §§ 106, 108 UrhG strafbar sein kann.

2 „MIND THE GAP“: Lücken zwischen ein- und ausgehenden Rechten vermeiden.

Um einen Verstoß gegen die Lizenzbedingungen zu erkennen und zu vermeiden, sollte eine sogenannte „IP Due Diligence“, also die Rechteanalyse durchgeführt werden. Entscheidend sind die Analyse und der Vergleich ein- und ausgehender Rechte, je nach Vertriebsmodell (bis hin zum Blockchain-Einsatz[16]) und Rechtemanagement des Unternehmens und des Geschäftsmodells (Abb. 1).

Jedes an Vertriebspartner oder Endkunden (oder ggf. innerhalb einer Unternehmensgruppe) vergebene Recht muss durch die eingehenden Rechte gedeckt sein. Eingehende Rechte können von angestellten Programmierern stammen, welche diese Rechte nach § 69b UrhG gesetzlich an ihren Arbeitgeber übertragen. Stammen

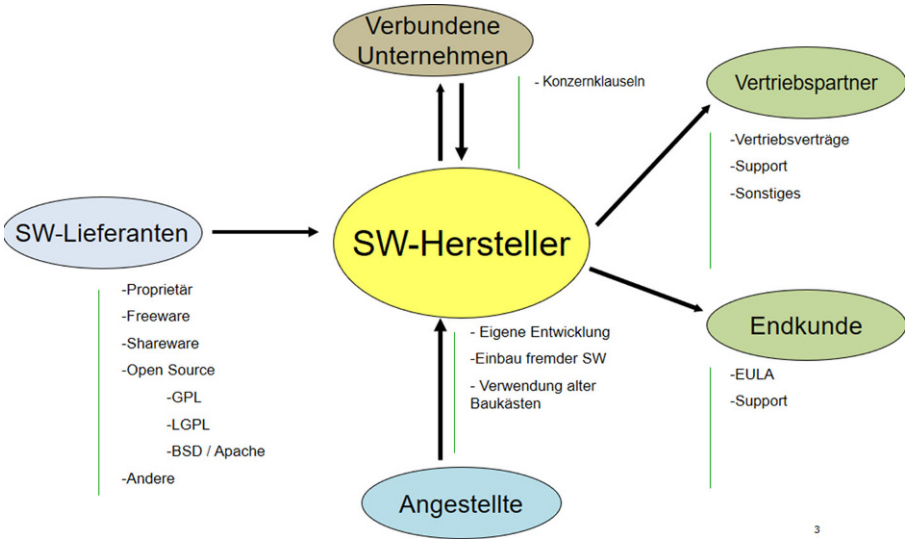


Abb. 1 Lizenzfluss bei softwareerstellenden Unternehmen

[14] Welser, 164 m. w. N.

[15] OLG Hamburg ZUM-RD 2019, 575; Siehe Einzelheiten zur Klagberechtigung und Miturheberschaft bei OSS bei Thum, in: Wandtke, Artur-Axel/Bullinger, Winfried; Urheberrecht, 5. A. 2019, § 8 UrhG, Rn. 182 ff.

[16] Rein, in: Sassenberg, Thomas/Faber, Tobias, Rechtshandbuch Industrie 4.0 und Internet of Things, 2. Auflage 2020, § 14, Rn. 57.

die Rechte von den Urhebern freier Lizenzen (durch Download der Software von entsprechenden Plattformen), muss geprüft werden, ob die Bedingungen der Lizenz für eine Weitergabe erfüllt worden sind. Zu keinem Zeitpunkt dürfen mehr Rechte herausgegeben werden, als das Unternehmen von den Urhebern erhalten hat, um die oben geschilderten Unterlassungsansprüche zu verhindern. Ansonsten liegt der Fall einer unbedingt zu vermeidenden Lizenzlücke vor („Mind the Gap“).

3 Wahrnehmung und Beachtung der Lizenzbedingungen

In der Praxis wird die rechtliche Qualitätssicherung bei der Verwendung freier Lizenzen immer noch nicht ausreichend beachtet[17]. Zwar sind zahlreiche Tools im Einsatz für die Vermeidung von Über- oder Unterlizenzierungen bei proprietären, also klassischen Kauf- oder Mietlizenzen. Bei freien Lizenzen fällt jedoch eine Standardisierung schwerer, da insbesondere bei Nicht-Open-Source-Lizenzen zahllose Lizenzmodelle existieren, welche eine Einzelfallprüfung erfordern.

Zentral beim Einbau fremder Software sind daher die Analyse und Beachtung der einzelnen Lizenz. Um festzustellen, ob und welche OSS in einer Software enthalten ist, empfehlen sich Scanprogramme wie Black Duck, Fossid, Palamida und White Source[18].

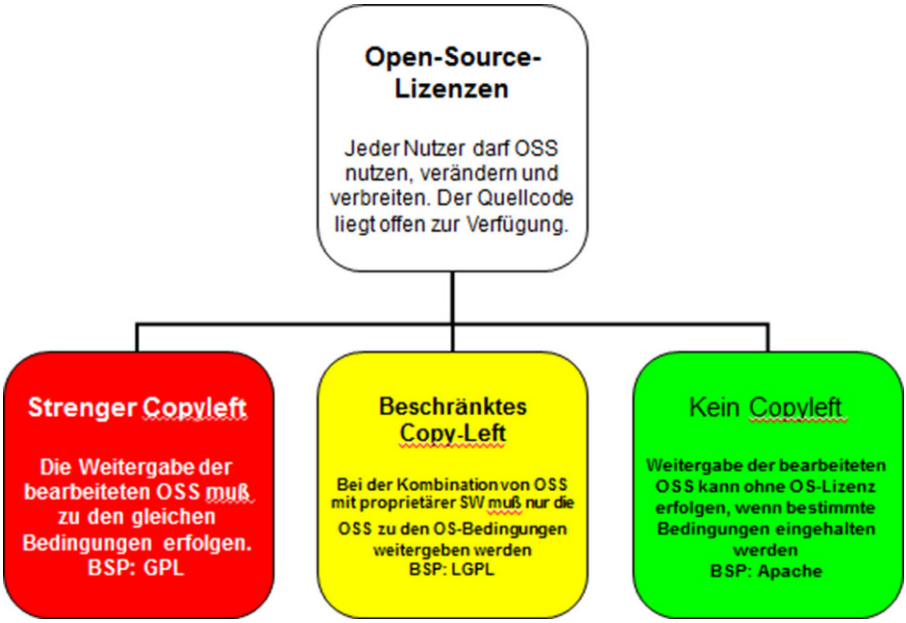


Abb. 2 Basismodelle der Open Source Lizenzen

[17] Beispiele für Statistiken zu Verletzungen von freien Lizenzen finden sich beispielsweise unter https://www.blackducksoftware.com/.

[18] Siehe dazu https://orcro.co.uk/services/openchain-compliance/.

3.1 Beispiel Open Source Lizenzen

Der Grundgedanke der Open Source-Software zeichnet sich durch die freie Bereitstellung des Quellcodes zur Weiterbearbeitung aus[19]. Bei Open Source gilt die Idee, dass jeder an einem Gemeinschaftswerk mitarbeiten kann, aber dafür als Lizenzbedingung auch das Ergebnis frei zur Verfügung stellen muss. Dies nennt man – als Gegensatz zum klassischen „Copyright“ bei „Verkaufssoftware“ – das „Copyleft“.

Open Source Lizenzen gliedern sich bei der Umsetzung des „Copyleft“ – Prinzips in drei verschiedenen Grade: Strengem Copy-Left, beschränktem Copy-Left und Lizenzen ohne Copyleft (Abb. 2).

In den Basisformen der jeweiligen Lizenzmodelle bedeutet dies für die Integration der freie Software Folgendes:

3.1.1 Strenges Copyleft

Jeder, der Software unter einer solchen strengen Lizenz (Rot) mit einer eigenen Software (grün) kombiniert und verbreitet, muss das Gesamtergebnis unter der strengen Lizenz weitergeben (und den Quellcode veröffentlichen) (Abb. 3).

Allerdings gibt es keine klare Regelung[20] welche etwa besagt, wieviel Prozent „OSS“ diesen viralen Effekt erzeugt, es kann daher auch ein kleiner „Schnipsel“ OSS dazu führen, dass der überwiegend eigene selbstprogrammierte Anteil unter

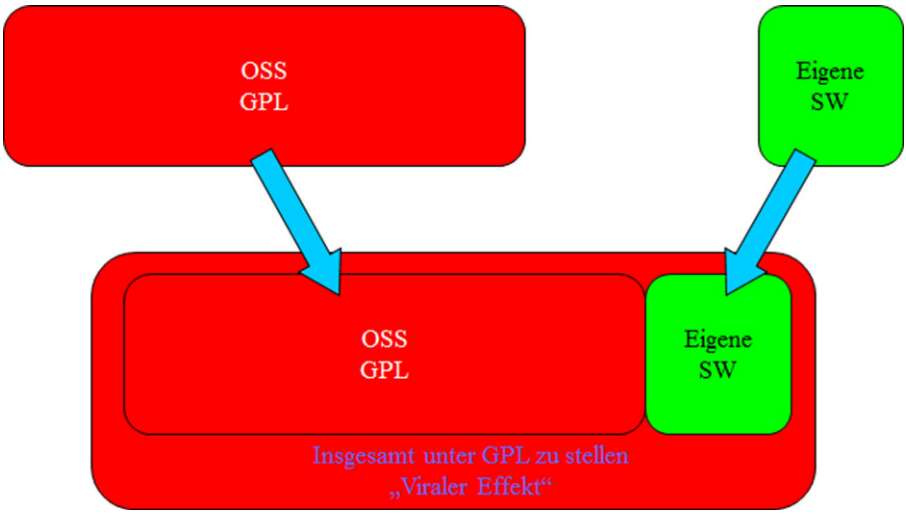


Abb. 3 Strenges Copyleft: Integration großer OSS-Anteile

[19] Siehe zur Genese, zu den Lizenzarten und den Auslegungshinweisen die Informationen der Free Software Foundation (fsf.org) und der Open Source Initiative (www.opensource.org), nützliche rechtliche Informationen zur Zuordnung der zahlreichen freien Lizenzen in das System der Copyleft-Arten finden sich weiterhin auf den Seiten des Instituts für Rechtsfragen der OSS unter ifross.org.

[20] Siehe zu den Auslegungsschwierigkeiten Redeker, A II.1h, Rn. 98.

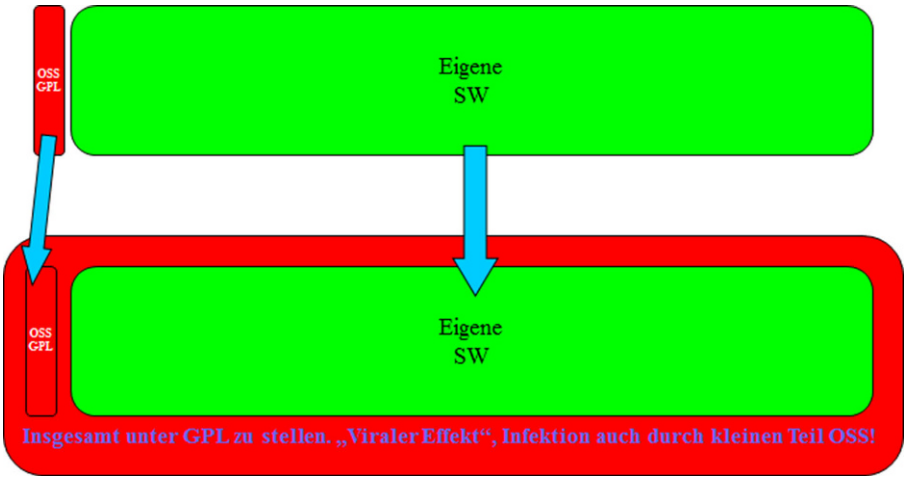


Abb. 4 Strenges Copyleft: Integration kleiner OSS-Anteile

die strenge Copyleft-Lizenz gestellt werden muss[21]. Die Ziffer 2 b) GPL V 2 regelt dies wie folgt: „You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License." Demnach ist zu klären, ob ein aus der OSS „abgeleitetes Werk" vorliegt (in der GPL v 3 lautet die entsprechende „virale" Bestimmung in Ziffer 5 c) Abs. 2 als „work based on the earlier work".) (Abb. 4).

Die Abgrenzung zwischen einer unzulässigen bzw. zu einem viralen Effekt führenden Kombination eigener Software mit GPL-OSS und einer zulässigen Verbindung beider Programme kann in der Praxis sehr problematisch sein. So sind Tools auf Basis von der GPL unterstelltem Linux-Betriebssystems in der Regel ohne viralen Effekt zulässig; ebenso werden Contentmanagementsysteme auf Basis von OSS in aller Regel zulässig sein[22]. Die Entwicklung von Treibern für Linux kann dagegen zum viralen Effekt führen, wenn zu eine enge Interoperabilität der Programm vorliegt[23].

Der virale Effekt tritt nur bei einer Verbreitung der OSS ein. Der Begriff der Verbreitung ist jedoch nach deutschem Recht schwierig zu definieren[24]. Bei bestimmten Lizenzen besteht die Gefahr, dass die Weitergabe innerhalb einer Unternehmensgruppe bereits den viralen Effekt auslöst[25]. Vielfach wird daher überlegt, OSS als

[21] Ob die virale Regelung nach deutschem AGB-Recht wirksam ist, ist umstritten, vgl. Czychowski, S. 5.

[22] Siehe im Einzelnen Galetzka, Der strenge Copyleft-Effekt der GNU General Public License bei Interaktion von proprietärer Software mit Standard-CMS wie Joomla, Typo3 oder Wordpress, DSRITB 2015, 647 ff.

[23] Galetzka, 652, 654 zu den formalen und inhaltlichen Kriterien.

[24] Wiebe in Spindler/Schuster, Recht der elektronischen Medien, 4. A. 2019, § 69c, Rn. 56; Redeker, A II.1h, Rn. 100.

[25] Siehe zur Diskussion zur Weitergabe in der Gruppe Heinzke/Burke, Open-Source-Compliance, CCZ 2017, S. 56; siehe auch die FAQ-Hinweise zur GPL unter https://www.gnu.org/licenses/gpl-faq.html#

Bestandteil der eigenen Software zu nutzen, ohne sie formal zu „verbreiten“. Dennoch kann in der Praxis der Versuch scheitern, die OSS nicht gemeinsam mit der eigenen Software zu verbreiten, sondern sie etwa als Systemumgebung zu definieren, welche der Kunde selbst stellen muss[26]. Wird die eigene Software nach der Installation beim Kunden so eng mit der (als Systemumgebung definierten) OSS zusammen funktionieren, dass ein gemeinsames Werk vorliegt, kann trotz des getrennten Verbreitungswegs der virale Effekt eintreten. Das LG Berlin hat die entsprechende Verbindung von OSS und eigener Software wie folgt als „Sammelwerk“ beschrieben:

> „Die Infizierung eines Sammelwerks insgesamt bei Verwendung von Open-Source-Software in einzelnen Teilen eines Sammelwerks begegnet keinen Bedenken, da das Sammelwerk eine einheitliche Funktionalität aufweist und maßgeblich von den Open-Source-Bestandteilen abhängt“.[27]

Liegt der virale Effekt vor, führt dies zu einem Unterlassungsanspruch gegenüber der gemeinsamen Verbreitung der OSS mit eigener Software des Herstellers zu

GPL 4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. **However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.**

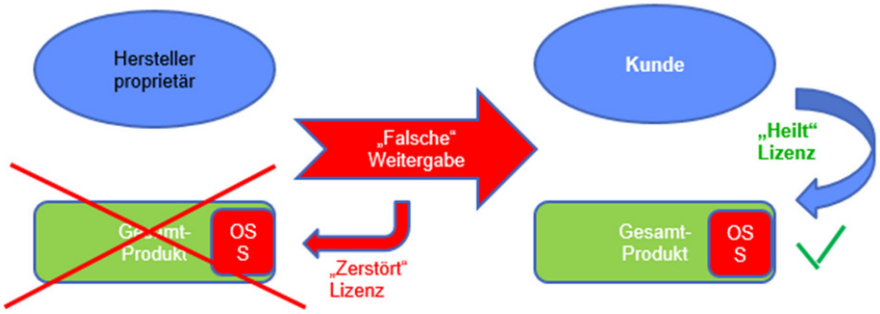


Abb. 5 Strenges Copyleft: Heilung der Lizenzverstöße

InternalDistribution: *„Is making and using multiple copies within one organization or company ‚distribution‘“? (#InternalDistribution). „No, in that case the organization is just making the copies for itself. As a consequence, a company or other organization can develop a modified version and install that version through its own facilities, without giving the staff permission to release that modified version to outsiders. However, when the organization transfers copies to other organizations or individuals, that is distribution. In particular, providing copies to contractors for use off-site is distribution“*; gegen eine Auslegung als Verbreitung bei konzerninterner Weitergabe u. A. Dreier/Schulze, § 17, Rn. 7 ff.

[26] Teilweise wird auch versucht, die OSS-Anteile separat zu „verschenken“, während der eigene Softwareanteil verkauft wird, siehe dazu Redeker, A II.1h, Rn. 618.

[27] Landgericht Berlin, GRUR-RR 2012, 107.

dessen Lizenzbedingungen, da der Hersteller für das Gesamtprodukt die virale Open Source Lizenz hätte verwenden und den Quellcode offenlegen müssen[28].

Das Erlöschen der Lizenz aufgrund dieses Verstoßes hat nach deutschem Lizenzrecht nicht nur ein Erlöschen der Lizenz gegenüber dem Softwarehersteller, sondern auch gegenüber dessen Kunden zur Folge. Anders als bei der Verbreitung von gegenständlichem Eigentum existiert kein „gutgläubiger Rechteerwerb", welcher die Nutzung durch den Endkunden gestattet, wenn dieser nicht von der Lizenzverletzung wusste oder wissen musste. Wurde die OSS falsch integriert und damit der virale Effekt ausgelöst, kommt jedoch bei manchen Lizenzen eine „Heilung" durch eine nachholende Lizenzierung des Endnutzers in Betracht[29], so etwa nach Art. 4 der GPL V 2 (Abb. 5).

Ergänzt wird dieser Absatz der GPL V 2 durch das sogenannte „Linux Kernel Enforcement Statement" der Linux-Entwicklergemeinschaft:

> „Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice."

Ob diese Heilung durch den Endkunden jedoch tatsächlich durchgreift und welche Voraussetzungen dafür im Einzelnen erforderlich sind, ist umstritten[30]. Unabhängig von der Heilungsmöglichkeit hat der Erwerber der Software gegen den Lieferanten jedoch einen Anspruch auf die Stellung einer lizenzgemäßen Software – einschließlich rechtmäßig integrierter Open Source Software.

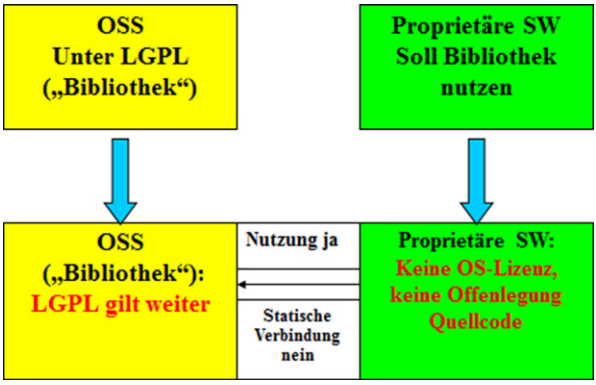


Abb. 6 Beschränktes Copyleft: Integrationsvorgaben

[28] LG Köln, CR 2014, 704.

[29] LG München GRUR-RR 2004, 350, 351.

[30] Redeker, A II.1h, Rn. 100.

3.1.2 Beschränktes Copyleft

Weniger gravierende Einschänkungen enthalten Lizenzen mit dem sogenannten „beschränkten Copyleft“, etwa der „LGPL“-Lizenz („Lesser General Public License“). Nach dieser Lizenz darf die freie Software nur als Programm-Bibliothek[31] genutzt und nur in einer bestimmten Weise (dynamisch, nicht statisch, siehe Abb. 6) mit dem eigenen ausführbaren Programm kombiniert werden.

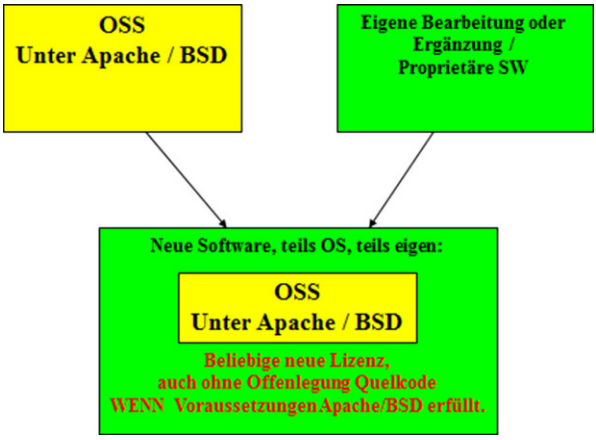
Danach ist es von entscheidender Bedeutung, dass das unter der LGPL stehende Programm „sichtbar“ in einem Bibliotheksordner gespeichert ist die entsprechende Verlinkung zwischen dem eigenen ausführbaren Programm und der Bibliothek mit der OSS-Software auf die Einhaltung der Lizenzvorgaben geprüft wird.

3.1.3 Kein Copyleft

Die am wenigstens restriktiven OSS-Lizenzen sind diejenigen ohne den Copyleft-Effekt, wie etwa die „Berkeley Software Distribution (‚BSD‘) License“ oder der „Apache License“. Diese Lizenzen sind somit „nicht viral“, dennoch sind sie – wie bei Freeware-Lizenzen – nur unter Beachtung der enthaltenen Hinweise zu verwenden (Abb. 7).

Ebenso wie bei Freeware-Lizenzen (s. unten) ist hier trotz der vermeintlich „liberalen“ Lizenzen zu beachten, dass auch Lizenzhinweise wie die Aushändigung von Lizenztexten an die Endkunden oder die Information über Haftungsausschlüsse oder Lizenzbedingungen penibel eingehalten werden müssen[32].

Abb. 7 Kein Copyleft



[31] Dateien mit den Endungen „.dll“ oder „.lib“, welche sich dann in einem gesonderten Ordner befinden.

[32] Siehe zur Formverletzung bei der Weitergabe von OSS-Lizenztexten LG München, CR 2008, 57.

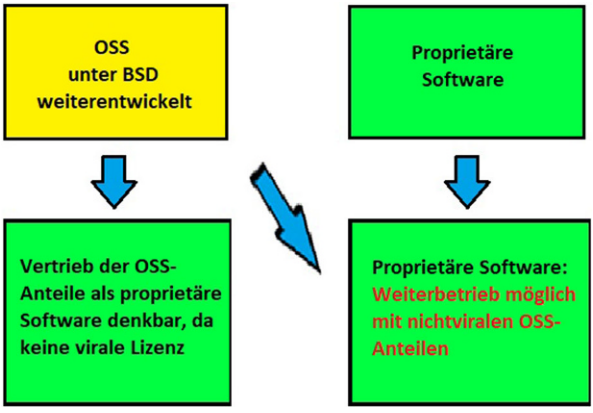
3.2 Weitere Lizenzarten, Interkompatibilität

Neben der Aufteilung nach den Copyleft-Arten im Bereich der Open Source Software können freie Lizenzen auch noch nach weiteren Merkmalen unterschieden werden[33]. Es gibt freie Lizenzen, bei denen der Nutzer die Wahl hat, verschiedene Versionen der Lizenz, insbesondere spätere Versionen („any later“) zu benutzen. Da die jeweiligen Versionen teilweise große Unterschiede im Hinblick auf die Einsatzmöglichkeiten und eventuelle Vergütungsmöglichkeiten beinhalten (so etwa GPL V 2 und GPL V 3[34]) ist genau auf die Auswahl der jeweiligen Version zu achten. Nicht zulässig ist dagegen eine Auswahl einer Lizenzart durch den Nutzer, so etwa ein willkürlicher Wechsel von der durch den Urheber gewählten GPL zu der liberalen LGPL[35].

Insbesondere strenge Copyleftlizenzen können Bedingungen enthalten, welche mit dem Einsatz anderer Lizenzen inkompatibel sind. Daher muss beim Einsatz mehrerer OSS-Bestandteile auch auf die Vereinbarkeit der zu integrierenden Lizenzmodelle geachtet werden, dies wiederum kann etwa dazu führen, dass bestimmte Versionen von OSS-Lizenzen, wie etwa die GPL V 3 zu wählen sind, die besser mit anderen Lizenzen kompatibel sind[36].

Last not least gibt es Lizenzen, die eher für den Cloud-Einsatz von ihren Urhebern gedacht sind[37] und die entsprechenden Einsatzkonstellationen abdecken, auch hier sind die unterschiedlichen Nutzungsmöglichkeiten zu bedenken.

Abb. 8 Dual Licensing



[33] Siehe auch Jaeger/Metzger, 1D, Rn. 30; unterschieden werden kann auch nach Lizenzen, welche Patentfragen (insbesondere nach US-Recht) berücksichtigen.

[34] Redeker, A II.1h, Rn. 98.

[35] LG Köln, CR 2014, 704.

[36] Redeker, A II.1h, Rn. 99.

[37] Die Affero GPL („AGPL“) V 3-Lizenz berücksichtigt dies in ihrem § 13 („13. Remote Network Interaction; Use with the GNU General Public License“).

3.3 Dual Licensing

Denkbar sind auch – etwa unter der (für Cloud-Einsatz geeigneten) AGPL V 3-Lizenz – duale Lizenzen, also die zusätzliche Freigabe des Programmcodes unter einer Open Source-Lizenz neben der Veröffentlichung unter einer eigenen „Kauflizenz". Es entstehen damit zwei unterschiedliche „Lizenzschienen" der Software. Hierbei darf jedoch keine ungeprüfte Reintegration (also Wiedereingliederung in den proprietären, eigenen Quellcode) der unter OSS erfolgten Weiterentwicklungen durchgeführt werden (Abb. 8).

Dual Licensing bietet auch den Vorteil, dass bei Lizenzverletzungen durch Dritte Schadensersatz für die Vergangenheit verlangt werden kann[38].

4 FREEWARE mit geschlossenem Quellcode

Freeware-Lizenzen ohne offenen Quellcode erlauben grundsätzlich eine Weitergabe ohne den viralen Effekt der strengen Open Source-Lizenzen. Dennoch sind die Freeware-Lizenzbedingungen ebenso strikt zu beachten wie etwa diejenigen der Open Source-Lizenzen ohne Copyleft. Auch hier gilt, dass die Verletzung der Lizenzbedingungen zu einem Entfallen der Nutzungsberechtigung führt, sei es bei der eigenen Nutzung der Software oder bei der Integration in ein eigenes Produkt.

Auch wenn die Freeware-Lizenz kostenlos ist und nur eine einzige Bedingung enthält, wie etwa eine Registrierung oder eine Danksagung oder die Weitergabe des Lizenztextes in schriftlicher Form, hat die Nichtbeachtung wiederum das Nichtbestehen der Lizenz mit den oben geschilderten Konsequenzen zur Folge. Auch hier kann es daher neben Schadensersatzforderungen zu einem Unterlassungsanspruch durch den Urheber der Freeware kommen, der insbesondere bei der Integration in ein Produkt zu dem gefürchteten „Nachkaufzwang" führen kann[39].

Auch bei Freeware ist somit jede Voraussetzung in der Lizenz unbedingt zu lesen und zu erfüllen, wenn man die Software verwenden und insbesondere in ein eigenes Produkt integrieren und vertreiben möchte:

Ein Beispiel ist die wie folgt lautende „Picture-postcard-with-nice-stamp-Lizenz"[40]:

> „Copyright (C) 1997–2006 by François PIETTE Rue de Grady 24, 4053 Embourg, Belgium francois.piette@overbyte.be Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions: 1. The origin of this software must not be misrepresented, you must not claim that you

[38] OLG Hamm, GRUR-RR 2017, 421.

[39] Welcher auch bei OSS theoretisch dadurch möglich ist, dass der Urheber dem (verletzenden) Nutzer für die Zukunft exklusive Nutzungsrecht abzüglich der bis dahin eingeräumten allgemeinen Nutzungsrechte für die Allgemeinheit einräumt (LG Frankfurt/Main, ZUM-RD 2006, 528), was allerdings voraussetzt, dass der Urheber eindeutig als alleiniger Rechteinhaber definiert werden kann.

[40] Siehe auch http://thundercloud.net/rainbow/html/gb/licencja.htm als weiteres Beispiel für eine Postkartenlizenz.

wrote the original software. (...) 4. You must register this software by sending a picture postcard to the author. Use a nice stamp and mention anything you like to say."

Es muss daher in der Praxis die Einhaltung der Lizenzvorgaben dokumentiert werden. Dies kann hier durch ein Einwurfeinschreiben mit einer Bildpostkarte mit einer „netten" Briefmarke geschehen (als Briefmarkenmotiv sollten beispielsweise Sonderbriefmarken verwendet werden, wie etwa Marken zu besonderen Anlässen, Marken mit freundlichen Motiven). Das Einschreiben wäre hier dann an François Piette Rue de Grady 24, 4053 Embourg, Belgium zu senden und ein Scan der Postkarte (beidseitig) mit Einschreibebeleg zu archivieren, um die Lizenzberechtigung nachweisen zu können.

Da in der Praxis Freeware regelmäßig nicht nach der Geeignetheit der Lizenz, sondern der Geeignetheit der lizenzierten Software ausgesucht wird, kann es passieren, dass solch kuriose Lizenzen (wie auch etwa die Chocolateware-Lizenz[41]) eingehalten und dokumentiert werden müssen.

Auch bei den „liberalsten" Lizenzen, wie Freeware-Lizenzen oder Open Source-Lizenzen ohne Copyleft ist nicht jeder Einsatz möglich, selbst wenn sie keinerlei Lizenzbedingungen, also nicht einmal Urheberhinweispflichten oder Haftungsaus-

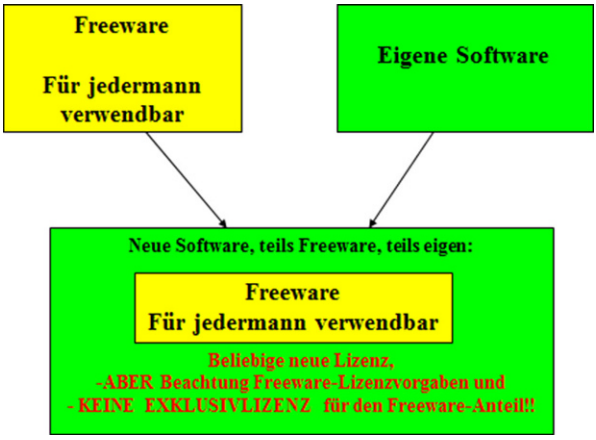


Abb. 9 Integration von Freeware

[41] „*PRICE AND LICENSING:—SBRelay is ChocolateWare! ChocolateWare is a combination of Freeware and DonationWare, but with a twist ... All donations of $1 or more are converted directly to chocolate and given to my wife. It's my mission to hear her say, ‚I love Flight Simulator!' some time before I die. This is also good for you, because if my wife is happy, it means I have more time to work on programs like this. I'll make every effort to convert your donations to chocolate and give them to my wife within 48 h of receiving your donation. I'll also send an email confirmation back to you, to let you know how it was received (and eaten). You can even request a particular type of chocolate if you like. This can be candy, cakes, pies, ice cream, or just about anything else you can think to put chocolate in (or on, or with). To make a chocolate donation, please donate to my PayPal account*. https://www.chocolatesoftware.com/sbrelay/Readme.txt".

schlüsse[42] enthalten. Die Urheber der Freeware räumen dem Nutzer nie Exklusivrechte, also ausschließliche Nutzungsrechte ein. Daher können auch nie Exklusivrechte an einer Gesamtsoftware, welche Freeware enthält, eingeräumt werden. Es können nur Exklusivrechte an der eigenen Software eingeräumt und einfache Rechte an der Freeware weitergegeben werden. Dementsprechend muss unbedingt bei der Integration solcher Lizenzen in zu vertreibende eigene Software bei der Nutzungsrechtsübertragung an Dritte differenziert werden zwischen der eigenen Exklusivlizenz und der nicht ausschließlichen Freeware-Lizenz (Abb. 9).

5 Freie Entwicklungswerkzeuge

Zur Freeware (bzw. Shareware) gehören auch etliche Entwicklungswerkzeuge, welche für die Programmierung eingesetzt werden können. Auch hier ist die Lizenz penibel einzuhalten. Zwar wird in aller Regel nicht davon auszugehen sein, dass der Urheber des Entwicklungswerkzeugs Rechte an den mit dem Werkzeug erzeugten Arbeitsergebnissen postulieren kann (dies wird allenfalls bei komplexen Grafikerzeugnissen der Fall sein). Werden solche Werkzeuge aber unter der falschen Lizenz eingesetzt, kann die damit erstellte Software nicht benutzt werden. Dies gilt beispielsweise für die Nutzung der Shareware-Version eines Werkzeugs, welche nur für nichtkommerzielle Einsatzzwecke freigegeben ist. Sollten also unternehmenseigene Entwickler statt des vom Unternehmen lizenzierten Entwicklungswerkzeugs ihr eigenes „Lieblingswerkzeug“ einsetzen, kann dies ebenfalls zu Unterlassungsansprüchen (auch gegenüber den Kunden des Unternehmens) führen.

6 Empfehlungen

Grundsätzlich sollten die Lizenzbedingungen ernst genommen und akribisch umgesetzt werden, dies gilt sowohl für die Art der Nutzung, die mögliche Registrierung der Software oder die Verbindung der Software mit eigenen Programmen.

6.1 Nutzer/Endkunde

Als Nutzer der Software sollte bei relevanten Softwaresystemen geprüft werden, ob bei einem Bezug der Software von einem Hersteller zusammen mit dessen eigenen Programmteilen die Vorgaben der Lizenzen eingehalten worden sind. Im Zweifel sollte eine Zusicherung der Einhaltung der Lizenzvorgaben der freien Software verlangt werden.

[42] So etwa die „WTF“-Lizenz, hier zur „What the Hell“-Lizenz umformuliert: „*DO WHAT THE HELL YOU WANT TO PUBLIC LICENSE Version 2, December 2004. Copyright (C) 2004 Sam Hocevar 14 rue de Plaisance, 75014 Paris, France. Everyone is permitted to copy and distribute verbatim or modified copies of this license document, and changing it is allowed as long as the name is changed. 0. DO WHAT THE HELL YOU WANT TO PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION § 0. You just DO WHAT THE HELL YOU WANT TO.*“

Insbesondere wenn die freien Softwarebestandteile zwar nicht durch den Hersteller des Gesamtprodukts geliefert werden, sondern dieser die freien Softwarebestandteile als Systemsoftware ausweist, welche der Nutzer selbst installieren soll, ist Vorsicht geboten. Hier sollte abgeklärt werden, ob die freien Softwareanteile unter einer viralen Lizenz stehen und möglicherweise aufgrund engen Zusammenwirkens mit den eigenen Anteilen des Herstellers „infektiös" wirken könnten.

Der Lizenzvertrag des Herstellers sollte typischerweise auf die Lizenzbedingungen der freien Softwareanteile hinweisen und/oder entsprechende Drittlizenzbedingungen weitergeben. Dies kann nur dann entbehrlich sein, wenn gesichert ist, dass keine solchen Softwareanteile implementiert wurden (was jedoch in der Praxis untypisch wäre).

Handelt es sich bei dem Nutzer um eine Unternehmensgruppe, sollte die Integration der freien Softwareanteile in eigene Systeme durch unternehmensinterne Entwicklungsgesellschaften erfolgen. Diese wären dann von den nutzenden Mitgliedern der Unternehmensgruppe mit der Entwicklung (wie ein externer Auftragnehmer) zu beauftragen werden („verlängerter Arm des Integrators"). Aufgrund der ungewissen Rechtslage sollte im Zweifel vermieden werden, dass – unabhängig von der Weitergabe in der Gruppe – die an eine Verbreitung angeknüpften Lizenzpflichten ausgelöst werden. Verbreitete Methoden zur Vermeidung des viralen Effekts bei der Weitergabe wären etwa die formale Trennung von OSS und proprietären Komponenten, je nach Lizenz.

6.2 Softwarehersteller

Für Softwarehersteller empfiehlt es sich, die eigenen Lieferanten proprietärer Software (welche typischerweise OSS in nicht geringem Umfang[43] integriert hat) der gleichen Prüfung zu unterziehen wie unter 6.1 dargestellt. Grundsätzlich sollte bei der Integration freier Software der Einsatz viraler Lizenzen soweit möglich aufgrund der damit verbundenen Rechtsunsicherheiten vermieden werden, soweit nicht bewusst die Freilegung des Quellcodes bei der Weitergabe akzeptiert werden soll. Ist dies nicht möglich, ist zu prüfen, ob der Begriff der Verbreitung dadurch vermieden werden kann, dass der Endkunde gebeten wird, die freien Softwareanteile selbst zu installieren. Diese „Abmilderung" des Vertriebs der freien Anteile wird jedoch (s. oben) dann keine ausreichende Sicherheit bieten, wenn die freien Anteile auf den Systemen des Kunden mit den Anteilen des Herstellers im Sinne des Sammelwerks eine gemeinsame Funktionalität ausüben, die wesentlich auf den freien Anteilen beruht. Daneben ist auch zu prüfen, ob beim Einsatz mehrerer freier Softwareanteile, die unter verschiedenen Lizenzen stehen, diese – insbesondere beim strengen Copyleft – miteinander kompatibel sind.

Im Entwicklungs- und Vertriebsprozess ist neben der technischen Qualitätssicherung auch auf eine fortlaufende rechtliche Qualitätssicherung zu achten: Erforderlich ist eine strukturierte Prüfung der integrierten freien Software auf die Einhaltung der Lizenzvorgaben und die Kompatibilität der Lizenzen untereinander. Hierzu sollte eine Unterstützung der Entwicklungsabteilung durch Schulungen und Unterlagen über

[43] Siehe zur typischerweise stattfindenden OSS-Integration Jaeger/Metzger, 1D, Rn. 30.

Standards der OSS-Integration erfolgen. Bei der Einhaltung der Lizenzvorgaben zur Lieferung von Texten und Urheberhinweisen helfen Programme wie Fossology[44] oder Scancode[45], die Ergebnisse zum Codescan sollten in einem OSS-Lizenzverwaltungssystem dokumentiert werden[46].

Stellt man nach einer initialen Prüfung eine Lizenzverletzung fest, ist zu klären, ob und in welcher Form diese – etwa durch Ersetzen der freien Software ausgeräumt werden kann. Hilfreich ist es, wenn beim Einsatz von Vertriebspartnern die Vertragshändlerverträge eine gewisse Flexibilität zur Ersetzung von Softwareanteilen aufweisen, um nicht einem Nacherfüllungsanspruch und dem gefürchteten Zwang zum Nacherwerb der Lizenzen zu ungewissem Preis ausgesetzt zu sein.

Leilinien für die unternehmensinterne OSS-Compliance finden sich beim „Open Chain Project“/(https://www.openchainproject.org/), so etwa zu Supply Chain-Fragen des OSS Einsatzes im Dokument „Open Source Software License Compliance General Public Guide“[47].

Werden diese Maßnahmen beachtet, sind die rechtliche Grundprobleme beim Einsatz von OSS beherrschbar, auch wenn sich zahlreiche formale Aufgaben bei der Weitergabe von Urhebervermerken, Lizenztexten, Ausschlussklauseln und anderen Informationen stellen, die sowohl Vertriebsdokumente als auch den Quellcode selbst betreffen können[48].

Im Übrigen sollte keine Angst vor dem Einsatz von OSS auch unter strengen Lizenzen bestehen (soweit diese untereinander kompatibel sind), wenn das Geschäftsmodell des Unternehmens nicht von der Vergütung des Gesamtprodukts abhängt. Dies kann etwa bei Software der Fall sein, die als solche kein relevantes Know-How enthält und lediglich als Umgebung für zu lizenzierende Inhalte genutzt wird oder bei Software, die nur zum Betrieb von zu vertreibender Hardware in „embedded systems“[49] dient. Es ist also abschließend auch zu überlegen, ob die Zurverfügungstellung eines Produkts mit offenem Quellcode überhaupt als Problem gesehen werden muss. Es bleibt zu hoffen, dass die Bemühungen der OSS-Community, die offenen Rechtsfragen durch neue, einfacher anzuwendende Regelungen zu klären

[44] https://www.fossology.org/.

[45] scancode-toolkit.readthedocs.io.

[46] Chiampi Ohly, Softwarerecht: Von der Entwicklung zum Export. Probleme und Lösungen nach deutschem und amerikanischem Recht, 3. A., Frankfurt a.M. 2020, 409; siehe auch zum Lizenzmanagement Mantz in Kilian/Heussen, Computerrechts-Handbuch, Werkstand: 35. EL Juni 2020, 32.6 Open Source Software, Rn. 82.

[47] Abrufbar unter https://raw.githubusercontent.com/OpenChain-Project/Reference-Material/master/Suppliers/Leaflet/Official/2.0/en/supplier-leaflet-en.pdf, siehe einführend auch den Bitkom Open Source Leitfaden unter https://www.bitkom.org/sites/default/files/file/import/FirstSpirit-1498131485664160229-OSS-Open-Source-Software.pdf.

[48] Zur Umsetzung im Einzelnen: Witzel: Vertrieb von Open Source Software in „embedded systems“ – Welche besonderen Lizenzpflichten sind einzuhalten und wie können diese praktikabel umgesetzt werden?, ZVertriebsR 2018, 360.

[49] Dazu Schöttle, Open Source Compliance bei Embedded Systems, in Taeger (Hrsg.), Recht 4.0, Innovationen aus den rechtswissenschaftlichen Laboren Tagungsband DSRI-Herbstakademie 2017, 463; Witzel, 360.

und den Einsatz der OSS zu erleichtern[50], in der Zukunft noch stärker ihre Niederschlag in neuen Lizenzen finden.

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access Dieser Artikel wird unter der Creative Commons Namensnennung 4.0 International Lizenz veröffentlicht, welche die Nutzung, Vervielfältigung, Bearbeitung, Verbreitung und Wiedergabe in jeglichem Medium und Format erlaubt, sofern Sie den/die ursprünglichen Autor(en) und die Quelle ordnungsgemäß nennen, einen Link zur Creative Commons Lizenz beifügen und angeben, ob Änderungen vorgenommen wurden.

Die in diesem Artikel enthaltenen Bilder und sonstiges Drittmaterial unterliegen ebenfalls der genannten Creative Commons Lizenz, sofern sich aus der Abbildungslegende nichts anderes ergibt. Sofern das betreffende Material nicht unter der genannten Creative Commons Lizenz steht und die betreffende Handlung nicht nach gesetzlichen Vorschriften erlaubt ist, ist für die oben aufgeführten Weiterverwendungen des Materials die Einwilligung des jeweiligen Rechteinhabers einzuholen.

Weitere Details zur Lizenz entnehmen Sie bitte der Lizenzinformation auf http://creativecommons.org/licenses/by/4.0/deed.de.

Thomas Wilmer Prof. Dr. Thomas Wilmer, Geschäftsführender Direktor des Instituts für Informationsrecht der Hochschule Darmstadt.

[50] Jaeger/Metzger, 1D, Rn. 29.