

In der juristischen Diskussion ist durchaus streitig, ob die von der GPL gewählte Konstruktion wirksam vereinbart werden kann. Es wird befürchtet, dass, wenn man jede Lizenz so gestalten könnte, dass bei einem Verstoß sofort die urheberrechtlichen Nutzungsrechte entfallen, ein hohes Missbrauchsrisiko bestehe. Jeder Softwarehersteller könne seine Lizenzbedingungen so gestalten, dass sich der Erwerber einer Software bei einem Verstoß gegen x-beliebige Lizenzbestimmungen wie ein Raubkopierer behandeln lassen müsse.

Diese Befürchtungen greifen allerdings nur zum Teil. Das Gesetz gewährt jedem, der zur Nutzung des Programms berechtigt ist (also jedem, der die Software ordnungsgemäß erworben hat), das Recht zur bestimmungsgemäßen Benutzung (§ 69d UrhG). Dieses gesetzlich gewährte Recht kann auch bei einem Verstoß gegen vertragliche Pflichten nicht »automatisch« entzogen werden. Da bei »proprietärer« Software in der Regel vielfach sowieso keine darüber hinausgehenden Rechte gewährt werden, kann in solchen Fällen auch keine weitere Beschränkung der Nutzerrechte durch eine automatische Rechterückfallklausel erreicht werden.

Im Übrigen ist der Kritik insoweit zuzustimmen, als es jeweils einer Betrachtung der Klauseln im Einzelfall bedürfen wird, um festzustellen, ob diese den Lizenznehmer nicht zu sehr benachteiligen. Im Falle der GPL sprechen dabei die besseren Argumente dafür, dass dies nicht der Fall ist. Die GPL betrifft allein die Einräumung von Entwicklungs- und Vertriebsrechten (also von Rechten, die über die schlichte Programm Benutzung hinausgehen) und diese Einräumung erfolgt auch noch kostenlos. Der Lizenznehmer hat also kein Geld für etwas bezahlt, das ihm dann plötzlich und automatisch entzogen werden soll.

Konsequenzen in der Praxis

Habe ich beim Erwerb von Open Source-Software ein größeres Haftungsrisiko als bei proprietärer Software?

Gerade Unternehmen und Behörden, die über eine Migration ihrer Systeme auf Freie Software nachdenken, sehen sich über kurz oder lang mit den Fragen zur Haftung und Gewährleistung konfrontiert. Wer steht eigentlich dafür ein, wenn Open Source Software fehlerhaft ist oder Schäden an anderen Programmen oder der Hardware verursacht? Wer haftet, wenn die Programme eines Tages plötzlich nicht mehr benutzt werden dürfen, weil irgendwelche Inhaber von Urheberrechten dies verbieten? Sind die Risiken bei Freier Software nicht höher als bei der proprietären Konkurrenz? Man bekommt doch schließlich alles geschenkt, zudem von einer juristisch nicht wirklich greifbaren internationalen Community. Wer soll hier für Schäden haften?

In den typischen Konstellationen sind entsprechende Ängste unberechtigt. Natürlich werden die Rechte an den Programmen stets lizenzgebührenfrei eingeräumt, dies bedeutet aber nicht, dass man für die Distribution als solche keine Vergütung bezahlen würde.

Der normale Nutzer einer GPL-Software erwirbt diese gegen Entgelt im Einzelhandel oder bei einem Dienstleister. Dies gilt sowohl für den Privatanwender als auch für das Unternehmen. Oftmals ist der Erwerb der Software in ein umfassendes Leistungsangebot eingebunden. Man denke etwa an eine Behörde, die sich bei einer Umstellung der Systeme von einem entsprechend spezialisierten Unternehmen unterstützen lässt. Hier bestehen im Grundsatz die gleichen Haftungs- und Gewährleistungsansprüche wie bei proprietären Angeboten (siehe zu den Ansprüchen im Einzelnen Ziffern 11, 12 GPL Rz. 23 ff.).

Dies gilt allerdings nur, solange etwas für die Software bezahlt worden ist. Wer ein Programm kostenlos erhält – sei es direkt von den Urhebern, sei es vom Server eines Dritten, etwa eines Distributors –, muss mit einem Weniger an Haftung leben. Es ist nicht unbillig, dass man eine geschenkte Sache im Wesentlichen so akzeptieren muss, wie sie ist. Nach deutschem Recht besteht hier eine Gewährleistungspflicht des Distributors nur für arglistig verschwiegene Mängel. Freilich sind auch andere Konstellationen praktisch relevant: etwa der Fall der Anpassung einer Standard-Open Source-Software an die Bedürfnisse des Kunden oder die völlige Neuentwicklung. Zu den Einzelheiten und weiteren Fallgestaltungen siehe die Anmerkungen zu den Ziffern 11 und 12 GPL.

Wenn man die Haftung und Gewährleistung bei Freier und proprietärer Software vergleicht, lohnt es sich durchaus, auch einen Blick auf die Praxis proprietärer Anbieter zu werfen. Diese steht nicht im Mittelpunkt dieses Buchs. Es sei gleichwohl darauf hingewiesen, dass es in den letzten 20 Jahren in Deutschland nahezu keine gerichtlich anhängigen Fälle gegeben hat, in denen die Anbieter herkömmlich vertriebener Software für Fehler ihrer Produkte in die Haftung genommen wurden. Dem stehen tausende veröffentlichte Gerichtsentscheidungen zur Mängelhaftung bei Kfz oder Industriemaschinen gegenüber. Was ist der Grund für die fehlende Rechtspraxis? Wohl kaum, dass die Softwareanbieter freiwillig Fehler beseitigt und Schäden ersetzt haben. Der Blick auf die Praxis der Softwarebranche legt eher den Schluss nahe, dass sich die Kunden damit abgefunden haben, dass Programme stets mit Fehlern behaftet sind, und sich für alles weitere auf den kostenpflichtigen Support haben vertrösten lassen. In Märkten, die durch stärkeren Wettbewerb geprägt sind, formulieren Kunden ihre berechtigten Ansprüche offensiver.

Im Vergleich erscheint es nicht berechtigt, von Nachteilen der Freien Software im Bereich der Haftung und Gewährleistung zu sprechen.

Welche Risiken bestehen gegenwärtig auf Grund von Softwarepatenten?

In der gegenwärtigen Diskussion um Softwarepatente dominieren die rechtspolitischen Fragen. Sie sind nicht Thema dieses Buchs, welches in erster Linie praktische Informationen für die Entwicklung, den Vertrieb und die Nutzung Freier Software liefern möchte. Hier sind deswegen nur einige, wenige Anmerkungen möglich.

Es ist mittlerweile auch einer breiteren Öffentlichkeit bekannt, dass die Europäische Union an einer Richtlinie zum Thema Softwarepatente arbeitet. Der ursprüngliche Vorschlag der Kommission vom 20.02.2002 (http://www.europa.eu.int/comm/internal_market/en/ind-

prop/comp/com02-92de.pdf) dürfte bei einer unveränderten Verabschiedung weitere Hürden einreißen auf dem Weg zu einer unbegrenzten Patentierbarkeit »computerimplementierter Erfindungen«. Es wird dabei allerdings oft übersehen, dass wir bereits heute mit ungefähr 10.000 deutschen und 30.000 europäischen Patenten im Bereich Informationstechnologie leben, die nicht etwa illegal, sondern im Einklang mit den gesetzlichen Vorgaben erteilt worden sind. Diese vorhandenen Patente werden weder der deutsche Gesetzgeber noch die europäische Legislative beseitigen können, ohne sich Entschädigungsansprüchen auszusetzen. Man wird also mit den Schutzrechten leben müssen. Auf europäischer Ebene entscheidet sich in den nächsten Monaten »nur«, ob man den Patentämtern für die Zukunft eine engere oder weitere Erteilungspraxis vorschreiben wird. Es ist dabei allgemein bekannt, dass sich das Europäische Parlament in seiner Entschließung vom 24.09.2003 für eine engere Fassung stark gemacht hat (vergleiche http://europa.eu.int/comm/internal_market/en/indprop/comp/com02-92de.pdf).

In Anbetracht der heute bereits vorhandenen Patente ist es erstaunlich, wie gering die Zahl der patentrechtlichen Konflikte im Bereich Open Source letztlich ist.

Für einiges Aufsehen hat der Fall der Stadt München im Sommer 2004 gesorgt. Die Stadt hatte ihr Projekt »LiMux« (die Migration von ungefähr 14.000 Arbeitsplätzen von MS Windows/Office auf GNU/Linux, Open Office) zunächst wegen patentrechtlicher Bedenken auf Eis gelegt, nach Einholung eines Gutachtens aber nach wenigen Wochen wieder aufgenommen. Das Gutachten kam zu folgendem Ergebnis (<http://www.jurpe.de/aufsatz/20050010.htm>): Es bestehe keine erhöhte Gefahr von rechtlichen und wirtschaftlichen Konsequenzen beim Einsatz von Freier Software, wenn man die Situation mit derjenigen von proprietärer Software vergleiche. Das erscheint plausibel. Es kann immer vorkommen, dass man den Einsatz von gutgläubig erworbener Informationstechnologie später verboten bekommt, weil geistige Eigentumsrechte Dritter entgegenstehen. Professionelle Kunden wenden sich in diesem Fall an die Lieferanten und Dienstleister, die ihnen diese Produkte geliefert haben und verlangen Abhilfe oder Schadensersatz. Diese müssen die Software grundsätzlich frei von Rechten Dritter, die einer Verwendung entgegenstehen können, verschaffen. Dabei ist es egal, ob es sich um Open Source-Software oder um proprietäre Software handelt.

Wie bei der Kommentierung der Ziffern 11 und 12 GPL im Einzelnen ausgeführt, muss Schadensersatz wegen Patentverletzung nur gezahlt werden, wenn dem Distributor oder sonstigen Anbieter ein Verschulden zur Last fällt. Dies gilt sowohl im Verhältnis zu seinen Kunden als auch im Verhältnis zum Patentinhaber. Für Unternehmen ist hier der Grundsatz »Patente muss man kennen« zu beachten. Es ist also allenfalls für private Einzelentwickler und Gruppen möglich, sich auf die Unkenntnis der in Frage stehenden Patente zu berufen. Unabhängig von einem etwaigen Verschulden können allerdings Unterlassungsansprüche des Patentinhabers auf Entwickler und Distributoren zukommen. Für den Unterlassungsanspruch ist es unerheblich, ob der Patentverletzer Kenntnis von dem Patent hatte oder hätte haben müssen. Die weitere Nutzung einstellen zu müssen, kann schon ärgerlich genug sein. Unter Umständen muss man mit dem eigenen Pro-

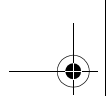
jekt von vorne anfangen. Hinzukommen können die Anwaltskosten des Patentinhabers wegen Abmahnungen oder gerichtlicher Vertretung.

Von der Frage, welche Bedeutung Patentansprüche Dritter für die Entwicklung und den Vertrieb von Open Source-Software haben können, ist die Frage zu unterscheiden, wie mit Patentanmeldungen in den eigenen Reihen der Open Source-Entwickler umgegangen werden soll. Erste praktische Beispiele hierzu hat es bereits gegeben, z.B. die Patenterteilung für ein Verfahren, das in das Echtzeit-Betriebssystem RTLinux implementiert wurde. In der Open RTLinux Patent License Version 1, durch die der Patentinhaber Victor Yodaiken die Benutzung des Patents zunächst gestattete, wurden zugleich aber auch eine Reihe von restriktiven Bedingungen diktiert, die man als Benutzer des Patents zusätzlich zu den Bestimmungen der GPL parallel erfüllen sollte. Diese zusätzlichen Pflichten verstießen gegen das Gebot aus Ziffer 2 b) GPL, wonach Bearbeitungen nur nach den Bedingungen der GPL verbreitet werden dürfen. Yodaiken ließ sich schließlich in außergerichtlichen Verhandlungen von dieser Auslegung der GPL überzeugen. Die neu gefasste Open RTLinux Patent License Version 2 gestattet nunmehr eine freie Benutzung des Patents, sofern die Erfindung mit GPL-Software ausgeführt wird (http://www.fsmlabs.com/products/rtrlinuxpro/rtrlinux_patent.html).

Eine weitere praktisch relevante Frage betrifft den Schutz von Open Source-Entwicklern für ihre Erfindungen gegen unberechtigte Patentanmeldungen Dritter. Patente werden grundsätzlich nur auf neue Erfindungen erteilt. Was zum Stand der Technik gehört, kann nicht mehr als Erfindung patentiert werden. Es ist deswegen zu raten, dass Entwickler Freier Software ihre Innovationen früh veröffentlichen und diese Veröffentlichungen in einer aussagekräftigen Form dokumentieren. Dies kann in Form eines Fachartikels sein, denkbar sind aber auch Datenbanken, sofern man nachweisen kann, dass der Zeitstempel nicht manipuliert werden kann. Entscheidend ist dabei, dass die Erfindung gegenüber der Öffentlichkeit kundgegeben wird, es muss also eine Kenntnisnahme durch einen unbestimmten, für den Erfinder nicht mehr kontrollierbaren Personenkreis möglich sein.

Hat sich die GPL in der Praxis und vor Gericht bewährt?

Die GPL existiert nun schon seit gut 15 Jahren. Da erstaunt es, dass – soweit bekannt – erst *eine* gerichtliche Entscheidung über ihre Auslegung und ihre Wirksamkeit existiert. Das Landgericht München I hat am 19. Mai 2004 entschieden, dass derjenige, der GPL-Software vertreibt, ohne dabei die Pflichten aus der GPL zu beachten, seine Rechte aus der GPL verliert (vergleiche Ziffer 4 GPL Rz. 14). Er darf dann die Software überhaupt nicht mehr weitergeben, solange er diesen Pflichten nicht nachkommt (vergleiche dazu auch die Frage »Welche Pflichten bestehen bei der Nutzung von unter der GPL lizenzierter Software?«). In diesem Zusammenhang hat das Gericht auch die Vereinbarkeit der Ziffern 2 und 3 der GPL mit deutschem Recht bestätigt. Behandelt das Urteil auch nur einige wenige Fragen, die die GPL aufwirft, so stellt es dennoch einen Meilenstein dar, weil erstmals das Lizenzmodell an sich bestätigt wurde.



Aber auch außerhalb der Gerichtssäle wurde die GPL schon vielfach durchgesetzt und hat gezeigt, dass sie eine praktisch funktionierende Lizenz ist. Die Free Software Foundation und ihr Rechtsbeistand, Professor Eben Moglen von der Columbia University, haben bereits eine Reihe von außergerichtlichen Einigungen herbeigeführt, um die Einhaltung der GPL sicherzustellen (vergleiche dazu Moglen, Enforcing the GPL, <http://emoglen.law.columbia.edu/publications/lu-12.html>). Die größte Aufmerksamkeit hat dabei der Fall »RTLinux« erlangt (vergleiche dazu die Frage »Welche Risiken bestehen gegenwärtig aufgrund von Softwarepatenten?«). Letztlich hat sich stets die Überzeugung durchgesetzt, dass die GPL in ihren entscheidenden Regelungen auch gerichtsfest ist.

Die Bedeutung einer Lizenz bemisst sich allerdings nicht nur darin, ob sie vor Gericht durchsetzbar ist. Ebenso wichtig ist die praktische Akzeptanz, die sich gerade in der immens weiten Verbreitung der GPL zeigt. Die GPL darf wohl mit Recht als die bedeutendste und meistgenutzte Open Source-Lizenz bezeichnet werden. Sie stellt den Prototyp einer Copyleft-Lizenz dar, die sicherstellt, dass Freie Software auch bei Weiterentwicklungen frei bleibt (zum Begriff des Copyleft vergleiche Ziffer 2 GPL Rz. 10).

Natürgemäß besteht bei Copyleft-Lizenzen das Problem, dass abgegrenzt werden muss, welche Weiterentwicklungen oder Veränderungen ebenfalls der GPL unterstellt werden müssen und welche Softwarebestandteile auch unter abweichenden Lizenzbedingungen vertrieben werden dürfen. Hier zeigen sich in der Praxis die weitaus meisten Probleme und Unsicherheiten (vergleiche dazu ausführlich Ziffer 2 GPL Rz. 14 ff) der im Übrigen gut funktionierenden Lizenz.

