

Juristisches IT-Projektmanagement  
Wintersemester 2017/18

Dr. Frank Sarre

Freie Software im kommerziellen Einsatz

Diana Irscher

14. Januar 2018



Diese Arbeit behandelt allgemein freie Softwarelizenzen mit Fokus auf GPL (GNU General Public License). Zu Beginn werden die Begriffe Freie Software und Open Source Software definiert. Es wird zudem auf die Problematik mit den unterschiedlichen Ansichten der Free Software Foundation und der Open Source Initiative eingegangen. Weiterhin wird behandelt, welche Lizenz-Modelle es gibt. Genauer wird dabei auf die GPL Bezug genommen. Im darauffolgenden Abschnitt wird analysieren, ob und wie Open Source- und Freie Software für kommerzielle Zwecke genutzt werden kann, und was dabei beachtet werden muss. Zum Schluss wird die Rechtsprechung in Deutschland zu freier Software / Open Source Software betrachten (dabei mit starker Einschränkung auf GPL).

## 1 Freie Software

Der Begriff *Freie Software* hat sich um 1983 entwickelt [1]. Obwohl es sich um eine relativ neue Thematik handelt, hat die *Freie Software* bis heute um ihre Bedeutung und Verwendung einen großen Wandel erfahren. So gibt es von unterschiedlichen Institutionen abweichende Definitionen und Benennungen. Im Folgenden werden diese einzeln betrachtet und verglichen. Auf Vollständigkeit wird an dieser Stelle nicht gesetzt. Es handelt sich vielmehr um einen groben Überblick zur Einführung in die Thematik.

### 1.1 GNU/Linux: Freie Software

Mit dem GNU Project (GNU) unter Richard Stallman, dem Gründer des Projekts, bildete sich der Begriff *Freie Software* erstmals [2]. Die Bezeichnung GNU ist ein rekursives Akronym und steht für *GNU's not unix*. Dabei handelt es sich um ein freies, unix-ähnliches Betriebssystem, welches die Nutzung eines Rechners ohne die zwingenden Vorgaben anderer Betriebssysteme [3] möglich macht. Aufgrund dessen wird auch oft die Bezeichnung GNU/Linux verwendet. Das Projekt ist der Free Software Foundation (FSF) zugeordnet, welche ebenfalls von Stallman gegründet wurde [4]. Die FSF fördert *Freie Software* und finanziert diese gemeinnützige Tätigkeit [5].

GNU definiert *Freie Software* sehr ausführlich [6]. Nach dieser Definition gibt es vier wesentliche Freiheiten, die *Freie Software* ausmachen.

**Freiheit 0** Freiheit, das Programm beliebig auszuführen

**Freiheit 1** Freiheit, das Programm analysieren zu können und ggf. an die eigenen Wünsche anzupassen (die Verfügbarkeit des Quellcodes ist eine Voraussetzung dafür).

**Freiheit 2** Freiheit, Kopien verteilen zu können (Hilfe in der Community umsetzen)

**Freiheit 3** Freiheit, Kopien der modifizierten Form zu verteilen und der Community die Möglichkeit geben, von den Änderungen zu profitieren.

Treffen alle vier Freiheiten auf ein Programm bzw. eine Software zu, handelt es sich um *Freie Software*. Anhand dieser Definition kann man gut erkennen, worum es Stallman bzw. GNU geht: Das Miteinander der Nutzer steht im Fokus. Das Prinzip der *Freien Software* basiert auf einer Ethik des Teilens. Unfreie Software betrachtet GNU als unethisch [6]. Die Definition der FSF mit ihren vier Freiheiten ist die konkreteste aller Definition zum Begriff *Freie Software* [7].

### 1.1.1 GNU/Linux: Das Debian Projekt

Das Debian Projekt ist ein Zusammenschluss mehrerer Entwickler, die es sich zum Ziel gesetzt haben, ein freies Betriebssystem zu entwickeln – Debian [8]. Das Debian System nutzt derzeit den Linux Kernel<sup>1</sup> oder den FreeBSD Kernel<sup>2</sup>. Es wird auch daran gearbeitet, Debian für andere Kernel zur Verfügung zu stellen, wie z. B. für Hurd, welche von GNU entwickelt wird.

Das Debian Projekt [9] führt einen *Debian Social Contract* [10], welcher besagt, dass das Projekt unter den Debian Free Software Guidelines (DFSG) [11] geführt wird. Anhand der DFSG wird laut dem *Debian Social Contract* festgestellt, ob es sich bei einer Software um freie Software handelt. Die DFSG basieren auf den vier Freiheiten, welche von GNU aufgestellt wurden (siehe 1.1, S. 3), d. h., in zehn Punkten wird genau beschrieben – auf Debian Pakete angepasst –, was erlaubt bzw. vorgesehen ist. Grob zusammengefasst beinhaltet das die Verbreitung und Modifizierung der Software. Programme dürfen mit anderer Software weiterverwendet werden, ohne dass Lizenzgebühren anfallen. Der Quellcode der Programme muss jeden Nutzer zur Verfügung stehen, um ein Programm ggf. anpassen (und anschließend verbreiten) zu können. Sollte eine modifizierte Version eines Programmes veröffentlicht werden, dann muss dieses Programm unter der gleichen Lizenz wie die Vorgängersoftware vertrieben werden. Die Lizenz darf weiterhin keine Personen oder Personengruppen diskriminieren. Die Lizenz darf niemand daran hindern, das Programm in jedem Bereich einsetzen zu können. Das schließt auch Aufgabengebiete in der Rüstungsindustrie oder genetischer Forschung nicht aus. Es ist weiterhin ausdrücklich betont, dass die Lizenz nicht ausschließlich für das Debian Projekt gilt, sondern auch für jede einzelne Komponente der Debian Software. Wird ein Paket bzw. Programm ohne das Debian System vertrieben, gelten die gleichen Lizenzbedingungen wie für das gesamte Debian System. Die Lizenz darf weiterhin keine andere Software ausschließen, d. h. Programme unter dieser Lizenz können mit Programmen mit anderen Lizenzbedingungen eingesetzt werden (siehe 3.2.1, S. 9). Für das Debian System (und vielen anderen Linux Distributio-

---

<sup>1</sup><https://www.kernel.org>

<sup>2</sup><https://www.freebsd.org>

nen, die auf GNU/Linux basieren) besteht die Möglichkeit, neben freien Treibern für die Hardwarekomponenten in einem Rechner wie z. B. der Grafikkarte auch solche unfreien Treiber zu verwenden wie sie der Hardwarehersteller anbietet. Unter dem letzten Punkt sind drei Beispiele solcher Lizenzen aufgelistet, GPL, BSD und Artistic. Welche das sind und was genau diese Lizenzen beinhalten, wird in Abschnitt 2 ausführlicher behandelt.

Zum Schluss ist unter [11] angemerkt, dass die DFSG wenig später verwendet wurden, um den Begriff *Open Source Software* zu definieren. Worum es sich dabei handelt und worin der Unterschied zu *Freier Software* liegt, wird im Folgenden analysiert.

## 1.2 Open Source oder Freie Software?

Die Begriffe *Freie Software* und *Open Source Software* werden oft als Synonyme verwendet. Dass es sich dabei allerdings wirklich um ein und dasselbe handelt, ist z. T. umstritten. Warum überhaupt der Begriff *Open Source Software* entstanden ist und worüber man heute noch diskutiert, ist im folgenden Abschnitt zusammengetragen.

### 1.2.1 Die Open Source Initiative

Der Begriff *Open Source Software* ist erst 1998 entstanden. Erich S. Raymond schrieb ein Paper über die Vorteile der Entwicklung mit quelloffener Software [12]. Große Konzerne wie Netscape, IBM und HP reagierten auf die Veränderung auf den Märkten und der wachsenden Kongruenz aus dem Internet, indem sie Teile aus dem Quellcode ihrer Produkte veröffentlichten [13]. Im Zuge dessen befürchtete man, der Begriff *Freie Software* sei für den Wettbewerb mit proprietären Produkten nicht gewachsen. Er sei nicht geschäftsfreundlich, zu ideologisch und mit den ethnischen und sozialen Aspekten der FSF überladen [14]. So wurde 1998 der neue Begriff *Open Source Software* in Umlauf gebracht. Die Open Source Initiative (OSI) definierte unter Bruce Perens und S. Raymond diese Bezeichnung. Die Definition entstand aus den Debian Social Contract und den DFSG (siehe 1.1.1). Die für Debian spezifischen Formulierungen wurden entfernt und der Begriff *Freie Software* durch *Open Source Software* ersetzt. Die Open Source Definition (OSD) beinhaltet wie die DFSG zehn Punkte und stimmt mit den Definitionen der *Freien Software* weitgehend überein.

### 1.2.2 Zwiegespaltene Meinungen

Die OSI und der Begriff *Open Source* findet nicht überall guten Anklang. Richard Stallman findet [15], der Begriff sei mit philosophischen Ansichten in Verbindung gebracht worden, der sich von denen der FSF unterscheidet. Viele würden darin eine Marketing-Kampagne sehen, und lehnen die ethnischen und sozialen Werte der Freien Software Bewegung ab. Der Begriff *Open Source* sei nur mit praktischen Werten verbunden. Die beiden Begriffe

beschreiben zwar die selbe Kategorie Software, die Absichten basieren jedoch auf sehr verschiedenen Werten. Er bemerkt weiterhin, dass Open Source nur einen Entwicklungsmethode sei, Freie Software aber eine soziale Bewegung.

Die Free Software Foundation Europe (FSFE) teilt diese Ansichten [7]. Sie gibt der OSI die Schuld daran, dass noch mehr Verwirrung herrscht. *Freie Software* würde mit *Open Source Software* verwechselt werden. Die proprietären Anbieter würden die Nutzer täuschen, indem sie Teile des Quellcodes ihrer Produkte veröffentlichen. Eine Weitergabe oder gar modifizierte Weitergabe sei im Gegensatz zu GPL meist nicht erlaubt.

### 1.3 Open Source und Freie Software

Die Auseinandersetzungen spalten die Lager [16]. Die FSF kämpft für ihre Ideologie und Prinzipien. Die OSI teilt diese Wertevorstellungen nicht. Oft ist *Open Source Software* auch mit einer Lizenz für *Freie Software* vergleichbar, wie z. B. GNU General Public Licence. In einigen Fällen allerdings auch nicht. Trotz aller Differenzen kommt der Begriff *Open Source* gut an und ist heutzutage sehr bekannt. Nahezu überall wird von *Open Source* gesprochen, auch wenn es sich vielmehr um *Freie Software* handelt. Das betrifft auch neue und aktuelle Linux Distributionen, wie z. B. Manjaro<sup>3</sup>. Auch hier spricht man von einem Open Source Betriebssystem.

Wolle man einen neutralen Begriff für *Freie Software* und *Open Source Software* wählen, so könne man laut Stallman Free Libre Open Source Software (FLOSS) oder Free Open Source Software (FOSS) verwenden [15]. Eine Definition für beide zu finden, ist aufgrund der unterschiedlichen Ansichten schwierig. Unter [17] ist eine allgemeine Definition aufgeführt:

Briefly, OSS/FS programs are programs whose licenses give users the freedom to run the program for any purpose, to study and modify the program, and to redistribute copies of either the original or modified program (without having to pay royalties to previous developers).

## 2 Open Source Lizenzen

In diesem Abschnitt werden die unterschiedliche Lizenzmodelle betrachtet und anhand ihrer Eigenschaften kategorisiert. Die Einteilung der unterschiedlichen Lizenzen ist eine Anlehnung an die Gliederung der vorgestellten Lizenzen in [16] und auch [18]. Allerdings werden hier nur die wichtigsten Modell genannt. Es gibt neben diesen noch weitere, wie z. B. *Lizenzen mit Wahlmöglichkeiten* und *Lizenzen mit Sonderrechten*. Auf diese wird in dieser Arbeit nicht weiter eingegangen.

---

<sup>3</sup><https://manjaro.org/>

Alle Open Source Lizenzen müssen die allgemein bestimmten Definitionsmerkmale erfüllen (siehe unter 1.1 und 1.1.1). Dennoch unterschieden sie sich zum Teil erheblich in ihrer Art und Verwendung. Diese großen Unterschiede haben dazu beigetragen, dass es für Open Source Lizenzen verschiedene Kategorien gibt. Im Folgenden sind die drei wichtigsten Kategorien näher beschreiben.

## 2.1 Lizenzen mit strenger Copyleft-Klausel

Das wesentliche Merkmal einer strengen Open Source Lizenz ist eine Schutzklausel, welche sicherstellt, dass Weiterentwicklungen der Software unter der gleichen Lizenz weitergegeben werden wie ihre Vorgängerversion [16]. Die bekannteste Lizenz dieser Kategorie ist die GNU General Public Licence (GPL). Es gibt mittlerweile drei Versionen. Die erste wurde für das GNU Project 1989 von Stallman geschrieben (siehe auch 1.1, S. 3). Die FSF empfiehlt, die aktuellste Version der Lizenz zu verwenden, Version 3 [19]. Die weltweit verbreitetste ist die zweite Version. Vermutlich liegt die Ursache dessen in dem großen Zeitabstand der Veröffentlichungen der Lizenzen (Version 2 wurde 1991 veröffentlicht, Version 3 erst 2007). Aufgrund der weiten Verbreitung von GPLv2 wird in dieser Arbeit vorwiegend mit dieser Version gearbeitet. Die Unterschiede zu Version 3 liegen im Detail und werden an dieser Stelle nicht weiter aufgeführt. Es sei noch angemerkt, dass die Versionen kompatibel zueinander sind [20].

Die GNU General Public Licence hat diese sogenannte *Copyleft*-Klausel als zentrale Vorschrift [21]:

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.

Software mit einer solchen Klausel in der Lizenz wird als Copyleft Software bezeichnet. Neben der GPL fallen unter diese Kategorie noch die GNU Affero General Public License (AGPL), die Eclipse Public License (EPL) und die Deutsche Freie Software Lizenz (d-fsl) [16].

## 2.2 Lizenzen mit beschränkter Copyleft-Klausel

Lizenzen mit beschränktem Copyleft erlauben im weiteren Umfang die Kombination der Ursprungssoftware mit Erweiterungen unter abweichenden Lizenzbedingungen [16]. Das macht eine einfache Kombination mit anderen, proprietären Lizenzbedingungen möglich. Bekannte Vertreter dieser Lizenzkategorie sind die GNU Lesser General Public License (LGPL) und die Mozilla Public License (MPL). Wie in Abschnitt 4.3, S. 13 zu sehen, ist die LGPL mit den Versionen der GPL nicht kompatibel.

## 2.3 Lizenzen ohne Copyleft-Klausel

Ein bekanntes Beispiel für eine Non-Copyleft-Lizenz ist die Berkeley Software Distribution (BSD). Die BSD verlangt bei Weitergabe den Lizenztext, den Haftungsausschluss sowie einen Urheberrechtsvermerk [18]. Im Gegensatz zur GPL untersagt die BSD bei der Verbreitung einer veränderten Version die Angabe der Urheber der ursprünglichen Version, und eine Werbung für die neuere Version zu vermeiden [16]. Der Schwerpunkt dieser Lizenz liegt in der Einräumung der Nutzungsrechte. Mangels der Copyleft-Klausel führt es bei Verwendung dieser Lizenz zu weniger rechtlichen Problemen. Allerdings führt es bei dieser Kategorie der Software-Lizenzen oft zu Kompatibilitätsproblemen mit anderen Kategorien, insbesondere mit den Copyleft-Lizenzen. Eine gute Auflistung von Lizenzen, welche mit GPL kompatibel sind, ist unter [20] zu finden. Weitere Beispiele für Lizenzen der Non-Copyleft-Kategorie sind die Apache (Software) License und die MIT License (MIT) [16].

Für alle folgenden Abschnitte sei an dieser Stelle angemerkt, dass diese Arbeit sich auf den Begriff *Freie Software* konzentriert, d. h. es wird mit der GNU General Public License gearbeitet. Die anderen Lizenz-Modelle werden nicht weiter betrachtet und wurden nur kurz definiert.

## 3 Open Source- und freie Software für kommerzielle Zwecke

Dass *Freie Software* etwas kosten kann, scheint auf den ersten Blick nicht logisch, schließlich steckt doch das Wort *Frei* im Begriff. Und viele bekannte Programme, welche unter der GPL lizenziert sind, können kostenlos bezogen werden. Warum allerdings Freie Software sehr wohl einen Preis haben kann, wird im folgenden Absatz definiert.

### 3.1 Freie Software und kommerzieller Einsatz - wie passt das zusammen?

Wie oben bereits erwähnt, betrachtet GNU unfreie Software als unethisch. Das bedeutet allerdings nicht, dass Freie Software nicht "nicht kommerziell" sein muss. In [18] steht

[Die] Gemeinsamkeit aller Softwaremodelle ist die kostenlose Überlassung [des Programms].

Unter anderen genau diese Annahme führte zu der OSI, da oft angenommen wurde, das Wort *free* beziehe sich auf den Preis. Gemeint ist mit *free* allerdings die Freiheit der Nutzung. Die Nutzung umfasst die Ausführung, die Untersuchung, die Modifizierung und Verbreitung mit oder ohne Änderung des Programms. Im Gegenteil – *Freie Software* muss für kommerzielle Nutzung, kommerzielle Entwicklung und kommerziellen Vertrieb verfügbar sein. [2].

Free software is a matter of liberty, not price. To understand the concept, you should think of “free” as in “free speech”, not as in “free beer”.

GNU erachtet kostenlose kommerzielle Software als sehr wichtig. Die FSF findet sehr wohl, dass *Freie Software* verkauft werden kann und auch sollte. Denn wenn sich ein Entwickler entscheidet, seine Arbeit unter der GPL zu veröffentlichen, kann er eine Aufwandsentschädigung verlangen. Ob man die Kopien gegen einen Aufpreis oder kostenlos erhalten hat, spielt keine Rolle. Frei ist die Software im Sinne der Freiheit, die Software zu kopieren, zu verändern und zu verbreiten, wenn man so will auch zu verkaufen. In GPLv2 ist das unter §1 geregelt:

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Es dürfen allerdings keine Lizenzgebühren erhoben werden, denn diese verwehren den freien Zugang zur Software [16], dies ist unter §2b GPL geregelt.

### 3.2 Arbeiten mit Freier Software aus der Perspektive eines Softwarehauses

Der Unterschied zwischen freier und “herkömmlich“ lizenzierter Software liegt in der Einräumung urheberrechtlicher Nutzungsrechte [16]. Dies beinhaltet das freie Kopieren, Bearbeiten, Untersuchen und Verbreiten der Software. Durch die *urheberrechtlichen Ausschließlichkeitsrechte* werden die Nutzungsmöglichkeiten der freien Software eröffnet und auch gesichert. Damit können Softwarehäuser eigene (kommerzielle) Entwicklungen mit Freier Software anbieten. Dabei gibt es aus Sicht des Softwareherstellers jedoch einige Punkte zu beachten. Welche das sind, wird in den folgenden beiden Abschnitten behandelt. An dieser Stelle wird speziell auf die GPL Version 2 eingegangen, da diese die meist verbreitetste Lizenz ist.

#### 3.2.1 Wann muss Software unter der GNU General Public Licence vertrieben werden?

Ob eigene Entwicklungen eines Softwareherstellers in Zusammenhang mit *Freier Software* unter die GPL fallen oder nicht, hängt davon ab, wie die Programme zusammenarbeiten. Laut §2 GPLv2 muss ein (von einem unter GPL lizenziertes Programm) abgeleitetes Programm ebenfalls unter der GPL lizenziert werden. Was genau ein abgeleitetes Programm bzw. Werk ist, lässt sich jedoch nicht immer so einfach entscheiden [22]. Allerdings kann man anhand einiger Beispiele einordnen, ob es sich um ein abgeleitetes Werk handelt.

Wird ein unter GPL lizenziertes Programm dahingehend verändert, dass Erweiterungen in das Programm eingefügt werden, oder ganzen Programmteile ersetzt werden, handelt es sich eindeutig um ein abgeleitetes Werk, welche nach §2b GPL unter die GPL fällt.

Die gemeinsame Verteilung von bspw. zwei Programmen auf einem Datenträger wie einen USB-Stick oder einem CD-Rom erfordert keine GPL, wenn es sich um zwei eigenständige Programme handelt und eines der beiden Programme unter die GPL fällt. Das eigene Programm kann über eine beliebige Lizenz vertrieben werden, da es sich nicht um ein abgeleitetes Werk handelt.

Bei Anwendungsprogrammen für Linux ist es noch entscheidbar, unter welcher Lizenz das entwickelte Programm fällt. Solang das Programm ausschließlich über Systemaufrufen mit dem Linux-Kernel läuft, muss das Programm nicht unter die GPL fallen. Ein gutes Beispiel dafür ist die Möglichkeit vieler Linux-Distributionen, unterschiedliche Treiber für die Hardware zu verwenden. So gibt es für dedizierte Grafikkarten in heute gängigen Laptops die Möglichkeit, *Freie Software* für die Treiber zu verwenden oder bereitgestellte Treiber vom Hersteller. Für Nvidia-Grafikkarten verwendet man entweder den Nouveau-Treiber<sup>4</sup>, welcher unter der GPL vertrieben wird, oder den Treiber des Herstellers selbst<sup>5</sup>, welcher unter einer anderen Lizenz steht.

Die Verwendung von Softwaretools, Editoren und Compilern benötigt keine Lizenz, solange kein Quellcode der *Freien Software* in die Entwicklung einfließt. Ist das nicht der Fall, handelt es sich um zwei eigenständige Programme. Ein gutes Beispiel dafür ist das Versionsverwaltungstool *git*<sup>6</sup>, welches unter GPLv2 lizenziert ist.

Für die Entwicklung von Kernelmodulen selbst ist nicht klar definiert, unter welcher Lizenz diese veröffentlicht werden dürfen oder müssen. In Linux werden sogenannte *binary only modules* geduldet. Es gibt jedoch keine einheitliche Aussage dazu, weswegen die lizenzrechtliche Zulässigkeit nicht klar ist. Hier gilt es, besondere Vorsicht walten zu lassen und im Zweifelsfall eine genau Prüfung und Beratung zu erzielen.

Für die Verwendung von Programmbibliotheken gibt es keine eindeutige Abgrenzung. Wird eine Programmbibliothek, welche unter GPL steht, für ein eigenes Programm entwickelt, oder umgekehrt, in beiden Fälle ist nicht zu unterscheiden, ob es sich um ein abgeleitetes Werk handelt oder nicht. Dazu gibt es leider keine Rechtsprechung [18].

Die Verwendung einer Schnittstelle gibt nicht vor, ob das mit der Schnittstelle kommunizierende Programm unter die GPL fallen muss. Findet lediglich ein Informationsaustausch statt, muss das Programm nicht der GPL unterliegen. Das können bspw. Pipes oder Sockets sein. Bei Verbindungen, welche über die Kommunikation hinausgehen, kann allerdings ein abgeleitetes Werk vorliegen, welches unter der GPL vertrieben werden muss.

Anhand dieser groben Einteilung möglicher Entscheidungsfälle kann ein Softwarehaus einschätzen, unter welcher Lizenz die entwickelte Software ggf. vertrieben werden kann oder muss. In jedem Fall sollte beachtet werden, dass im Falle einer nötigen GPL auch die Bedingungen dran eingehalten werden müssen. Welche das sind, wird im folgenden

---

<sup>4</sup>[https://www.archlinux.org/packages/extra/x86\\_64/xf86-video-nouveau/](https://www.archlinux.org/packages/extra/x86_64/xf86-video-nouveau/)

<sup>5</sup>[https://www.archlinux.org/packages/extra/x86\\_64/nvidia/](https://www.archlinux.org/packages/extra/x86_64/nvidia/)

<sup>6</sup><https://git-scm.com/about/free-and-open-source>

Absatz erläutert.

### 3.2.2 Bedingungen bei der Verbreitung von Software unter der GNU General Public Licence

Wird eine Software unverändert unter der GPL verbreitet, müssen dabei einige wichtige Bedingungen beachtet werden [23]. Sollte eine dieser Bedingungen nicht eingehalten werden, wird die Lizenz verletzt. Folgende Bedingungen sind bei einer Verbreitung einzuhalten:

- Wird ein Programm unter der GPL verbreitet, ist laut §1 GPLv2 der Lizenztext mitzuliefern.
- In jedem veröffentlichten Programm muss laut §1 GPLv2 ein Vermerk auf das Copyright deutlich sichtbar vermerkt sein, sowie auch ein Haftungsausschluss. Alte Copyright-Vermerke und Haftungsausschlüsse dürfen nicht entfernt werden.
- Der Quelltext des Programms muss laut §3 GPLv2 jedermann zugänglich gemacht werden. An die Form der Bereitstellung sind einige Vorgaben zu beachten.
- Es dürfen laut §6 GPLv2 dem Nutzer keine zusätzlichen Pflichten neben denen der GPL auferlegt werden.

Ist die Software, welche verbreitet wird, zusätzlich auch verändert, kommen einige Bedingungen hinzu [24]:

- Laut §2a GPLv2 muss die Software einen Hinweis auf die Änderungen enthalten, damit für den Nutzer ersichtlich ist, dass diese enthalten sind und gleichzeitig die Autoren vorhergehender Versionen mit ihrer Arbeit geschützt sind.
- Sollte das Programm interaktiv mit Kommandozeilenbefehlen arbeiten, muss beim Start laut §2c GPLv2 ein entsprechender Urheberrechtshinweis und eine Information ob, eine Garantie besteht oder nicht, angezeigt werden. Ferner ist der Benutzer darüber zu unterrichten, dass er das Programm unter diesen Bedingungen weiter verteilen darf und wie er eine Kopie dieser Lizenz einsehen kann.

Sollten die Pflichten der GPL nicht eingehalten werden, sind die Lizenzbedingungen verletzt. Was passiert, wenn dies der Fall ist, zeigen einige Urteile hier in Deutschland, welche im folgenden Absatz behandelt werden. Davor werden einige Bedingungen der GPL betrachtet, die mit der deutschen Rechtsprechung nicht zusammenpassen.

## 4 Rechtsprechung in Deutschland zu Freier Software unter der GNU General Public Licence

In diesem Abschnitt werden die rechtlichen Bedingungen der GPL in Hinblick auf die Rechtssprechung in Deutschland näher betrachtet. Zu diesem Thema ist einiges an Literatur in Deutschland zu finden, unter anderem seien hier [16] und [25] genannt, welche für die Recherche dieser Arbeit vorwiegend verwendet wurden. Im Folgenden ist der Umfang der vielen Punkte zu diesem Thema stark eingeschränkt. Es wird lediglich das Urheberrecht und der Haftungsausschluss betrachtet, sowie drei bekanntere Urteil in der deutschen Rechtssprechung.

### 4.1 Wer ist Urheber der Freien Software?

Freie Software ist urheberrechtlich geschützt. Jeder Entwickler, welcher das Programm geschrieben oder verändert hat, ist auch Urheber. Der größte Anreiz, Software unter der GPL zu veröffentlichen, besteht laut [16] in der Aussicht, sich damit einen Namen machen zu können. Gerade in der Softwareentwicklung kann man seine Arbeit so gut präsentieren. Für die Entwicklungen, und seien das auch Änderungen an einem bestehenden Programm, ermöglicht die GPL mit der Pflicht des Urheberhinweises, dass die Arbeit auf den Entwickler zurückgeführt und angerechnet werden kann [26]. Im Vorwort der GPL ist dazu schon erwähnt [27]:

[...] for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

#### 4.1.1 Das Urheberpersönlichkeitsrecht und die Freiheit der Freien Software

Wie in 1.1.1, S. 4 beschreiben, ist in den DFSG festgesetzt, dass niemand daran gehindert werden kann, das Programm in jedem Bereich einsetzen zu können. Das umfasst auch die Weiterentwicklung der Software. Die Befugnis der Nutzer, *Freie Software* beliebig zu ändern und verbreiten zu können, führt allerdings zu einer Problemstellung mit dem Urheberrechtsgesetz (UrhG) in Deutschland [16]. Zwar ist dem Urheber nach §31 Abs. 3 UrhG freigestellt, Nutzungsrechte einzuräumen und Veränderungen seines Programmes zu gestatten. Jedoch könnten die Änderungen zur Beeinträchtigung des Urheberpersönlichkeitsrecht führen. Hier stößt die (Vertrags-)Freiheit, wie sie in der GPL beschrieben ist, an ihre Grenzen, denn gemäß §14 UrhG hat der Urheber die Befugnis, Beeinträchtigungen oder Entstellungen seines Werkes zu verbieten, welche die persönlichen oder geistigen

Interessen am Werk gefährden. Das könnte zu Problemen führen. Die GPL schreibt dazu allerdings in Ziffer 7 Abs. 1, S. 1 und 2, dass im Falle einer Einschränkung durch Rechtssprechung oder ähnliches das Programm nicht veröffentlicht werden darf [27]:

If, as a consequence of a court judgment or allegation of patent infringement or for any other reason [...], conditions are imposed on you [...] that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all.

## 4.2 Haftungsausschluss in Deutschland

Laut Ziffer 1 GPLv2 ist in einer modifizierten Kopie eine GPL-lizenzierten Programms ein Hinweis auf Haftungsausschluss zu vermerken. Allerdings kommt diesem in Deutschland keine rechtliche Bedeutung zu. Laut [16], Rn. 219 ist es grundsätzlich möglich, die gesetzlichen Gewährleistungsvorschriften bis auf Arglist vertraglich abdingbar zu machen, wie in §§. 444 BGB, 639 BGB geregelt. Dies gilt allerdings nicht, wenn die entsprechende Klausel eine Allgemeine Geschäftsbedingung (AGB) im Sinne von §305 Abs. 1 BGB ist. Open Source Lizenzen werden nach allgemeiner Auffassung allerdings als AGBs angesehen [16], Rn. 179. Ein pauschaler Haftungsausschluss verstößt in Deutschland gegen die AGB-Vorschriften. In anderen Rechtsordnungen kann dieser Haftungsausschluss allerdings von Bedeutung sein, weshalb es grundsätzlich nicht unnützlich ist, diesen in die Kopie aufzunehmen. Eine Lösung für diese Problemstellung ist laut [25] die von [16] R. 205 vertretene Schenkungslösung. Diese gibt nach BGB eine eindeutige Regelung der Gewährleistung für Open Source Software. Allerdings setzt dies eine kostenlose Überlassung voraus.

## 4.3 Aktuelle Rechtssprechung

Im Jahr 2004 wurde am LG München unter dem Zeichen Az.21 0 6123/03 die GPL von den Richtern grundsätzlich anerkannt. Verhandelt wurde ein juristischer Streit zwischen dem Routerhersteller Sitecom und dem Open Source Projekt netfilter/iptables. Das Open Source Projekt hatte eine einstweilige Verfügung gestellt, welche auch bestätigt wurde. Sitecom nutze die Software von netfilters/iptables für ihre Router, ohne die Lizenzbedingungen der GPL einzuhalten. Dieses Urteil ist laut [28] das "weltweit erste Urteil zur Wirksamkeit und Durchsetzbarkeit der GPL".

In [29] ist das Urteil des LG Frankfurt zusammengefasst. Hier handelt es sich um einen ähnlichen Fall. Der Entwickler des Netfilter-Codes klagte gegen D-Link. Diese hatten ein Linux-basiertes NAS-Gerät vertrieben und die Open Source Software verwendet, ohne die GPL zu beachten. Auch in diesem Urteil wurde die grundsätzliche Gültigkeit der GPL von den Richtern bestätigt.

In einem neueren Urteil im Jahr 2017 von LG Köln wurde über die Kompatibilität der GPL und LGPL geurteilt. Das Gericht entschied, dass wie auch unter Ziff. 9 GPLv2 / Ziff. 14 GPLv3 gefordert, Software bei einer “Umlizenzierung nur eine andere Lizenzversion desselben Typs gewählt werden darf, nicht ein anderer Lizenztyp“ [30].

## 5 Fazit

Zusammenfassend lässt sich sagen, dass *Freie Software* (hier ist konkret die GPL gemeint) sehr wohl verkauft und damit kommerziell eingesetzt werden kann. Das bedeutet, als Entwickler kann ich meine Arbeit der Community zur Verfügung stellen und dafür eine Aufwandsentschädigung verlangen. Das ist durchaus rechtens. Es dürfen einzig keine Lizenzgebühren anfallen, und die Lizenz muss beibehalten werden.

*Freie Software* bietet damit eine gute Alternative zu proprietärer Software. Schon heute können wir sehen, wie stark (und nicht mehr verzichtbar) der Markt dafür ist. Gerade in den Zeiten der Digitalisierung sollten wir bedenken, welche Software eingesetzt und verwendet wird. Dieser Bereich bestimmt zunehmend unseren Alltag. In Anbetracht dessen finde ich, dass die ethischen und sozialen Richtlinien der FSF und damit der GPL durchaus wichtig und aktuell sind.

## 6 Abkürzungsverzeichnis

|  |    |
|--|----|
| <b>FOSS</b> Free Open Source Software .....        | 6  |
| <b>FLOSS</b> Free Libre Open Source Software.....  | 6  |
| <b>FSF</b> Free Software Foundation.....           | 3  |
| <b>DFSG</b> Debian Free Software Guidelines.....   | 4  |
| <b>OSI</b> Open Source Initiative.....             | 5  |
| <b>GPL</b> GNU General Public Licence.....         | 7  |
| <b>LGPL</b> GNU Lesser General Public License..... | 7  |
| <b>BSD</b> Berkeley Software Distribution.....     | 8  |
| <b>GNU</b> GNU Project.....                        | 3  |
| <b>OSD</b> Open Source Definition.....             | 5  |
| <b>FSFE</b> Free Software Foundation Europe.....   | 6  |
| <b>UrhG</b> Urheberrechtsgesetz.....               | 12 |
| <b>BGB</b> Bürgerliches Gesetzbuch                 |    |
| <b>AGB</b> Allgemeine Geschäftsbedingung.....      | 13 |
| <b>AGPL</b> GNU Affero General Public License..... | 7  |

|   |   |
|---|---|
| <b>EPL</b> Eclipse Public License .....           | 7 |
| <b>d-fsl</b> Deutsche Freie Software Lizenz ..... | 7 |
| <b>BSD</b> Berkeley Software Distribution .....   | 8 |
| <b>Apache License</b> Apache (Software) License   |   |
| <b>MIT</b> MIT License .....                      | 8 |
| <b>MPL</b> Mozilla Public License .....           | 7 |

## Literatur

- [1] I. Free Software Foundation. (2018) Free software is software that gives you the user the freedom to share, study and modify it. we call this free software because the user is free. Online. Free Software Foundation, Inc. [03.01.2018]. [Online]. Available: <https://www.fsf.org/about/what-is-free-software>
- [2] ——. (2018, Jan.) What is gnu? Online. Free Software Foundation, Inc. [03.01.2018]. [Online]. Available: <https://www.gnu.org/>
- [3] J. Bager. (2015, Jul.) Windows 10: Neue datenschutzbestimmungen – windows wird zur datensammelstelle. Online. [31.12.2017]. [Online]. Available: <https://heise.de/-2765536>
- [4] I. Free Software Foundation. (2018) Staff and board. Online. Free Software Foundation, Inc. [13.01.2018]. [Online]. Available: <https://www.fsf.org/about/staff-and-board>
- [5] ——. (2017) The free software foundation (fsf) is a nonprofit with a worldwide mission to promote computer user freedom. we defend the rights of all software users. Online. Free Software Foundation, Inc. [31.12.2017]. [Online]. Available: <https://www.fsf.org/about/>
- [6] ——. (2017, Dec.) What is free software? Online. [31.12.2017]. [Online]. Available: <https://www.gnu.org/philosophy/free-sw>
- [7] F. S. F. Europe. (2017, Dec.) Wir sprechen von freier software. online. Free Software Foundation Europe. [30.12.2017]. [Online]. Available: <https://fsfe.org/documents/whyfs.de.html>
- [8] I. Software in the Public Interest. (2017, Jun.) What is debian? Online. Software in the Public Interest, Inc. [03.01.2018]. [Online]. Available: <https://www.debian.org/intro/about>
- [9] ——. (2017, Jun.) debian: The universal operating system. Online. Software in the Public Interest, Inc. [01.01.2018]. [Online]. Available: <https://www.debian.org/>
- [10] ——. (2017, Oct.) Debian social contract. Online. Software in the Public Interest, Inc. [02.01.2018]. [Online]. Available: [https://www.debian.org/social\\_contract](https://www.debian.org/social_contract)
- [11] ——. “The debian free software guidelines (dfsg),” Online, Oct. 2017, [31.12.2017]. [Online]. Available: [http://www.debian.org/social\\_contract.en.html#guidelines](http://www.debian.org/social_contract.en.html#guidelines)
- [12] E. S. Raymond, “The cathedral and the bazaar: Musings on linux and open source by an accidental revolutionary,” *Linux Kongress on May 27, 1997 Würzburg*, 1997.

- [13] (1998, Mar.) Netscape sets source code free. Online. CBS Interactive Inc. [13.01.2018]. [Online]. Available: <https://www.cnet.com/news/netscape-sets-source-code-free/>
- [14] A. d. J. Victor van Reijswoud, *Free and Open Source Software for development : exploring expectations, achievements and the future*. Polimetrica, 2008, vol. 113 pages, [01.01.2018]. [Online]. Available: <https://arxiv.org/abs/0808.3717>
- [15] R. Stallman. (2016, Nov.) Why open source misses the point of free software. Online. Free Software Foundation, Inc. [03.01.2018]. [Online]. Available: <https://www.gnu.org/philosophy/open-source-misses-the-point.en.html>
- [16] T. J. und A. Metzger, *Open Source Software: Rechtliche Rahmenbedingungen der Freien Software*. München: C. H. Beck, 2016, vol. 4. Aufl.
- [17] D. A. Wheeler. (2015, Jul.) Why open source software / free software (oss/fs, floss, or foss)? look at the numbers! online. [01.01.2018]. [Online]. Available: [https://www.dwheeler.com/oss\\_fs\\_why.html](https://www.dwheeler.com/oss_fs_why.html)
- [18] C. Weinkraut, "Softwareentwicklung 2017 :open source software als elementare bausteine," in *Recht 4.0 - Innovationen aus dem rechtswissenschaftlichen Laboren: Tagungsband Herbstakademie 2017*, Deutsche Stiftung für Recht und Informatik / Herbstakademie <18., 2017, Heidelberg>. Edewecht ; OIWIR, Oldenburger Verlag für Wirtschaft, Informatik und Recht, 2017, pp. XVII, 940 Seiten.
- [19] I. Free Software Foundation. (2018, Jan.) How to choose a license for your own work. Online. Free Software Foundation, Inc. [14.01.2018]. [Online]. Available: <http://www.gnu.org/licenses/license-recommendations.html#software>
- [20] —. (2018, Jan.) Various licenses and comments about them. Online. Free Software Foundation, Inc. [14.01.2018]. [Online]. Available: <http://www.gnu.org/licenses/license-list.html#GPLCompatibleLicenses>
- [21] —, "Gnu general public license," Online, Nov. 2016, [31.12.2017]. [Online]. Available: <https://www.gnu.org/copyleft/gpl.html>
- [22] D. T. Jaeger. Wann muss eigenentwickelte software unter der gpl lizenziert werden? Online. Institut für Rechtsfragen der Freien und Open Source Software (ifrOSS). [06.01.2018]. [Online]. Available: <http://www.ifross.org/wann-muss-eigenentwickelte-software-unter-gpl-lizenziert-werden>
- [23] T. Jaeger. Welche pflichten bestehen bei der verbreitung von unter der gplv2 lizenzierter software? Online. Institut für Rechtsfragen der Freien und Open Source Software (ifrOSS). [07.01.2018]. [Online]. Available: <http://www.ifross.org/welche-pflichten-bestehen-bei-verbreitung-unter-gplv2-lizenzierter-software>

- [24] ——. Welche pflichten bestehen bei der verbreitung von veränderten versionen unter der gplv2 lizenzierter software? Online. Institut für Rechtsfragen der Freien und Open Source Software (ifrOSS). [07.01.2018]. [Online]. Available: <http://www.ifross.org/welche-pflichten-bestehen-bei-verbreitung-veraenderten-versionen-unter-gplv2-lizenzierter-software>
- [25] O. Koglin, *Opensourcerecht : die urheber- und schuldrechtlichen Beziehungen zwischen Lizenzgeber und Lizenznehmer bei Open Source Software am Beispiel der General Public License (GPL)*. Frankfurt am Main [u.a.] ; Lang, 2007, vol. XIII, 252 S. [Online]. Available: [http://www.ebook.de/de/product/15971688/olaf\\_koglin\\_opensourcerecht.html](http://www.ebook.de/de/product/15971688/olaf_koglin_opensourcerecht.html)
- [26] I. Free Software Foundation. (2018, Jan.) I want to get credit for my work. i want people to know what i wrote. can i still get credit if i use the gpl? Online. Free Software Foundation, Inc. [07.01.2018]. [Online]. Available: <https://www.gnu.org/licenses/gpl-faq.html#IWantCredit>
- [27] ——. (2017, Sep.) Gnu general public license, version 2. Online. Free Software Foundation, Inc. [07.01.2018]. [Online]. Available: <https://www.gnu.org/licenses/old-licenses/gpl-2.0.html>
- [28] H. Bleich. (2004, Jul.) Deutsches gericht bestätigt wirksamkeit der gpl. Online. Heise Medien GmbH & Co. KG. [07.01.2018]. [Online]. Available: <https://www.heise.de/newsticker/meldung/Deutsches-Gericht-bestaetigt-Wirksamkeit-der-GPL-101616.html>
- [29] O. Diedrich. (2006, Sep.) Deutsches gericht bestätigt gültigkeit der gpl. Online. Heise Medien GmbH & Co. KG. [07.01.2018]. [Online]. Available: <https://www.heise.de/newsticker/meldung/Deutsches-Gericht-bestaetigt-Gueltigkeit-der-GPL-164927.html>
- [30] L. M. Keppeler. (2015, Jul.) Lg köln: Zum schadensersatz bei umlizenzierung einer open source-software. Online. Heuking Kühn Lüer Wojtek. [07.01.2018]. [Online]. Available: <https://www.heuking.de/de/news-events/fachbeitraege/lg-koeln-zum-schadensersatz-bei-umlizenzierung-einer-open-source-software.html>