

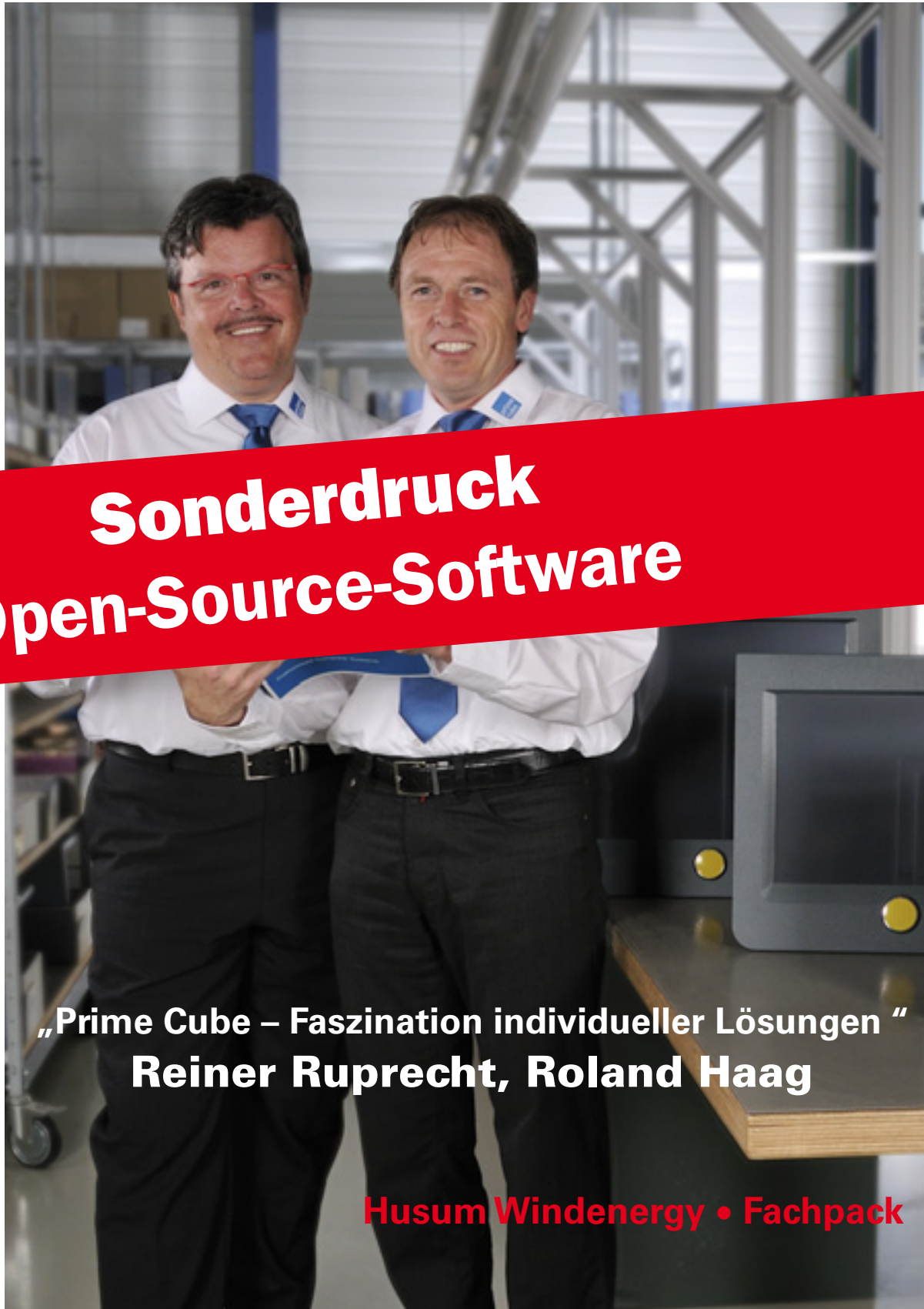
open automation

Märkte - Trends - Innovationen



National Instruments
**Im Zeichen
der Zeit**

S. 20



Sonderdruck Open-Source-Software

Automatisierung für
Windenergieanlagen

S. 38

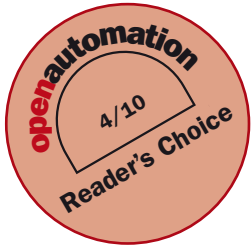


Roland Bent
**Auswirkungen
der globalen
Herausforderungen**

S. 60

„Prime Cube – Faszination individueller Lösungen“
Reiner Ruprecht, Roland Haag

Husum Windenergy • Fachpack



Communities und Open Source in der Automatisierung

Open-Source-Software gewinnt zunehmend an Bedeutung für automatisierungstechnische Systeme und Communities sorgen für deren Weiterentwicklung. Eine Expertenrunde mit Hans-Jürgen Hilscher, Geschäftsführer der Hilscher Gesellschaft für Systemautomation mbH, Herbert Köbel, zuständig für die Software-Entwicklung bei der Leuze Electronic GmbH & Co. KG, Wolfgang Hopfes vom Produktmarketing der Softing AG und Dr. Carsten Emde, Geschäftsführer der OSADL e.G., setzt sich mit aktuellen und zukünftigen Entwicklungen auseinander.

Welche Vorteile bietet die freie Software in der Automatisierungstechnik?

H.-J. Hilscher: Wir sind im Bereich der Kommunikationstechnik unterwegs, bei der die Funktionalität exakt definiert ist. Diese Technologien sind prädestiniert dafür, auf einer breiten Basis entwickelt zu werden. Und dazu können unserer Meinung nach ein offenes System und eine freie Software wertvolle Dienste leisten. Wir haben vor, diesen Weg nun einzuschlagen.

W. Hopfes: Bei uns ist die Sache etwas anders gelagert. Wir sind im Bereich Wireless der WITECK beigetreten, einer Einkaufsgemeinschaft, um schneller am Markt

agieren und günstiger entwickeln zu können. Wenn unsere Entwickler neue Funktionen schaffen, die dem Nutzer einen Mehrwert bieten, entsteht oft die interne Diskussion, ob wir diese auch der Allgemeinheit zur Verfügung stellen sollen. Geben wir damit nicht einen Vorteil an den Wettbewerb weiter?

H. Köbel: Ich habe noch keine Erfahrung mit freier Software, erwarte aber, dass die Eingangsschwelle, zum Beispiel bei Kommunikations-Software, tiefer gelegt wird. Mit freier Software sollte der Einstieg erleichtert werden, insbesondere auch für Hochschulen und den gesamten Bildungssektor.

Dr. C. Emde: Wir sollten den Begriff Open Source genauer definieren. Zunächst einmal sollten wir nur diejenigen Lizenzen als „Open Source“ bezeichnen, die von der Open Source Initiative (OSI) als solche anerkannt wurden. Diese Organisation wacht nämlich darüber, dass nicht irgendwo Open Source draufsteht, wo nicht Open Source drin ist. In dieser Gruppe von „echten“ Open-Source-Lizenzen existieren aber durchaus noch Abstufungen. Die wichtigste Abstufung ist das sogenannte Copyleft. Das Copyleft ist eine Eigenschaft, die man kennen muss, wenn man über Open Source spricht. Denn normalerweise hat jeder Au-



Dr. Carsten Emde von OSADL, Wolfgang Hopfes von Softing, Herbert Köbel von Leuze und Hans-Jürgen Hilscher von Hilscher loten die Chancen von Open-Source-Software für die Automatisierung auf dem Automatisierungstreff in Böblingen aus

tor, der auch nur eine einzige Code-Zeile schreibt, das Urheberrecht an diesem Code; d. h. in dem Augenblick, in dem er ihn weitergibt, darf er festlegen, unter welcher Lizenz er den Code weiterreicht. Diese normale Regelung gilt unabhängig davon, ob es sich um neuen Code handelt oder ob der Code eine bestehende Software verbessert bzw. verändert. Bei Open-Source-Software gilt dies nur, wenn es sich um eine Copyleft-freie Lizenz handelt. Bei einer Copyleft-Lizenz – z. B. der GNU General Public License

lung ist vor allem in den letzten zwei Jahren sehr schnell gegangen, und immer mehr Architekturen und Plattformen werden von keinem anderen Betriebssystem mehr unterstützt. Auch bei der Sicherheit gibt es viele Aspekte, die sich nur mit Open Source realisieren lassen. Je mehr Tester auf die Systeme schauen, um so sicherer werden sie. Daher gibt es immer häufiger die Auffassung, dass für Security- und neuerdings auch Safety-Anwendungen Open Source eine wichtige Voraussetzung ist.

men zusammenzuarbeiten, wobei kartellrechtliche und wirtschaftliche Aspekte verbindlich und klar geregelt sind. Open Source ist also eine Sonderform von „Open Innovation“, d.h. ein Konzept bei dem verschiedene Unternehmen – wobei es sich durchaus um Mitbewerber handeln kann – gemeinsam diejenigen Basistechnologien entwickeln, die für keinen Teilnehmer alleinstellungsrelevant sind. Open Source ist in erster Linie so erfolgreich, weil es ökonomisch erfolgreich ist. Was die Frage anbelangt, inwieweit jemand von seinem Arbeitgeber die Erlaubnis erhält, Open-Source-Software zu entwickeln, sollte man wissen, dass z. B. über 95 % der Linux-Kernel-Entwickler bei bekannten Linux-Nutzern, wie Google, IBM und Novell, angestellt sind. Diese Mitarbeiter entwickeln Open-Source-Software im Auftrag ihres Unternehmens. Daher geht es nicht um die Frage, ob der Mitarbeiter Zeit dafür hat. Communities können ihn natürlich dabei unterstützen, dies effektiver zu tun.

H.-J. Hilscher: Wir haben als Kunden einen großen Antriebshersteller, der noch mehr Produkte einsetzen würde, wenn er unsere Protokoll-Stacks unter VX Works zur Verfügung hätte. Dies können wir leider nicht leisten. Der Kunde könnte die Stacks portieren – für ihn ist jedoch der Aufwand zu groß. Dies ist für uns ein Ansatzpunkt für Open Source.

W. Hopfes: Meine Erfahrung mit Communities ist die: Sie haben von zehn Teilnehmern einen, der sich aktiv einbringt und der Rest versucht zu profitieren. Hier sehe ich ein Ungleichgewicht, welches die Aktivitäten zum Erliegen bringen kann.

Dr. C. Emde: Basierend auf Formeln, nach denen sich anhand der Lines of Code der Wert einer Software berechnen lässt, hat der Linux-Kernel inzwischen einen Wert von etwa einer Milliarde Euro. Dazu kommen jährliche Service-Leistungen in Höhe von 400 Mio. € bis 600 Mio. €. Diese Software steht genauso wie das in vielen Naturwissenschaften erworbene Wissen allen Menschen kostenlos zur Verfügung. Wenn man diese Software kommerziell nutzen will, muss man natürlich die Lizenzbedingungen beachten. Erfolgreiche Open-Source-Projekte wie Linux sind aus einem freiwilligen Zusammenschluss von Menschen entstanden. Die Community entwickelt immer nur diejenigen Dinge, die viele Teilnehmer benötigen und nutzen. Unternehmen haben dagegen primär ihre eigenen Projekte im Fokus. Dies ist eine natürliche Grenze. Es wird auch in Zukunft daher immer beide Varianten geben: Closed Source für spezielle Anwendungen und Open Source für breite Basistechnologien. Auch in Zukunft wird eine Maschine folglich vermutlich immer noch mindestens zur Hälfte Nicht-Open Source-Software enthalten.

Dr. C. Emde: „Closed Source für spezielle Anwendungen und Open Source für breite Basistechnologien“



(GPL) – hat der Autor diese Freiheit nicht grundsätzlich, sondern muss den Code unter die Ursprungslizenz stellen, wenn es sich um Arbeiten an existierendem Code handelt. Interessanterweise wird diese Verpflichtung wiederum mit dem Urheberrecht durchgesetzt.

Kann Open Source die notwendige Sicherheit für den Maschinenbau bieten?

H. Köbel: Produkte müssen zuverlässig arbeiten. Dabei ist es zunächst einmal unwichtig, ob es sich um Open Source handelt, ob wir Produkte zugekauft oder sie selbst entwickelt haben. Sollten wir Open-Source-Software einsetzen, müssen wir sie sorgfältig testen, sodass ein zuverlässiger Betrieb gewährleistet ist. Daher sehe ich zunächst keinen Unterschied, zum Beispiel zu einer gekauften Bibliothek, da wir in beiden Fällen für die Sicherheit sorgen müssen.

Dr. C. Emde: Zwei Aspekte dazu: Wir haben in vielen Bereichen gesehen, dass die Nicht-Open-Source-Systeme verschwunden sind oder dabei sind, zu verschwinden. Anwender haben also seltener die freie Wahl, ob Open Source eingesetzt wird oder nicht. Als Beispiel möchte ich die Echtzeit- Betriebssysteme nennen. Es gibt für viele Bereiche keine Alternative mehr zu Linux. Die Entwick-

Müssen wir zwischen Communities und Open Source unterscheiden? Werden Communities nicht immer dazu benötigt, Open-Source-Software zu entwickeln?

H. Köbel: Aus meiner Sicht steht bei Open Source immer ein konkretes Produkt im Fokus. Bei Communities muss ein Produkt noch nicht real existieren. Communities sind Informationsplattformen, von denen wir gerne profitieren, um Spezialwissen in die Produkte einfließen zu lassen.

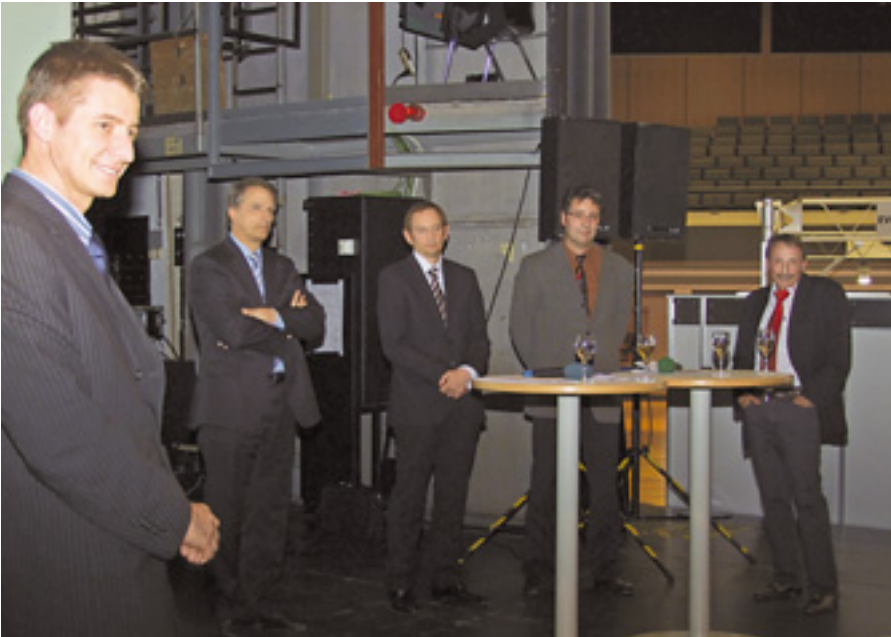
Dr. C. Emde: Open Source gibt es nicht ohne Communities. Aber Communities gibt es ohne Open Source.

H.-J. Hilscher: Wir haben eine Plattform bzw. Community geschaffen, die eine wichtige Voraussetzung für unsere zukünftigen Open-Source-Aktivitäten darstellt.

W. Hopfes: Communities sind ein wichtiges Instrument zur Informationsbeschaffung. Sie leben von Teilnehmern, die sich die Zeit nehmen, fundiertes Wissen der Allgemeinheit zur Verfügung zu stellen. Genau dort sehe ich das Problem.

Gibt es denn genügend Fachleute, denen die Zeit für solche Aktivitäten zur Verfügung steht?

Dr. C. Emde: Open Source ermöglicht einem Unternehmen, mit anderen Unterneh-



**Diskussionsrunde
moderiert von
openautomation**

Welche Ziele lassen sich mit den Communities verfolgen?

W. Hopfes: Mir ist der Community-Gedanke wichtiger als der Aspekt Open Source. Ich finde die Gemeinschaft wichtig, die etwas kollektiv nach vorne bringt. Wir denken auch bei uns im Unternehmen intensiv über Bereiche nach, die sich für diese Vorgehensweise eignen. Es gilt, Mittel und Wege zu finden, diese Communities anzuschließen und für eine gewisse Zeit am Leben zu halten. Wir machen auch Versuche bezüglich der benötigten Kommunikation. Viele Unternehmen haben heute schon ein eigenes „Wiki“; es wird jedoch nur immer von einem Teil der Belegschaft, meist aus der Entwicklungsabteilung, genutzt. Bei den Stacks setzten wir nach wie vor auf unsere eigenen Entwicklungen. Wir kapseln dieses Know-how und schaffen definierte Schnittstellen für unsere Anwender, um kundenspezifische Applikationen zu lösen.

H.-J. Hilscher: Wir wollen ein Marktsegment abdecken, welches wir mit eigenen Ressourcen sonst nicht bedienen könnten. Gerade bei Software-Stacks lohnt sich der finanzielle Aufwand für Neuentwicklungen im eignen Unternehmen zum Teil nicht. Mit der Community bieten wir Firmen, die an der Weiterentwicklung für eigene Anwendungen Interesse haben, oder Partnern, die ihre Dienstleistung anbieten, eine gemeinsame Plattform.

Herr Köbel, sie sind ein Anwender von „netX“. Ärgert es sie nicht, dass sie in Zukunft Software-Produkte kostenlos erhalten, für die sie vor einiger Zeit noch zahlen mussten?

H. Köbel: Auf den ersten Blick ja. Auf den zweiten Blick sehe ich in der Community

große Vorteile. Wir sind als Anwender auf einen guten Support und auf Entwicklungsunterstützung angewiesen. In Bezug darauf, stellt die Community eine weitere Quelle dar. Auf diese Weise sparen wir mehr Kosten ein, als für die Investition getätigt wurden.

Kann ein Anwender sicher sein, das Open-Source-Software fehlerfrei ist? Und wenn sie Fehler hat: Wer übernimmt dann die Gewährleistung?

Dr. C. Emde: Zu Beginn der Software-Entwicklung in den 1960-er und 1970-er Jahren des letzten Jahrhundert war es üblich, Software für einen jährlichen Betrag einzukaufen, der auch die Unterstützung im Fehlerfall beinhaltete. Software war zu dieser Zeit also eine Dienstleistung. In den 1980-er Jahren kam dann die (kommerziell sicher grandiose) Idee auf, Software in einer Schachtel zu verpacken – so wurde sie zum Produkt. Dies macht jedoch keinen Sinn, da bei Neuerungen, zum Beispiel bei der Hardware, der Wert des Produkts schlagartig auf Null sinken kann. Wir sollten Software in Zukunft also wieder als Dienstleistung sehen. Closed Source-Software wird sich nur noch bei branchenspezifischen Aufgaben durchsetzen, im Bereich der Betriebssysteme für Maschinen und Anlagen sicher nicht. Denn der Maschinenbauer ist in einer Art Sandwich-Position: Er muss seinem Kunden für bis zu 20 Jahre etwas versprechen, was sein Lieferant nicht hält. Das Management dieser Situation ist in den letzten Jahren so kostenintensiv geworden, dass das Open-Source-Modell schon aus diesem Grund so schnell angenommen wurde. Natürlich hat Software Fehler in beiden Varianten, in Closed Source- und in Open Source-Form: Bei Closed Source-Software werden diese Fehler aber irgend-

wann nicht mehr behoben, während bei Open Source die Weiterentwicklung der Software für lange Zeit gesichert ist.

Welche Rolle werden Open Source und Communities in fünf bis zehn Jahren in der Automatisierung spielen?

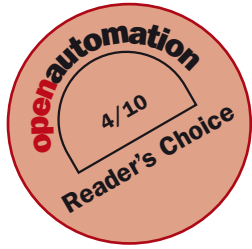
H. Köbel: Wir werden sehr genau beobachten, wie sich das Thema entwickelt. Momentan sind bei uns alle Grundsatzentscheidungen für die Entwicklung neuer Produkte getroffen. Sollte sich mithilfe von Communities jedoch eine Win-Win-Situation abzeichnen, sind wir natürlich für neue Wege offen.

H.-J. Hilscher: Wir haben den Trend erkannt und werden mit ihm wachsen. Basistechnologien, zu denen unter anderem auch die Protokoll-Stacks gehören, sind in Zukunft nicht mehr anders wirtschaftlich zu realisieren.

W. Hopfes: Wir werden in der Automatisierung eine Entwicklung durchlaufen, wie wir sie auch schon bei der Consumer-Elektronik erlebt haben. Eine Prognose zum Verhältnis Open Source zu Closed Source wage ich zurzeit jedoch noch nicht. Der Community-Ansatz wird aber sicher den größten Fortschritt bringen.

Dr. C. Emde: Der Umbruch hat schon stattgefunden. Meine Prognose: In naher Zukunft wird Linux so eine Art Weltkulturerbe und wird von allen Menschen so natürlich genutzt wie Mathematik, Physik und Chemie. Es wird eine noch klarere Trennung geben zwischen Software-Basistechnologien, die unter eine Open Source-Lizenz gestellt werden, und spezifischen Verfahrenstechniken und Branchenanwendungen, die aus gutem Grunde weiterhin als Closed-Source-Software vertrieben werden.

Ronald Heinze



Rechtliche Fragen zur Open Source Software

Die zunehmende Verbreitung Community-basierter Software-Entwicklung und speziell die Verwendung von Open-Source-Lizenzen hat auch das Interesse des Maschinenbaus und der Automatisierungsindustrie geweckt. Im Zusammenhang mit dem Einsatz von Open-Source-Software in der Industrie erläutert Rechtsanwalt Dr. Till Jaeger juristische Details von Open-Source-Lizenzen.

Herr Dr. Jaeger, Sie haben sich als Fachanwalt für Urheber- und Medienrecht unter anderem auf Open-Source-Lizenzen spezialisiert und ein internationales Renommee auf diesem Gebiet erworben. Was genau ist eine Open-Source-Lizenz?

Dr. T. Jaeger: Eine Open Source-Lizenz ist eine Software-Lizenz, die der Open-Source-Definition der Open-Source-Initiative entspricht (<http://www.opensource.org/docs/definition.php>). Diese Definition ist im Softwaremarkt allgemein anerkannt und verlangt im Wesentlichen, dass eine Software von jedermann zu jedem beliebigen Zweck lizenzgebührenfrei genutzt werden kann. Dies schließt den Vertrieb ebenso wie die Weiterentwicklung ein und setzt daher Zugang zum Sourcecode voraus.

Es gibt offensichtlich eine Reihe von Open-Source-Lizenzen, z. B. die Berkeley Software Distribution (BSD) oder die GNU General Public License (GPL). Bestehen typische Unterschiede zwischen den Open Source Lizenzen?

Dr. T. Jaeger: Open Source-Lizenzen unterscheiden sich aufgrund der Vorgaben der Open-Source-Definition nur geringfügig im Umfang der eingeräumten Rechte. Allerdings können bei den Pflichten Abweichungen bestehen. Die größte Bedeutung kommt dabei der Frage zu, wie Weiterentwicklungen lizenziert werden müssen. Bei sogenannten „Copyleft“-Lizenzen, wie der GPL, müssen Bearbeitungen ebenfalls wieder unter der Ursprungslizenz freigegeben werden, wenn die Software vertrieben wird – Open Source bleibt damit stets Open Source. „Non-Copyleft“-Lizenzen, wie die BSD License, machen insoweit keine Vorgaben, so dass Weiterentwicklungen auch proprietär lizenziert werden können – Open Source muss nicht Open Source bleiben.

Mitunter wird auch der Begriff „Free Software“ verwendet. Ist dies etwas anderes?

Dr. T. Jaeger: Lizenzrechtlich nicht, weil die Open-Source-Definition und die Free-Software-Definition der Free Software Foundation (FSF) grundsätzlich übereinstimmen. Während die FSF die Betonung mehr auf den Aspekt der Freiheit der Nutzer legt, wurde der Begriff „Open Source Software“ 1998 für ein besseres Marketing in der Softwarewirtschaft erfunden.



Dr. Till Jaeger ist Partner der Kanzlei JBB Rechtsanwälte in Berlin und hat sich auf Fragen des gewerblichen Rechtsschutzes und Urheberrechts spezialisiert. Er berät im Schwerpunkt Unternehmen aus der IT-Branche und ist Syndikus des Open Source Automation Development Lab (OSADL)

Wenn ein Unternehmen der Automatisierungsbranche beschließt, Open-Source-Software in seinen Produkten einzusetzen, d. h. mit dem Vertrieb seiner Produkte Open-Source-Software weiterzugeben, muss er dabei etwas beachten? Wenn ja, was?

Dr. T. Jaeger: Es sind vor allem die Vertriebspflichten der Open-Source-Lizenzen zu

beachten. Typisch sind etwa die Mitlieferung der Lizenztexte und ein Sourcecode-Angebot. Bei Copyleft-Software muss unbedingt auch überprüft werden, ob Eigenentwicklungen als Bearbeitungen freigegeben werden müssen oder als unabhängige Programme auch proprietär lizenziert werden dürfen. Bei der Kombination mehrerer Open-Source-Komponenten ist auf Lizenzkompatibilität zu achten. Für die Automatisierungsbranche sind zudem Patente von erhöhter Relevanz.

Wenn ein Automatisierer Open-Source-Software innerhalb seines Unternehmens einsetzt, also z. B. in der Fertigung, aber die Software nicht weitergibt, muss er dann auch Lizenzpflichten beachten?

Dr. T. Jaeger: Für die rein interne Benutzung sehen die allermeisten Open-Source-Lizenzen keine Lizenzpflichten vor. Die GPL betont dies sogar ausdrücklich.

Beschreiben die Begriffe „Open Source“ und speziell „Open Source Software“ eigentlich einen bestimmten Sachverhalt? Werden diese Begriffe von irgendeiner Organisation festgelegt? Oder darf jedermann diese Begriffe so verwenden, wie er will?

Dr. T. Jaeger: Wie bereits erwähnt, der Begriff wird im Markt einheitlich entsprechend der Open-Source-Definition verstanden. Ein abweichender Gebrauch kann irreführend sein und damit auch wettbewerbsrechtlich problematisch.

Wie verhält es sich mit dem Begriff „Community“ – wird damit etwas lizenztechnisch Relevantes bezeichnet oder ist eine „Community“ einfach eine Gruppe von Programmierern, die gemeinsam Software entwickeln, dabei aber frei sind bei der Wahl der Software-Lizenz?

Dr. T. Jaeger: Als Community wird die klassische dezentrale Organisation von Softwareentwicklern bezeichnet. Es gibt aber auch zahlreiche andere „Communities“ und



keine enge Definition wie bei Open Source Software.

Nehmen wir mal an, jemand schreibt eine Software, in der ein Verfahren implementiert wird, das durch ein Patent eines anderen Unternehmens geschützt ist. Kann eine solche Software überhaupt unter einer Open-Source-Lizenz weitergegeben werden?

Dr. T. Jaeger: Das geht nur in dem Ausnahmefall, dass der Patentinhaber für den Gebrauch der Software eine lizenzgebührenfreie Nutzungserlaubnis erteilt, die auch Weiterentwicklungen gestattet. Ansonsten würde das Element der lizenzgebührenfreien Nutzbarkeit fehlen – die Software wäre nicht mehr „frei“ bzw. Open Source. Daher ist nach der aktuellen BGH-Entscheidung zum Beispiel derzeit fraglich, inwieweit VFAT-Implementierungen als Open Source Software möglich sind.

Wenn nun eine Software mit einem patentgeschützten Verfahren ohne Freistel-

lung des Rechteinhabers unter einer Open-Source-Lizenz weitergegeben wird, und ein Maschinenbauer setzt diese Software in einem Produkt ein und verkauft dies im Glauben, es handele sich tatsächlich um Open Source Software, kann dies negative Folgen für den Maschinenbauer haben?

Dr. T. Jaeger: Das hängt vom Einzelfall ab. Denkbar wäre, dass die Lizenznehmer, die auf die Nutzung als Open Source Software vertraut haben, einen Schaden erleiden – etwa durch die Geltendmachung patentrechtlicher Ansprüche des Rechteinhabers – und Regress geltend machen. Jedenfalls funktioniert das Open-Source-Lizenzmodell auf diese Weise nicht.

Was ist eigentlich ein Standard, was ist ein „Open Standard“? Hat dies etwas mit „Open Source“ zu tun?

Dr. T. Jaeger: Der Begriff „Open Standard“ ist anders als der Begriff „Open

Source“ stark umstritten. Einige verstehen darunter nur einen freien Zugang zur Spezifikation des Standards, andere verlangen, dass der Standard auch durch Open Source implementierbar sein muss. Dies würde Lizenzgebühren für die Nutzung von Rechten an dem Standard ausschließen. Die Diskussion ist hier nicht abgeschlossen. Vielleicht sollte man besser zwischen „offenen Standards“ und „freien Standards“ unterscheiden und dabei klare Definitionen zugrunde legen.

Darf ein Unternehmen ein Verfahren als „Standard“ anerkennen lassen, die Verwendung dieses Standards aber an eine Patentnutzungslizenz knüpfen?

Dr. T. Jaeger: Es ist durchaus üblich, dass für die Nutzung von Standards Patentlizenzgebühren verlangt werden. Nur die komplette Verweigerung der Lizenzierung ist bei Standards kartellrechtlich unzulässig.

Ronald Heinze

cifX PC-Cards

CONNECTED BY
netX

cifX – reduced to the maximum

cifX – basierend auf der netX Technologie - heißt eine Karte für alle Markt relevanten Real-Time-Ethernet Systeme. Arbeitet als Master oder Slave in verschiedene Formfaktoren mit PCI und PCI Express Interface.

- Im Linuxkernel voll unterstützt (Mainline und PREEMPT_RT)
- GPL-konformer Treiber, da Protokoll-Stacks in Firmware
- Performance-Daten aus OSADL-Testlabor verfügbar
- Zehn Jahre Liefergarantie

hilscher
COMPETENCE IN COMMUNICATION
www.hilscher.com

- Basistechnologien gemeinsam entwickeln
- Alleinstellungsmerkmale erhalten
- Software, die nicht abgekündigt werden kann
- Auf die Bedürfnisse der Automatisierungsindustrie zugeschnitten
- Leistungsfähig durch eine große Entwicklergemeinschaft
- Sollte Ihr Unternehmen nicht auch Mitglied im OSADL werden?

Open Source Automation Development Lab eG
www.osadl.org
info@osadl.org