

Dieser Artikel ist Teil des
Open Source Jahrbuchs 2007

Bernd Lutterbeck
Matthias Bärwolff
Robert A. Gehring (Hrsg.)

Open Source
Jahrbuch 2007

Zwischen freier Software und Gesellschaftsmodell

erhältlich unter www.opensourcejahrbuch.de.

Die komplette Ausgabe enthält viele weitere interessante Artikel. Sie können diesen und andere Artikel im Open-Source-Jahrbuch-Portal kommentieren oder bewerten: www.opensourcejahrbuch.de/portal/. Lob und Kritik sowie weitere Anregungen können Sie uns auch per E-Mail mitteilen.

Einführung von Linux im öffentlichen Sektor*

HAL VARIAN UND CARL SHAPIRO



(CC-Lizenz siehe Seite 563)

Dieser Artikel untersucht die Möglichkeiten eines Einsatzes von Linux, hauptsächlich für den öffentlichen Sektor. Dabei wird deutlich, dass das freie Betriebssystem mit gängiger proprietärer Software konkurrieren kann. Es verfügt über eine große Anzahl interessanter Eigenschaften, wie z. B. die Möglichkeit für die Anwender, Lock-In-Effekten zu entgehen oder die grundsätzliche Nutzung von offenen Schnittstellen. Darüber hinaus gibt es besonders für den öffentlichen Sektor Gründe für eine breite Nutzung von Open-Source-Software. Dies könnte positive Auswirkungen auf die gesamte Softwarebranche eines Landes haben und so insgesamt zu wirtschaftlichen Vorteilen führen.

Schlüsselwörter: Lock-In-Effekte · öffentliche Verwaltung · Softwarequalität · offene Schnittstellen

1 Einführung

Das Linux-Betriebssystem stellt in der Informationstechnologie, sowohl im privaten als auch im öffentlichen Sektor, zunehmend eine interessante Alternative dar, vor allem für den Betrieb von leistungsstarken Servern.¹ Die Entscheidung zur Umstellung auf eine andere Software-Plattform hat generell nachhaltige Auswirkungen auf die nachfolgende Einführung von Anwendungssoftware oder plattformbezogener Soft-

* Aus dem Englischen von Annett Wendschuh.

1 Server stellen Endnutzern Informationen oder Dienste bereit und arbeiten üblicherweise ohne ständige Überwachung. Server-Hardware können Großrechner (*Mainframes*), Arbeitsplatzsysteme (*Workstations*) oder Arbeitsplatzrechner (PCs) sein, wobei die Wahl von der Größenordnung und den Betriebsanforderungen des Einsatzgebietes abhängig ist. Im Gegensatz dazu versteht man unter Desktop-Computern die normalen Arbeitsplatzrechner (PCs), die für unterschiedliche interaktive Aufgaben wie Textverarbeitung, Kalkulation, E-Mail-Bearbeitung oder Internetdienste eingesetzt werden.

ware.² Noch umfassender und komplexer werden die Auswirkungen der Einführung von Linux durch die Tatsache, dass es sich um Open-Source-Software (OSS) statt um proprietäre Software handelt.

Darüber hinaus kann die umfassende Einführung von OSS-Plattformen im öffentlichen Sektor starken Einfluss auf die wirtschaftliche Entwicklung der gesamten Softwareindustrie eines Landes haben, ein Gesichtspunkt, der vor allem für öffentliche Entscheidungsträger von zentraler Bedeutung sein dürfte.

Obwohl sich diese Erörterung vorwiegend mit Linux beschäftigt, wird sie notwendigerweise auch auf die Wirtschaft der Softwaremärkte und auf die Unterschiede eingehen, die zwischen dem traditionellen Entwicklungsmodell für proprietäre Software und dem Modell für OSS bestehen, das von Linux, Apache und anderen bekannten Open-Source-Anwendungen genutzt wird. Hintergrundinformationen zu den wirtschaftlichen Gegebenheiten auf Softwaremärkten enthält der diesbezügliche Artikel im vorliegenden Band.³ Für eine tiefgründigere Einarbeitung in diese Konzepte empfiehlt sich die Lektüre unseres Buches (Shapiro und Varian 1999).

Der vorliegende Artikel ist folgendermaßen aufgebaut: Zunächst wird ein kurzer Überblick über die grundsätzlichen Untersuchungsergebnisse und Empfehlungen gegeben. Der zweite Abschnitt beschäftigt sich mit der Definition und Erklärung der Grundkonzepte von OSS und proprietärer Software sowie offenen und proprietären Schnittstellen. Dabei wird besonderer Wert auf die Feststellung gelegt, dass OSS mit proprietärer Software interagiert und diese ergänzt. Außerdem sollen einige der grundlegenden wirtschaftlichen Bedingungen genauer bestimmt werden, die sich günstig auf das Open-Source-Modell für die Softwareentwicklung auswirken. Im dritten Teil werden die vorher beschriebenen Grundprinzipien angewendet, um Nutzen und Kosten der Einführung von Linux aus Sicht des Anwenders zu bewerten. Die Ergebnisse dieser Analyse gelten für private wie öffentliche Anwender gleichermaßen. Zum Abschluss beschäftigen wir uns kurz mit einigen zusätzlichen Überlegungen zur Rolle der öffentlichen Entscheidungsträger, insbesondere im Hinblick auf die Auswirkungen, die die Wahl einer bestimmten Software-Plattform auf die wirtschaftliche Entwicklung der Softwareindustrie eines Landes hat.

2 Untersuchungsergebnisse und Empfehlungen

Die Untersuchung kommt zu den folgenden grundsätzlichen Ergebnissen und Empfehlungen:

Kritische Masse Das Linux-Betriebssystem hat inzwischen eine kritische Masse erreicht.

Dadurch kann den Anwendern die Sicherheit geboten werden, dass das Sys-

2 *Software-Plattformen*, zu denen Betriebssysteme zählen, dienen der Bereitstellung unterschiedlicher Dienste für Anwendungssoftware, welche ihrerseits geschrieben wird, um auf einer Software-Plattform ausgeführt zu werden.

3 Siehe S. 125 für den Beitrag „Die Ökonomie der Softwaremärkte“.

tem auch zukünftig verfügbar bleiben und ständig verbessert werden wird, wodurch sich das finanzielle Risiko von Anwendern und Softwareentwicklern bei Investitionen in Linux verringert.

Interessante Eigenschaften Das Linux-Betriebssystem verfügt über eine Reihe äußerst interessanter Eigenschaften für IT-Manager im privaten und öffentlichen Bereich: So sind für Anwender, die auf Linux umstellen, Lock-in-Effekte weniger wahrscheinlich als bei der Nutzung von proprietären Software-Plattformen. Darüber hinaus haben sie größere Einflussmöglichkeiten auf ihre Computerumgebung. Diese Vorteile kommen ganz besonders in komplexen Computerumgebungen zur Geltung, wobei Anwender von der Fähigkeit profitieren, ihre Softwareumgebung an individuelle Bedürfnisse anpassen zu können, was gerade im öffentlichen Sektor häufig erforderlich ist.

Offene/freie Schnittstellen Charakteristisch für Open-Source-Software wie Linux ist die Nutzung offener Schnittstellen. Proprietäre Software verwendet sowohl offene als auch proprietäre Schnittstellen. Da offene Schnittstellen in der Softwareindustrie zu mehr Stabilität und Innovation führen, sollte ihnen und der entsprechenden Software von Mitarbeitern im öffentlichen Dienst der Vorzug gegeben werden, sofern die gebotene Qualität mit der von proprietären Alternativen vergleichbar ist.

*Vorteile für lokale SME-Server*⁴ Da Linux eine Open-Source-Software-Plattform ist, kann dessen Einführung dabei helfen, die Entwicklung des landeseigenen Softwaremarktes anzuregen, teilweise indem die Ausbildung von Programmierern gefördert wird. Diese befähigt sie zur Entwicklung von Anwendungen, die auf der Linux-Plattform ausführbar sind. Die Einführung von Linux kann sich darüber hinaus günstig auf die Entwicklung von proprietärer Software und Dienstleistungen auswirken, die in der Linux-Umgebung genutzt werden können.

Das Beste aus beiden Bereichen Dass sich die mit Linux verknüpften Lizenzbedingungen negativ auf die Entwicklung kommerzieller Software auswirken könnten, ist eine unbegründete Befürchtung. Auch wenn Linux Open-Source-Software ist, bedeutet das nicht zwangsläufig, dass die Entwicklung von Anwendersoftware, die unter Linux ausgeführt werden kann, ebenfalls dem Open-Source-Modell folgen muss. Eher ist die Entstehung von gemischten Umgebungen zu erwarten, die sowohl Open-Source-Software als auch proprietäre Software nutzen und über offene und proprietäre Schnittstellen verfügen, die in ihrer Gesamtheit den wirtschaftlichen Erfolg über Jahre hinaus garantieren können.

⁴ Small and medium size enterprises (SME).

3 Besprechung von proprietärer und Open-Source-Software

GNU/Linux ist derzeit das führende Beispiel für OSS.⁵ Wer die Einführung von Linux in Betracht zieht, muss ein Verständnis dafür entwickeln, wie das Open-Source-Modell für Softwareentwicklung funktioniert und inwiefern OSS eine Ergänzung zu proprietärer Software darstellen kann. Zur Verbesserung dieses Verständnisses werden im vorliegenden Artikel OSS, proprietäre Software und Schnittstellen zunächst allgemein besprochen. Diese Informationen dienen später als Grundlage, wenn wir Kosten und Nutzen bei der Einführung von Linux bewerten.

3.1 Open-Source-Software und proprietäre Software

Unter OSS versteht man Software mit öffentlich zugänglichem Quelltext⁶, der von jeder Person kopiert, verändert und weitergegeben werden kann. Der Zugang zum Quelltext bietet Programmierern und Anwendern die Möglichkeit, das zu Grunde liegende Programm einzusehen und zu verstehen. Sie können den Quelltext sogar erweitern oder verändern, wobei bestimmte Lizenzbeschränkungen bestehen können, auf die später noch eingegangen werden soll.

Im Gegensatz dazu versteht man unter proprietärer Software solche, die im Zusammenhang mit proprietären Lizenzvereinbarungen vertrieben wird, normalerweise unter Zahlung einer Lizenzgebühr. Obwohl es viele verschiedene Formen der proprietären Softwarelizenzierung gibt, erhalten Anwender von proprietärer Software sehr häufig keinen Zugang zu dem urheberrechtlich geschützten Quelltext und dürfen für gewöhnlich weder die Software selbst noch Erweiterungen zur Software weitergeben. Unternehmen, die proprietäre Software entwickeln, sichern sich in der Regel die geistigen Eigentumsrechte, um die alleinige Kontrolle über den von ihnen entwickelten Quelltext zu behalten.

3.2 Offene und proprietäre Schnittstellen

Es gibt praktisch keine Software, die vollkommen unabhängig arbeitet. Deshalb stellen Schnittstellen, d. h. die Art und Weise, wie ein Programm mit anderen Programmen, dem Anwender und mit der Hardware interagiert, für den Anwender und die Funktionalität entscheidende Komponenten dar. Schnittstellen sind besonders wichtig in komplexen Computerumgebungen, die in Regierungsbehörden und großen kommer-

5 Die allgemein als Linux bezeichnete Betriebssystem-Software sollte richtiger *GNU/Linux* genannt werden, da es sich dabei um eine Kombination von Software (GNU-Projekt) der *Free Software Foundation* und einem von Linus Torvalds entwickelten *Kernel* handelt. GNU ist ein Akronym für *GNU's not Unix*.

6 Programmierer schreiben Software in verschiedenen Computersprachen wie beispielsweise *Fortran*, *C/C++* und *Java*. Das ursprüngliche Format, in dem die Software entwickelt wurde, wird *Quelltext* genannt. Dieser Quelltext wird in ein ausführbares, binäres oder Zielformat übersetzt, mit dem der Computer arbeiten kann.

ziellen Einrichtungen charakteristisch sind. Es lassen sich drei verschiedene Arten von Schnittstellen unterscheiden:

Programmierschnittstellen, die allgemein als *APIs* bezeichnet werden und durch die beschrieben wird, inwiefern Anwendungen bestimmte Dienste vom Betriebssystem erfordern.

Benutzerschnittstellen, durch die beschrieben wird, wie der Benutzer Software visuell wahrnimmt und mit ihr interagiert.

Dokumentenformate, durch die beschrieben wird, wie Daten von den Anwendungen interpretiert und abgespeichert werden.

Es kann weiterhin zwischen offenen und proprietären Schnittstellen unterschieden werden. Eine Schnittstelle, die von einer einzelnen Gruppe kontrolliert wird und nicht jedem zur freien Nutzung zur Verfügung steht, wird proprietäre Schnittstelle genannt. Viele kommerzielle Softwareunternehmen halten an proprietären Schnittstellen fest, wofür Dateiformate ein gängiges Beispiel sind. Auf der anderen Seite gibt es offene Schnittstellen⁷, die in öffentlich zugänglichen Dokumenten vollständig dokumentiert werden und jedem zur freien Verfügung stehen. Das *Open Document Format (ODF)* beispielsweise ist ein Standardformat für editierbare Office-Dokumente, das vom OASIS-Konsortium vollständig beschrieben wurde und öffentlich zugänglich ist, anstatt sich unter alleiniger Kontrolle eines einzigen Unternehmens zu befinden – daher handelt es sich um eine offene Schnittstelle.⁸ In Zusammenhang mit OSS kommen in aller Regel offene Schnittstellen vor, die im Quelltext vollständig dokumentiert werden.

Offene Schnittstellen bieten Anwendern und Softwareentwicklern zahlreiche Vorteile. Sie erweitern das Angebot für die Verbraucher und fördern dadurch den Wettbewerb. Darüber hinaus erhöhen sie die Kompatibilität einer Vielzahl von Programmen untereinander, auch wenn diese im Laufe der Zeit im Zuge von Verbesserungen verändert werden. Wie ausführlich in unserem Buch (Shapiro und Varian 1999) diskutiert wird, können daher sowohl Softwarehändler als auch Anwender von der Einrichtung offener Schnittstellen profitieren, was zusätzlich zur Vermeidung von Konflikten um Standards zwischen inkompatiblen proprietären Schnittstellen beiträgt.

7 Eine offene Schnittstelle ist dokumentiert und steht jedem, ohne Einschränkungen durch den Autor, für die freie Nutzung zur Verfügung. Es gibt einige Grauzonen, z. B. veröffentlichte Schnittstellen, die durch einzelne Gruppen kontrolliert werden; veröffentlichte Schnittstellen mit Nutzungseinschränkungen; Schnittstellen, die innerhalb einer bestimmten Gruppe geteilt werden, aber nicht öffentlich zugänglich sind; multiple Schnittstellen usw. Die unten beschriebenen allgemeinen Fälle reichen jedoch für unsere Zwecke aus.

8 Siehe <http://www.oasis-open.org>.

3.3 Open-Source-Software und proprietäre Software als sich ergänzende Komponenten

Die Verdeutlichung der grundsätzlichen Unterschiede zwischen OSS und proprietärer Software als Modelle zur Softwareentwicklung soll nicht zu dem Schluss führen, dass sich Anwender zwischen der einen oder anderen Softwareart entscheiden müssen, um all ihre Bedürfnisse im Hinblick auf Computerarbeit zu erfüllen. Wir sind sogar im Gegenteil der Überzeugung, dass OSS und proprietäre Software zukünftig nebeneinander existieren und einander ergänzen können und werden. Schon jetzt ist eine Reihe von proprietären Anwendungen mit Linux ausführbar, darunter beispielsweise die *WebSphere* von *IBM* und Datenbankprogramme von *Oracle* und *IBM*. In gleicher Weise arbeitet OSS, einschließlich eines großen Teils der GNU-Software, mit den Betriebssystemen *Solaris* von *Sun* und *Windows* von *Microsoft*.

Aus genau diesen Gründen werden öffentliche Institutionen durch die Einführung von Linux in keiner Weise daran gehindert, für viele Anwendungen proprietäre Software einzusetzen, die unter Linux ausführbar ist. Tatsächlich stellen die Zugänglichkeit des Linux-Quelltexts und die offenen Schnittstellen zwischen Linux und Anwendungssoftware sogar große Vorteile für die Entwicklung kommerzieller Softwareanwendungen dar, die unter Linux ausgeführt werden können.

3.4 Für Open-Source-Software förderliche Wirtschaftsbedingungen

Weltweit werden die Anwender von dem zukünftigen Wettbewerb zwischen OSS und proprietärer Software profitieren können. Man kann sicherlich davon ausgehen, dass jeder der unterschiedlichen Softwaretypen in bestimmten Bereichen besonders erfolgreich sein wird, was zur Förderung des Wettbewerbs im Softwaresektor beiträgt und deshalb unserer Meinung nach positiv zu bewerten ist. Insbesondere Linux kann von einer Reihe vorteilhafter Marktbedingungen profitieren:

Nachweisbare Erfolge Es existieren zahlreiche Nachweise für den Erfolg von Linux als kostengünstige Variante beim Einsatz in großen, verlässlichen Computersystemen.

Flexibilität und Einflussnahme Viele Anwender legen großen Wert auf die hohe Flexibilität und den Erhalt weitreichender Einflussnahmemöglichkeiten auf ihre Server-Betriebssysteme, die ihnen Linux im Vergleich zu proprietärer Software bietet.

Weitverbreitete Anwendung Linux wird bereits von einer großen Zahl von Anwendern genutzt, wodurch sich einerseits das Angebot von Anwendungen erweitert, die unter Linux ausführbar sind, und andererseits immer mehr Programmierer zur Verfügung stehen, die sich mit Linux auskennen.

Starke Gemeinschaft Linux kann auf die Fähigkeiten einer vielfältigen und starken Entwicklergemeinschaft zurückgreifen, deren hohe Motivation aus der Tatsache resultiert, dass die Beteiligung am Linux-Projekt Status und Anerkennung verspricht.

Governance-Strukturen Linux verfügt über die notwendigen Führungsstrukturen und Institutionen, um einer Zersplitterung der Linux-Gemeinschaft vorzubeugen und einen Leitplan für die weitere Entwicklung zu erstellen.

Unterstützung durch Wirtschaftsunternehmen Linux wird von führenden Technologieunternehmen unterstützt, für die das Angebot eines auf die Anwenderbedürfnisse zugeschnittenen integrierten Paketes und/oder von Ergänzungen zum Linux-Betriebssystem wirtschaftlich interessant ist. Zu den wichtigsten Beispielen zählen Unternehmensplattformen und -dienstleistungen von *Red Hat*, Prozessoren von *Intel* oder Server und damit verbundene Dienstleistungen von *IBM*.

4 Einführung von Linux: Nutzen und Kosten für die Anwender

Auf Grundlage der bisherigen Überlegungen können nun Kosten und Nutzen systematisch erörtert werden, die für Anwender mit der Einführung des Linux-Betriebssystems als Plattform, auf der sie ihre Computerumgebung einrichten, verbunden sind. Wie schon angedeutet, wird Linux bereits von vielen großen Organisationen im privaten und öffentlichen Bereich genutzt. Im Folgenden geht es um Informationen, die bei der Entscheidung helfen können, ob der Einsatz von Linux im Einzelfall sinnvoll ist oder nicht.

4.1 Total Cost of Ownership

Fachleute und Berater stimmen darin überein, dass die *Total Cost of Ownership* (TCO) einen wichtigen Gesichtspunkt bei der weitreichenden Entscheidung über die Einführung neuer Software darstellen sollten. Obwohl die TCO in der Theorie genau definiert sind, kann ihre Bestimmung im Einzelfall eine äußerst komplexe Angelegenheit sein. Zwar können die Anschaffungskosten selbst problemlos angegeben werden, sie machen aber nur einen Teil der TCO aus. Hinzu kommen Ausgaben für Anwenderschulungen, Wartungsarbeiten, Upgrades und fachliche Unterstützung, die im Vergleich zum Kaufpreis sogar einen größeren Anteil der TCO darstellen können. Da viele dieser Kosten erst nach dem eigentlichen Softwareerwerb anfallen, sind Vergleiche zudem nur dann aussagekräftig, wenn durch sie die Ausgaben für den gesamten Zeitraum des Softwareeinsatzes berücksichtigt werden. Häufig muss sogar über die zeitliche Dauer des Projekts hinausgeblickt werden, für das die Entscheidung

zur Einführung neuer Software getroffen wurde. Denn in vielen Fällen können Daten, Schulungen und Arbeitsabläufe das Arbeitsprojekt überdauern, mit dem sie in Zusammenhang stehen.

Es hat unterschiedliche Versuche gegeben, die TCO von Windows und Linux in verschiedenen Computerumgebungen miteinander zu vergleichen.⁹ Der Unterschied bewegt sich meistens in Größenordnungen zwischen 10 und 15 Prozent und ist damit eher unbedeutend, denn solche Differenzen bei den TCO können leicht durch spezifische regionale Bedingungen, zufällige Ereignisse o. Ä. verwischt werden. Daher scheint auf den ersten Blick die Annahme begründet zu sein, dass bei einer konventionellen TCO-Analyse keine der beiden Plattformen über einen nennenswerten Vorteil gegenüber der anderen verfügt. Einen sehr großen Anteil der TCO machen die Lohnkosten für Systemadministratoren und Servicepersonal aus. Dabei wurden in den oben erwähnten Studien grundsätzlich die marktüblichen Löhne in den USA zu Grunde gelegt. Diese Kosten können jedoch in Ländern mit niedrigeren Lohnkosten wesentlich geringer sein. Da schon der Anschaffungspreis von Linux deutlich niedriger ist als für proprietäre Alternativen, würden die TCO, welche neben dem Kaufpreis für die Software auch die für ihren Einsatz erforderlichen Lohnkosten enthalten, in solchen Ländern wesentlich geringer ausfallen.

4.2 Umstellungskosten

Es ist ratsam, vor der Entscheidung für die Einführung neuer Software auch den Umstellungskosten besondere Beachtung zu schenken. Neben den bei der Umstellung auf das neue System direkt anfallenden Kosten sind auch die Ausgaben bei einer späteren Umstellung von diesem System zu einem anderen zu bedenken. Vorsicht ist geboten bei der Einführung eines Systems, das einen zukünftigen Wechsel schwierig machen könnte, beispielsweise durch Lock-In-Effekte, die bei der Nutzung des Systems auftreten und die zukünftige Verhandlungsposition dem Anbieter gegenüber schwächen.

Während Systemanbieter in gewisser Weise immer bestrebt sind, den Anwendern die Umstellung auf Alternativen zu erschweren, wollen diese normalerweise ihre Flexibilität erhalten. Aus der Sicht der Anwender ist es besonders wichtig, sicherzustellen, dass Dateiformate, Daten, Systemaufrufe, Programmierschnittstellen (APIs) und andere Schnittstellen, Kommunikationsstandards und Ähnliches ausreichend dokumentiert sind, um Daten und Programme leicht von einem Anbieter zum anderen übertragen zu können. In dieser Hinsicht bietet OSS mit ihren offenen Schnittstellen dem Anwender einen klaren Vorteil gegenüber proprietärer Software mit proprietären Schnittstellen. Natürlich kann proprietäre Software diesen Vorteil neutralisieren, wenn sie selbst über echte offene Schnittstellen verfügt. In diesem Zusammenhang

⁹ Siehe die zwei Studien, die auf Anfrage der Robert Francis Group (2002) und Cybersource (2002) durchgeführt wurden. Siehe auch die Studie von Bozman et al. (2002), die von *Microsoft* in Auftrag gegeben wurde.

wirkt sich die Unterstützung von OSS im Allgemeinen und Linux im Speziellen dadurch positiv aus, dass Druck auf Anbieter proprietärer Software ausgeübt wird, ihre Schnittstellen zum Vorteil der Anwender zu öffnen.

Eine weitere wichtige Feststellung ist das Bestreben der Marktführer einer Branche, die alleinige Kontrolle über ihre Schnittstellen auszuüben und anderen Anbietern Interoperabilität zu erschweren. Umgekehrt haben aber gerade Marktakteure, die sich nicht in der dominanten Position befinden, eine sehr starke Motivation auf diesem Gebiet mit dem Marktführer und untereinander zu interoperieren.

4.3 Softwarequalität

Die Softwarequalität lässt sich mit Hilfe verschiedener Kriterien bestimmen. Wichtig sind vor allem Verlässlichkeit, Wartbarkeit, Benutzerfreundlichkeit, Sicherheit und Flexibilität. Wer die Einführung neuer Software in Erwägung zieht, muss vor der Wahl eines Systems zunächst klären, welche Rolle die jeweiligen Kriterien in der eigenen Umgebung spielen. Am einfachsten lässt sich das Verhältnis zwischen den einzelnen Qualitätskriterien und den verschiedenen Modellen zur Softwareentwicklung erklären, indem die jeweiligen Vorzüge von Windows und Linux beschrieben werden. Allerdings muss darauf hingewiesen werden, dass diese beiden Systeme kaum „repräsentativ“ sind. Es handelt sich dabei um zwei der besten derzeit erhältlichen Softwarepakete, die eine zentrale und bestimmende Rolle in Computerumgebungen spielen. Von Linux kann jedoch ebenso wenig direkt auf andere Open-Source-Projekte geschlossen werden wie von Windows auf andere Softwaresysteme mit nicht zugänglichem Quelltext. Der Vergleich anderer Vertreter für OSS und proprietäre Software kann zu ganz anderen Ergebnissen führen. Dennoch können bestimmte Vergleiche die unterschiedlichen Ansätze der Entwicklungsmodelle aufzeigen.

Verlässlichkeit

Die Verlässlichkeit bestimmter OSS-Programme wird seit langem anerkannt. Die Tatsache, dass Linux große Webseiten wie *Google* betreibt, stellt seinen Ruf als verlässliche Software unter Beweis. Obwohl sich die Verlässlichkeit von Windows in den neueren Versionen erheblich verbessert hat, ist es dennoch fraglich, ob dasselbe Niveau wie bei Linux erreicht wird. Im FLOSS-Report, der durch Befragungen in 1452 europäischen Organisationen entstand, gaben 83 Prozent der Teilnehmer an, dass „größere Stabilität“ ein wichtiger oder sehr wichtiger Grund für die Einführung von OSS war (Gosh et al. 2002). Auch wenn die neueren Betriebssysteme von *Microsoft* den Ruf haben, verlässlicher zu sein als frühere Versionen, scheint ihre Architektur im Vergleich zu Linux einen inhärenten Nachteil aufzuweisen: das Fehlen eines umfassenden modularen Aufbaus. Es existiert sogar die Auffassung, dass die Verlässlichkeit des Linux-Systems eine Nebenerscheinung seines modularen Aufbaus ist. Demnach ist es einfacher, Bestandteile eines modular aufgebauten Softwaresystems zu ersetzen, ohne

die Arbeit anderer Teile zu beeinträchtigen, als bei holistischen, integrierten Systemen. Diese für das Open-Source-Entwicklungsmodell charakteristische Modularität ist geradezu eine Voraussetzung für Systeme, die von mehreren, mehr oder weniger unabhängig voneinander arbeitenden Programmierern entwickelt werden.

Wartbarkeit

Die Wartbarkeit bezieht sich auf den Aufwand im Zusammenhang mit der Aktualisierung und Erhaltung der Betriebsbereitschaft eines Systems. In der Vergangenheit konnten Paketaktualisierungen unter Linux leichter durchgeführt werden als unter Windows, weil Linux standardisierte Paketmanagement-Verfahren verwendet, um die Komplexität beherrschen zu können. Alle Dateien eines bestimmten Programms werden grundsätzlich an wenigen, genau dokumentierten Orten abgespeichert. Die meisten Konfigurationsprogramme sind textbasiert, so dass sie von den Anwendern problemlos konsultiert werden können. Auch hier wurden Berichten zufolge in den neuesten Versionen von Windows Verbesserungen im Hinblick auf die Wartbarkeit des Systems erreicht, so dass Patches und Updates fast automatisch durchgeführt werden können. Dennoch gaben im FLOSS-Report 60 Prozent der befragten Teilnehmer an, dass Einsparungen bei den Betriebs- und Administrationskosten ein wichtiger oder sehr wichtiger Grund für den Einsatz von Linux seien (Gosh et al. 2002).

Benutzerfreundlichkeit

Die Prüfung auf Benutzerfreundlichkeit ist mit hohen Kosten verbunden. Proprietäre Plattformen für Einzelanwender halten in dieser Beziehung der Open-Source-Konkurrenz gegenüber einen langjährigen Vorsprung, insbesondere bei der Benutzeroberfläche. Vor allem *Microsoft* hat in den letzten Jahren stark in die Benutzerfreundlichkeit seiner Anwendungen investiert. Dennoch haben auch einige viel beachtete Open-Source-Projekte wie *GNOME* und *KDE*, die sich speziell mit Benutzeroberflächen beschäftigen, in den letzten Jahren bedeutende Fortschritte gemacht.

Hinzu kommt, dass mit Linux eine umfassende Anpassung der Computerumgebung an individuelle Bedürfnisse möglich ist, wodurch jeder Benutzer mit seiner eigenen Umgebung arbeiten kann. Unter anderem können die gängigen Linux-Anwendungen so konfiguriert werden, dass sie wie unter Windows aussehen und arbeiten, wodurch den Anwendern der Systemwechsel zusätzlich erleichtert wird. Da die Technologien für graphische Benutzerschnittstellen immer mehr ausreifen, werden „revolutionäre“ neue Features immer seltener und stellen eher Zwischenschritte dar, wodurch Konkurrenten mehr Zeit haben, um auf bedeutsame Vorstöße zu reagieren. Dadurch ist zu erwarten, dass sich die Benutzerfreundlichkeit von Linux-Software zunehmend verbessern wird (Nichols und Twidale 2003).

Sicherheit

Es gibt eine anhaltende Diskussion über die Sicherheit von Windows und Linux. Da Linux als Mehrbenutzer-System angelegt ist, verfügt es über viele integrierte Schutzmechanismen zur Gewährleistung der Benutzersicherheit. Da der Quelltext des Systems frei verfügbar ist und von vielen Entwicklern auch genutzt wird, ist das Aufspüren von Programmfehlern (Bugs) wesentlich einfacher. Allerdings trifft das für beide Seiten zu: Angreifen wird es ebenso leicht gemacht, Bugs zu finden, wie Verteidigern, sie zu beseitigen. Wurde ein Programmfehler einmal aufgespürt, ist die Behebung des Problems mit OSS einfacher, weil jede Person die Quelle kontrollieren und den Bug analysieren kann. Darüber hinaus können versierte Anwender ihre Systeme selbst konfigurieren, um die gängigen Schwachpunkte zu beseitigen. Das Betriebssystem *Security-Enhanced Linux* von der NSA beispielsweise verzichtet an manchen Stellen auf Funktionalität zugunsten einer erhöhten Sicherheit. Die Offenlegung des Quelltexts von sicheren Systemen bietet potenziellen Anwendern daher eine gewisse Absicherung gegen etwaige Sicherheitslücken.¹⁰

Flexibilität

OSS verfügt über einen hohen Grad an Flexibilität, da sie individualisiert und an spezifische Bedürfnisse angepasst werden kann. Diese Eigenschaften haben die Experimentierfreudigkeit und die Arbeit an Verbesserungen gefördert, was zu einer beachtlichen Menge an Innovationen von Seiten der Anwender geführt hat (von Hippel 2002). Dadurch konnte sich Linux in vielerlei Hinsicht in ganz unvorhergesehene Richtungen entwickeln.¹¹ Insbesondere den folgenden Aspekten wurde bei den zahlreichen Entwicklungs- und Individualisierungsbemühungen Aufmerksamkeit geschenkt:

Klein Eine Art der individuellen Anpassung besteht in der Eliminierung aller Linux-Bestandteile, die für eine genau eingegrenzte, spezifische Aufgabe nicht gebraucht werden. So passt *Mindi Linux*, das für die Datenwiederherstellung eingesetzt werden kann, auf eine einzelne Diskette. Andere Ein-Disketten-Ausführungen von Linux sind *Coyote Linux*, *Trinux* und das *Linux Router Project*, die ebenfalls für diverse Anwendungen, wie beispielsweise Netzwerkaufgaben,

10 Die Nutzer von Open-Source-Software sind davon überzeugt, dass sie über bessere Sicherheitseigenschaften verfügt. Im FLOSS-Report gaben 75 Prozent der Befragten „besseren Zugangsschutz“ als wichtigen oder sehr wichtigen Grund für die Einführung von Open-Source-Software an (Gosh et al. 2002). Der bekannte Sicherheitsexperte Ross Anderson greift auf ein theoretisches Modell der Softwarequalität bei seiner Argumentation zurück, dass offene und geschlossene Softwaresysteme weitestgehend denselben Grad an Verlässlichkeit aufweisen (Anderson 2002).

11 Eine eindrucksvolle Auflistung der enormen Vielfalt an Linux-Distributionen ist unter <http://lwn.net/Distributions> zu finden. Die Liste enthält Distributionen für spezifische Hardware-Anforderungen, für sehbehinderte Anwender, für ISPs, für Echtzeitanwendungen (wie Einrichtungsüberwachung), für Multimedia und eine große Zahl anderer Bedürfnisse.

optimiert wurden. Die verschiedenen Ausführungen des eingebetteten Linux-Systems¹² können für bestimmte Hardwareaufgaben eingesetzt werden, etwa als Registrierkassen, PDAs, tragbare Videorecorder (z. B. Tivo), MP3-Player und Ähnliches.

Größer und Leistungsfähiger Mit Hilfe von Linux können Computer zu Clustern verbunden werden, die leistungsfähigere Rechnersysteme bilden und vielfältige Aufgaben übernehmen können, z. B. die Auswertung größerer Datenmengen, die zentrale Speicherung von Daten auf Dateiservern, die Bereitstellung von Datenbanken und den Betrieb von Webservern, aber auch Flugsimulationen, die Wiedergabe von Computergraphiken oder Wettersimulationsmodelle. In letzter Zeit gibt es bei Linux auch Bemühungen zur Einführung von *grid computing*, was Institutionen die Nutzung der Rechenleistung umfangreicher *computer arrays* erlaubt, die über das Internet bereitgestellt wird.¹³

Hochsicher Die bereits erwähnte Realisierung eines sicheren Linux-Betriebssystems der NSA ist nur eins von zahlreichen Projekten zur „Abhärtung“ von Linux. Andere Beispiele sind u. a. *Engarde Linux* und *Bastille Linux*.

Lokal angepasst Es ist kein Zufall, dass Open Source in kleineren Sprachgemeinschaften besonders beliebt ist. Ein häufiger Wunsch bei den Nutzern sind Anwendungen, mit denen sie in ihrer Muttersprache arbeiten können. Wo solche Anwendungen im derzeitigen proprietären Softwareangebot nicht existieren, haben die Nutzer teilweise ihre eigenen Open-Source-Versionen geschrieben.¹⁴

Freiheiten bei Anpassungen und bei firmeneigenen Entwicklungen werden durch Linux in einem Ausmaß ermöglicht, das mit proprietärer Software unmöglich zu erreichen ist. Daher sind Manager, die Flexibilität und Kontrolle anstreben, gut beraten, sich für Linux zu entscheiden, um so das Fachwissen in der Firma aufzubauen, welches zur Ausnutzung der Vorteile von Open Source nötig ist.

5 Open Source in Wirtschaft und Bildung

Wer sich in öffentlichen Institutionen mit der Wahl einer Software-Plattform befasst, ist verpflichtet, das System zu wählen, das für die anstehende Aufgabe am besten geeignet ist. Die bisherigen Betrachtungen konnten zeigen, dass Linux im Vergleich zu proprietären Alternativen in vielen Fällen über gleichwertige oder bessere Eigenschaften in Bezug auf Leistung, Verlässlichkeit und Sicherheit verfügt. Für den Fall,

12 Für mehr Informationen zu eingebetteten Linux-Systemen siehe auch <http://www.linuxdevices.com>.

13 Nähere Einzelheiten beim *Linux Clustering Information Center* unter <http://lcic.org>.

14 Für Teilaufflistungen von lokal angepassten Ausführungen von Linux siehe auch <http://www.linux.org/docs/ldp/howto/HOWTO-INDEXT/other-lang.html> und http://www.linuxselfhelp.com/cats/localization_language.html.

dass zwei Systeme gleich gut für eine bestimmte Aufgabe geeignet sind, werden offene Schnittstellen zu einem entscheidenden Aspekt, da sie üblicherweise mit niedrigeren Kosten für Zusammenschaltung und Umstellung verbunden sind. Dadurch verringert sich die Wahrscheinlichkeit, dass der Kunde an einen einzelnen Anbieter gebunden wird (*lock-in*). Offene Schnittstellen regen unabhängige Programmierer zur Entwicklung von Anwendungen, *add-ons* und Ergänzungsprodukten an. Der Nutzen solcher Produkte ist nicht nur für Anwender, sondern auch auf nationaler Ebene so groß, dass praktisch jeder Softwareanbieter Anspruch auf Offenheit erhebt. Die entscheidende Frage ist jedoch, ob die Anbieter tatsächlich bestrebt sind, diesem Anspruch in der Praxis auch gerecht zu werden, und zwar nicht nur vorläufig, sondern auch zukünftig, wenn die Umstellung auf ein anderes System sehr hohe Kosten verursachen kann.

Länder, die an einer starken einheimischen Softwareindustrie interessiert sind, sollten sich zunächst mit ihrem Universitätssystem befassen und sich die Frage stellen, wie die Ausbildung ihrer zukünftigen Softwareentwickler gefördert werden kann. Da durch offene Schnittstellen die lokale Entwicklung von unabhängigen Anwendungen ermöglicht wird, spielen sie hierbei eine entscheidende Rolle. Die entsprechenden Möglichkeiten werden einheimischen Softwareunternehmen nicht nur von Open-Source-Software-Plattformen selbst geboten, sondern auch von proprietären Software-Plattformen mit offenen Schnittstellen. Im Gegensatz dazu stellen proprietäre Software-Plattformen mit proprietären Schnittstellen für unabhängige Entwickler einen strategischen Nachteil den Unternehmen gegenüber dar, die die Schnittstellen zwischen Software-Plattform und Anwendungen kontrollieren. Eine solche strategische Abhängigkeit kann bei der Entwicklung proprietärer Anwendungssoftware vor Kapital- und Personalinvestitionen abschrecken.

Open Source spielt auch deshalb eine wichtige Rolle, weil dadurch die internen Arbeitsprozesse der Software sichtbar gemacht werden, wodurch Studenten genau verfolgen können, wie qualitativ hochwertige Software entsteht. Genau wie angehende Automechaniker während ihrer Ausbildung mit echten Motoren arbeiten, müssen sich angehende Systemprogrammierer an realen Betriebssystemen ausprobieren können. Die Verfügbarkeit von Systemen, die durch freien Zugang ein effektives Studium ermöglichen, wird einen großen Beitrag bei der Ausbildung fähiger Informatiker und der Entwicklung hochwertiger Produkte leisten.

6 Schlussfolgerung

Open-Source-Software ist zukünftig nicht mehr wegzudenken. Was einmal als völlig neuer, geradezu häretischer Ansatz für Softwareentwicklung galt, hat sich inzwischen in der Praxis hinreichend bewährt. Es existieren realisierbare Geschäftsmodelle für OSS-Entwickler, und jeder Anwender wird aus dem wohl überlegten Einsatz von OSS als Ergänzung zu proprietärer Software Nutzen ziehen können. Insbesondere Linux ist zum Paradebeispiel eines erfolgreichen Open-Source-Projekts gereift, weil

durch den Aufbau ständiger Institutionen der Linux-Quelltext permanent verbessert werden kann. Viele Nutzer im privaten und öffentlichen Bereich werden sicherlich stark von der Einführung von Linux in ihrer Computerumgebung profitieren. Linux hat gezeigt, dass es auch in sehr anspruchsvollen Computerumgebungen erfolgreich eingesetzt werden kann und dabei eine Reihe entscheidender Vorteile bietet. Dazu zählen Verlässlichkeit, Flexibilität, Sicherheit und die Vermeidung der Lock-In-Effekte von proprietären Alternativen.

Für Technologiemanager im öffentlichen Sektor bestehen noch weitere Gründe für die Einführung von Linux: Das Linux-Betriebssystem fördert die Ausbildung von Programmierern und stellt eine offene Plattform dar, mit deren Hilfe proprietäre und Open-Source-Anwendungen entwickelt werden können, was auf nationaler Ebene zur Stärkung der Softwareindustrie führt. Aus all diesen Gründen sollten IT-Manager, die im öffentlichen Bereich flexible Computerumgebungen aufbauen und gleichzeitig die einheimische Softwareindustrie fördern wollen, den Einsatz von Linux und von Open-Source-Software allgemein unbedingt in Erwägung ziehen.

Literatur

- Anderson, R. (2002), Security in Open and Closed Systems: The Dance of Boltzman, Coase, and Moore, Working Paper, Cambridge University. <http://www.cl.cam.ac.uk/ftp/users/rja14/toulouse.pdf> [27. Jan 2007].
- Bozman, J., Gillen, A., Kolodgy, C., Kusnetzky, D., Perry, R. und Shiang, D. (2002), Windows 2000 Versus Linux in Enterprise Computing, IDC White Paper, IDC. <http://www.microsoft.com/windows2000/docs/TCO.pdf> [27. Jan 2007].
- Cybersource (2002), Linux vs. Windows: Total Cost of Ownership Comparison, Study, Cybersource. http://www.cyber.com.au/cyber/about/linux_vs_windows_tco_comparison.pdf [27. Jan 2007].
- Gosh, R., Krieger, B., Glott, R. und Robles, G. (2002), FLOSS – Free/Libre and Open Source Software, Survey and Study commissioned by the EU, International Institute of Infonomics, University of Maastricht und Berlecon Research GmbH. <http://www.infonomics.nl/FLOSS/report/> [27. Jan 2007].
- Nichols, D. M. und Twidale, M. B. (2003), 'The Usability of Open Source Software', *First Monday* 8(1). www.firstmonday.org/issues/issue8_1/nichols/ [27. Jan 2007].
- Robert Francis Group (2002), Total Cost of Ownership for Linux in the Enterprise, Study, Robert Francis Group. <http://www.ibm.com/linux/RFG-LinuxTCO-vFINAL-Jul2002.pdf> [27. Jan 2007].
- Shapiro, C. und Varian, H. (1999), *Information Rules: A Strategic Guide to the Network Economy*, Harvard Business School Press, Boston.
- von Hippel, E. (2002), Open Source Software Projects as User Innovation Networks, Working Paper, MIT Sloan School of Management. http://idei.fr/doc/conf/sic/papers_2002/vonhippel.pdf [27. Jan 2007].