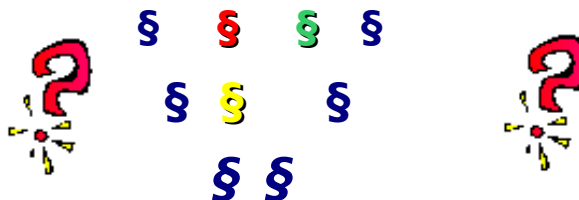


Software-Lizenzen

von Edgar Reis



Im letzten MagDriva hatten wir gesehen, daß Computerprogramme (also Software) urheberrechtlich geschützt sind.

Der Urheber entscheidet darüber, ob und gegebenenfalls wie sein Werk veröffentlicht und verwertet wird.

Im Rahmen seines Verwertungsrechtes bestimmt der Urheber, wer wie und zu welchen Bedingungen seine Schöpfung verbreiten, vervielfältigen, bearbeiten ... darf.

Verstöße gegen diese Bestimmungen werden (teilweise strafrechtlich) sanktioniert.

Das ist das, was landläufig unter dem „**Copyright**“ verstanden wird (wobei wir uns darüber im Klaren sein müssen, daß der Begriff des „Copyright“ dem US-amerikanischen Rechtssystem entstammt und nicht eins zu eins auf unser deutsches Rechtssystem übertragbar ist).

Der Inhaber des Copyrights schränkt bei proprietärer Software den Umgang des Nutzers (Käufers), mit der Software in erheblichem Umfang ein. In der Regel darf der Nutzer das Computerprogramm weder kopieren, verbreiten, verändern bzw. in geänderter Form verbreiten; für jede entsprechende Handlung ist eine besondere Erlaubnis des Rechteinhabers erforderlich, die regelmäßig nur gegen Entgelt erteilt wird. Der Nutzer ist demnach in der Nutzung seiner Software nicht „frei“.

Im Gegensatz hierzu steht „freie Software“ oder „open source software“.

„Freie Software“ wird laut GNU-Projekt wie folgt definiert:

„Freie Software bedeutet die Freiheit des Benutzers, die Software zu benutzen, zu kopieren, sie zu vertreiben, zu studieren, zu verändern und zu verbessern. Genauer gesagt, bezieht sich der Begriff »Freie Software« auf vier Arten von Freiheit, die der Benutzer der Software hat:

- Die Freiheit, das Programm für jeden Zweck zu benutzen (Freiheit 0).
- Die Freiheit, zu verstehen, wie das Programm funktioniert und wie man es für seine Ansprüche anpassen kann (Freiheit 1). Der Zugang zum Quellcode ist dafür Voraussetzung.

- Die Freiheit, Kopien weiterzuverbreiten, so dass man seinem Nächsten weiterhelfen kann (Freiheit 2).
- Die Freiheit, das Programm zu verbessern und die Verbesserungen der Öffentlichkeit zur Verfügung zu stellen, damit die ganze Gemeinschaft davon profitieren kann (Freiheit 3). Der Zugang zum Quellcode ist dafür Voraussetzung.

Ein Programm ist Freie Software,

wenn die Benutzer alle diese Freiheiten haben. Somit sollte man die Freiheit haben, Kopien weiterzuverbreiten, entweder mit oder ohne Veränderungen, gratis oder gegen Erhebung einer Gebühr für die Verbreitung, an jeden und überall. Frei zu sein in diesen Dingen bedeutet (abgesehen von anderen Punkten), dass man nicht nach Erlaubnis fragen oder für eine Erlaubnis bezahlen muss“

(<http://www.gnu.org/philosophy/free-sw.de.html>).

Ähnlich definiert sich „open source“ (vgl.

<http://www.opensource.org/docs/definition.php>).



Der Nutzer freier oder Open Source Software ist also im Umgang mit „seinem“ Computerprogramm frei, er darf es etwa beliebig kopieren, verbreiten, verändern ..., also entsprechend der Definition nutzen.

In dieser Freiheit liegt aber auch eine große Gefahr für die Software und deren Autor; denn wir haben im deutschen Urheberrecht z.B. den § 3 UrhG (gilt gem. § 69a IV UrhG auch für Computerprogramme):

„Übersetzungen und andere Bearbeitungen eines Werkes, die persönliche geistige Schöpfungen des Bearbeiters sind, werden unbeschadet des Urheberrechts am bearbeiteten Werk wie selbständige Werke geschützt. Die nur unwesentliche Bearbeitung eines nicht geschützten Werkes der Musik wird nicht als selbständiges Werk geschützt.“

Wer also freie Software erlaubtermaßen bearbeitet, also verändert, genießt an dem neuen Werk – der geänderten Software – urheberrechtlichen Schutz, er ist also Inhaber des sog. „Copyrights“ mit allen Konsequenzen.

Der Bearbeiter kann somit aus freier Software proprietäre Software schaffen. Die Rechte im Einzelnen ergeben sich dann aus § 69c UrhG.

Es ist einleuchtend, daß dies grundsätzlich nicht im Sinne freier Software sein kann.

Um zu verhindern, daß aus freier Software proprietäre Software wird, werden deshalb die Programme bestimmten **Lizenzen** unterstellt, die gewährleisten sollen, daß das Produkt, z.B. aus einer Bearbeitung, ebenfalls freie Software im Sinne der Definition bleibt.

Diese Handhabung steht in diametralem Gegensatz zu dem Schutzzweck des urheberrechtlichen Verwertungsrechtes, des „Copyrights“, das dem Schutz des geistigen Eigentums des Urhebers dient, während die Lizenzen freier oder Open Source Software gerade die **uneingeschränkte** Nutzung, Verarbeitung und Weiterverbreitung im Sinne haben.

Aus diesem Interessengegensatz ist dann auch der Begriff des „**Copyleft**“ entstanden, ein Kunstwort, das den Gegensatz zu dem einschränkenden „Copyright“ der proprietären Software veranschaulichen soll.

An dieser Stelle sei, um Missverständnissen vorzubeugen, klargestellt, daß auch freie Software einem Copyright unterliegt.

Der erste Autor einer Software genießt zunächst einmal den vollen Schutz des Urheberrechtsgesetzes bzw. der jeweiligen nationalen Rechte. Er entscheidet dann, ob sein Werk frei oder unfrei sein soll. Entscheidet er sich für freie Software, reduziert er seine Verwertungsrechte (sein Copyright) im Sinne der oben angeführten Definition für freie Software.

Nun das Wesentliche: In Ausübung seines Verwertungsrechtes/Copyrights bestimmt der Urheber nun, daß sämtliches Be- und Verarbeiten, Verändern und Ableiten etc. wiederum freie Software im Sinne der Definition ergeben muß. Derjenige, der die Software dann etwa weiterentwickelt, muß seinerseits dafür Sorge tragen, daß sein Arbeitsergebnis ebenfalls freie Software ist, ebenso der nächste Bearbeiter, der nächste ...

Hält sich der z.B. Bearbeiter hieran nicht, begeht er eine Urheberrechtsverletzung genauso wie bei proprietärer Software, mit allen Folgen des Urheberrechtsgesetzes.

Der Nutzer freier Software ist damit ebenso Lizenznehmer wie der User proprietärer Software, allerdings mit dem entscheidenden Unterschied, daß ihm im Rahmen der Copyleft-Lizenzen erhebliche Freiheiten eingeräumt werden.

Wie weit diese Freiheiten gehen, bestimmt sich dann nach der jeweiligen Lizenz, die je nach Intention des Urhebers enger oder weiter gefaßt sein kann immer aber muß das Resultat freie Software im Sinne der Definition sein. So vielfältig die Interessen der Autoren, so vielfältig auch die Ausgestaltung der einzelnen Lizenzen.

Eine (nicht vollständige) Zusammenstellung findet sich hier:

http://www.ifross.de/ifross_html/lizenzcenter.html

An dieser Stelle von Bedeutung sind die dort unter dem Oberbegriff der „Open Source Lizenzen“ aufgeführten, nämlich:

- Lizenzen ohne Copyleft-Effekt
- Lizenzen mit strengem Copyleft-Effekt
- Lizenzen mit beschränktem Copyleft-Effekt
- Lizenzen mit Wahlmöglichkeiten
- Lizenzen mit Sonderrechten

Lizenzen ohne Copyleft-Effekt



Bei diesem Modell hat der Nutzer die am weitestgehenden Rechte. Er entscheidet, welchem Statut etwa abgeleitete Software unterstellt wird; der Lizenznehmer darf sogar proprietäre Software aus vormalig freier Software schaffen.

Als Beispiel sei hier die BSD-Lizenz genannt, ein kleiner Artikel mit deutscher Übersetzung gibt's hier:

<http://de.wikipedia.org/wiki/BSD-Lizenz>

Diese unterliegt grundsätzlich keinerlei Beschränkungen, außer daß ein Copyright-Vermerk angebracht sein muß.

Lizenzen mit strengem Copyleft-Effekt

Charakteristikum dieser Lizenzen ist, daß der Lizenznehmer verpflichtet ist, „von der ursprünglichen Software abgeleitete Werke ebenfalls nur unter den Bedingungen der Ursprungslizenz weiterzuverbreiten“ (vgl.

http://www.ifross.de/ifross_html/lizenzcenter.html.



Hierunter fällt die populärste und am weitverbreitetste Lizenz, die „GNU Public License (GPL)“. Hier die inoffizielle deutsche Übersetzung:

<http://www.gnu.de/gpl-ger.html>

Eine ausführliche Darstellung und Kommentierung bietet:

http://www.ifross.de/ifross_html/gpl-seite.html

Lizenzen mit beschränktem Copyleft-Effekt

„Sofern Modifikationen der Software in eigenen Dateien realisiert werden, können diese Dateien auch unter anderen, z.B. proprietären Lizenzbedingungen weiterverbreitet werden. Damit soll die Kombination von Software unter verschiedenen Lizenztypen erleichtert werden“

http://www.ifross.de/ifross_html/lizenzcenter.html.

Als Beispiel mag die „Mozilla Public License (MPL)“ dienen

<http://www.mozilla.org/MPL/MPL-1.1.html>).

Lizenzen mit Wahlmöglichkeiten

Der Umfang einer Modifikation entscheidet darüber, welche Rechtsfolgen eintreten. Sie enthalten Wahlmöglichkeiten hinsichtlich der Weiterverbreitung von Weiterentwicklungen (vgl.

http://www.ifross.de/ifross_html/lizenzcenter.html).

Beispiel: die „Artistic License v. 2.0“

<http://www.opensource.org/licenses/artistic-license.php>

Lizenzen mit Sonderrechten

„... gewähren den Lizenznehmern zwar alle diejenigen Rechte, die Freie Software ausmachen, sehen aber zugleich besondere Privilegien für den Lizenzgeber bei Weiterentwicklungen durch den Lizenznehmer vor. Diese Lizenzen werden zumeist bei Programmen verwendet, die ursprünglich proprietär vertrieben wurden“

http://www.ifross.de/ifross_html/lizenzcenter.html

Beispiel: „Netscape Public License (NPL)“

<http://www.mozilla.org/MPL/NPL-1.1.html>

Dieser knappe Überblick zeigt, daß auch mit sogenannter „freier Software“ gerade nicht nach Belieben verfahren werden darf.

Der Lizenznehmer darf sich also nur im Rahmen der zugrunde liegenden Lizenz bewegen – nur, aber gerade auch, insoweit ist er „frei“, je nach Copyleft-Effekt mal mehr, mal weniger.

Im Mittelpunkt steht immer die Definition der freien Software; alle Derivate der Ausgangssoftware müssen dieser Definition entsprechen. Und dies wollen die Lizenzen erreichen.

Edgar Reis

