

Dieser Artikel ist Teil des  
Open Source Jahrbuchs 2006

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

Open Source  
Jahrbuch 2006

Zwischen Softwareentwicklung und Gesellschaftsmodell

erhältlich unter [www.opensourcejahrbuch.de](http://www.opensourcejahrbuch.de).

Die komplette Ausgabe enthält viele weitere interessante Artikel. Lob und Kritik zu diesem Artikel sowie weitere Anregungen können Sie uns einfach und unkompliziert mitteilen per E-Mail oder auf [www.opensourcejahrbuch.de/feedback/](http://www.opensourcejahrbuch.de/feedback/).

# Führt Open-Source-Software zu ineffizienten Märkten?

MARKUS PASCHE UND SEBASTIAN VON ENGELHARDT



(CC-Lizenz siehe Seite 499)

Der Artikel beschäftigt sich mit dem Argument, Open-Source-Software führe zu ineffizienten Märkten, da die geringen Lizenzpreise von freier bzw. Open-Source-Software (F/OSS) (ggf. in Höhe von Null) den Preismechanismus wettbewerblicher Märkte außer Kraft setzen: Weil diese Preise nicht den volkswirtschaftlichen Ressourcenverzehr widerspiegeln, würden falsche Anreizsignale gesetzt und die Ressourcenallokation fehlgesteuert. Dies betreffe auch F/OSS-Geschäftsmodelle, bei denen die Softwareentwicklung durch überhöhte Preise für Komplementärgüter quersubventioniert werden müsse. Effizienz- und Wohlfahrtsverluste seien die Folge. Der Artikel entwickelt einige theoretische Gegenargumente. Es wird gezeigt, dass die Preisstruktur bei F/OSS keineswegs eine Verzerrung darstellt und entsprechende Geschäftsmodelle positive Wohlfahrtseffekte erzeugen können. Anschließend wird diskutiert, weshalb es einem angeblich ineffizienten Produktionsmodell gelingen kann, in wettbewerblichen Märkten zu überleben. Schließlich wird argumentiert, dass aus theoretischer Sicht weder kommerzielle noch freiwillige Leistungen für F/OSS-Projekte Effizienzverluste begründen können, sofern keine negativen externen Effekte bestehen. Abschließend wird darauf verwiesen, dass F/OSS für Wettbewerb und Funktionsfähigkeit der Märkte sogar förderlich ist.

*Schlüsselwörter:* Volkswirtschaft · Märkte · Wettbewerb · Geschäftsmodelle

## 1 Einleitung

Dieser Artikel beschäftigt sich mit einem unter anderen von Kooths et al. (2003) vorgetragenen Argument, nach welchem die geringen Lizenzpreise von freier bzw. Open-Source-Software (F/OSS) und vor allem die Möglichkeit des kostenlosen Zugangs zu den meisten F/OSS-Produkten den Preismechanismus wettbewerblicher

Märkte außer Kraft setzen. Flexible Marktpreise, welche die Knappheitsverhältnisse, also die volkswirtschaftlichen Kosten des Ressourceneinsatzes zur Softwareproduktion und die Zahlungsbereitschaft (Präferenzen) der Kunden, widerspiegeln sollen, würden dadurch massiv verzerrt. Ein Preis von (nahe) Null würde nicht den volkswirtschaftlichen Ressourcenverzehr reflektieren und daher falsche Anreizsignale setzen. Eine Steuerung der Softwareentwicklung gemäß der durch Preissignale vermittelten Kundenpräferenzen sei nicht gewährleistet. Auch kommerzielle F/OSS-Geschäftsmodelle werden nach dieser Argumentation als sehr problematisch angesehen, weil überhöhte Preise für die angebotenen Komplementärgüter Teile der Softwareentwicklung quersubventionieren müssten, und somit die relativen Preise verzerrt würden. Die Open-Source-Idee, so wohlwollend man die Motive der freiwilligen Entwickler auch bewerten mag, würde die Steuerungslogik der Märkte unterlaufen. Wettbewerbsverzerrungen, Fehlallokation von Ressourcen und Effizienzverluste seien die Folge, wovon letztlich auch die konkurrierenden proprietären Anbieter betroffen seien.

Da Software und die damit verbundenen Komplementärleistungen nicht nur ein sehr bedeutender Wirtschaftsfaktor sind, sondern Software in nahezu allen Zweigen der Volkswirtschaft eine zentrale Produktionsvoraussetzung ist, muss dieses volkswirtschaftstheoretisch tiefgreifende Argument ernst genommen werden. In diesem Artikel werden überwiegend theoretische, teils auch empirische Gegenargumente entwickelt. In Abschnitt 2 wird begründet, weshalb ein geringer Lizenzpreis oder sogar der kostenlose Zugang zu F/OSS sehr wohl den volkswirtschaftlichen Ressourceneinsatz widerspiegelt. Dabei wird auch erläutert, weshalb die zu proprietären Lösungen deutlich verschiedene Preisstruktur bei F/OSS-Geschäftsmodellen keine Verzerrung darstellt. In Abschnitt 3 wird ausgeführt, weshalb selbst dann, wenn man F/OSS-Geschäftsmodelle als eine Form der „Quersubventionierung“ begreift, diese Preisstruktur wohlfahrtsökonomisch überlegen ist. Schließlich wird in Abschnitt 4 die Gegenfrage gestellt, wie sich denn der (zunehmende) Markterfolg von F/OSS erklären lässt, wenn doch diese Produktionsweise angeblich ökonomisch ineffizient ist. Wettbewerbliche Märkte müssten ja eigentlich dafür sorgen, dass ineffiziente Anbieter verdrängt werden. Abschnitt 5 beschäftigt sich mit einem abstrakt-theoretischen Argument, nämlich der Unmöglichkeit, Ineffizienzen aus freiwilligen Tätigkeiten abzuleiten, die zu keinen negativen externen Effekten führen. Schließlich wird in Abschnitt 6 die Fragestellung genau umgekehrt, indem einige Argumente skizziert werden, nach denen F/OSS das effiziente Funktionieren der Märkte für Software nicht nur nicht behindert, sondern sogar fördert.

## **2 Werden für F/OSS keine kostendeckenden Preise bezahlt?**

Freie Software (FS) zeichnet sich dadurch aus, dass dem Nutzer zusätzliche Freiheiten und damit Verfügungsrechte eingeräumt werden: Die Software für beliebige Zwecke zu nutzen sowie den Code einzusehen, zu modifizieren und ohne Beschränkungen

weitergeben zu können. Damit ist nicht gesagt, dass FS zwingend kostenlos ist. Vertreter der Freie-Software-Community legen viel Wert auf die Feststellung, dass die Idee freier Software nicht anti-kommerziell ist.<sup>1</sup> Der Begriff der Open-Source-Software (OSS) besagt zunächst nur, dass der Quellcode offengelegt wird; die genannten Freiheiten im Umgang mit dem Quellcode sind damit noch nicht gewährleistet. Allerdings gibt es auch Definitionen von OSS, die der von FS sehr ähnlich sind,<sup>2</sup> so dass im Folgenden zusammenfassend von F/OSS gesprochen wird.

Nun ist die Feststellung, dass „frei“ oder „offen“ nicht unbedingt „kostenlos“ bedeutet, für einen großen Teil von F/OSS nur sehr bedingt relevant. Die Rechte, die dem Nutzer eingeräumt werden, führen dazu, dass jeder, der eine F/OSS-Kopie kommerziell erworben hat, weitgehende Verfügungsrechte an dem Code hat und zu einem Reseller werden könnte, um zumindest einen Teil seiner Lizenzkosten wieder hereinzuholen. Das liegt daran, dass auf die Anwendung des Ausschlussprinzips bewusst verzichtet wird. Da sich der Code aber nahezu kostenlos replizieren lässt und die Kopien vollkommen identisch sind, würde dies bei Preiswettbewerb zu einem „race to the bottom“ führen: Der Preis für eine F/OSS-Lizenz würde schnell in Richtung der Grenzkosten<sup>3</sup>, nämlich nahezu Null konvergieren. In der Tat ist es so, dass der Quellcode und häufig auch vorkompilierte Pakete bestehender F/OSS im Internet frei und dadurch tatsächlich kostenlos zur Verfügung stehen. Die öffentliche Zugänglichkeit des Codes ist sogar für das Funktionieren vieler F/OSS-Projekte von essentieller Bedeutung, weil sich auf diese Weise eine breite Community an dem Entwicklungsprozess beteiligen kann.

Der Erfolg kommerzieller F/OSS-Geschäftsmodelle beruht darauf, dass es eine Zahlungsbereitschaft der Kunden für „Problemlösungen“, d. h. für Güterbündel gibt, die aus der Software und weiteren Leistungen wie z. B. Beratung, Support, kundenspezifische Anpassungen usw., d. h. den sogenannten Komplementärgütern bestehen (vgl. Leiteritz 2004). Auf diesem Weg ist es möglich, F/OSS-Lizenzen *im Paket* mit den Komplementärgütern kommerziell zu vertreiben. Die Zahlungsbereitschaft der Kunden, die über die Kosten der Komplementärgüterherstellung hinausgeht, trägt zur Finanzierung möglicher Beiträge des Anbieters zu F/OSS-Projekten bei. Sofern solche Geschäftsmodelle Gewinn abwerfen, sind offenbar auch die F/OSS-Beiträge voll finanziert. Der Preis, den der Kunde für das Güterbündel zahlt, spiegelt daher sehr wohl den Ressourcenverzehr wider, den der Anbieter finanzieren muss. Nach Schätzungen der *Boston Consulting Group* werden F/OSS-Projekte etwa zur Hälfte auf dieser kommerziellen Basis erstellt (vgl. Boston Consulting Group und Open Source Technology Group 2002). Obwohl die Kunden prinzipiell die Möglichkeit hätten, die Software als quasi-öffentliches Gut kostenlos aus dem Internet zu beziehen, werden

---

1 Siehe <http://www.fsf.org>.

2 Vgl. <http://www.opensource.org>.

3 Unter Grenzkosten versteht man diejenigen zusätzlichen Kosten, welche durch die Herstellung der letzten Gütereinheit entstehen.

sie durch die Bündelung mit den für sie notwendigen Komplementärgütern faktisch gezwungen, ihre positive Zahlungsbereitschaft auch für die Softwarekomponente offenzulegen und sich so an den Entwicklungskosten zu beteiligen. Im Resultat besteht hierbei kein wesentlicher Unterschied zu Geschäftsmodellen mit proprietärer Software. Man kann also feststellen, dass bei F/OSS-Geschäftsmodellen der Preis die volkswirtschaftlichen Kosten deckt. Noch offensichtlicher ist dies der Fall, wenn es sich um kundenspezifische Aufträge zur Entwicklung von F/OSS handelt, die ein einzelner Kunde selbst voll finanziert.

Wie sieht es aber mit dem immer noch erheblichen Anteil der von einer verstreuten Community in der Freizeit, also nicht in kommerzieller Absicht, erstellten F/OSS aus? Diese Programmierer werden nicht über den Preismechanismus für ihre Faktorleistung entlohnt. Auf den ersten Blick scheint das eingangs zitierte Argument wenigstens für diese F/OSS-Produktionsform zuzutreffen. Dagegen lässt sich Folgendes einwenden: Freiwillige Entwickler haben unterschiedliche Motive, z. B. Spaß am Programmieren, Reputationserwerb oder schlicht das Motiv, ein Problem zu lösen, für das es am Markt keine adäquate Lösung gibt. Freiwilligen Entwicklern muss Rationalität unterstellt werden. Sie werden daher nur dann ohne Entgelt einen Entwicklungsbeitrag leisten und dafür Opportunitätskosten<sup>4</sup> auf sich nehmen, wenn sie sich davon einen positiven Nettonutzen versprechen. Wenn sie folglich einen Beitrag zu einem F/OSS-Projekt leisten, so ist der volkswirtschaftliche Ressourcenverzehr bereits entschädigt. Die Entwicklungskosten sind bereits internalisiert, die Grenzkosten sind mit einem Preis von Null ebenfalls abgegolten. Insofern ist auch die kostenlose Distribution von F/OSS vollständig kompatibel mit den volkswirtschaftlichen Effizienzbedingungen (siehe dazu Abschnitt 5).

Aus der empirischen Beobachtung, dass einerseits für F/OSS-Lizenzen keine oder im Vergleich zu proprietären Produkten sehr moderate Lizenzgebühren bezahlt werden, andererseits mit F/OSS-Komplementärleistungen gutes Geld verdient werden kann, wird oft gefolgert, dass hier eine Preisverzerrung vorliege. Proprietäre Softwarehersteller argumentieren gerne mit TCO-Studien („Total Cost of Ownership“), nach denen den niedrigen Lizenzkosten für F/OSS deutlich höhere Kosten für Komplementärleistungen gegenüberstehen (etwa Giera 2004). Dies wird als Indiz gesehen, dass die Ressourcenallokation verzerrt würde, weil die Anreize zur professionellen Softwareentwicklung zu gering seien und die Ressourcen zu sehr in den Komplementärmarkt umgelenkt würden. Als Beleg für die dadurch erzeugten Ineffizienzen wird dann darauf verwiesen, dass auf längere Sicht die TCO bei F/OSS höher seien als bei proprietärer Software. Aus der oben angeführten Argumentation ist aber deutlich geworden, dass der geringe Marktpreis für F/OSS-Lizenzen volkswirtschaftlich völlig

---

4 Unter Opportunitätskosten versteht man die *entgangenen* Alternativen, wenn man sich für eine Alternative entschieden hat. Der freiwillige Programmierer *verzichtet* z. B. auf Zeit, die er auch anderweitig nutzen könnte bzw. auf Einkommen, wenn er die Möglichkeit hätte, in der Zeit entgeltlich zu arbeiten. Ein rationaler Ressourceneinsatz erfordert eine Minimierung der Opportunitätskosten.

korrekt ist. Er ist vor allem aus drei Gründen deutlich niedriger als der proprietärer Produkte:

- In F/OSS stecken immer noch erhebliche Anteile freiwillig erbrachter Leistungen, die keine Beteiligung des Kunden an den Opportunitätskosten der Produktion erfordern.
- Kommerzielle F/OSS-Unternehmen profitieren zum (großen) Teil von der Community. Dadurch werden die Entwicklungskosten breiter gestreut, d. h. der auf einen einzelnen Anbieter entfallende und zu finanzierende Fixkostenblock ist deutlich geringer.
- Die Märkte für F/OSS sind aufgrund der Eigenschaften von F/OSS kompetitiver als Märkte für proprietäre Software (vgl. Mundhenke 2005 und Abschnitt 6).

Als Gründe für die zum Teil in TCO-Studien behaupteten höheren Kosten für Komplementärleistungen können u. a. aufgeführt werden:

- Teile der Kosten sind oft Migrationskosten, die sich z. B. in einem erhöhten Schulungsbedarf niederschlagen (vgl. z. B. Wild und Herges 2000, S. 18). Solche Kosten entstehen aber nicht durch die *Verwendung* von F/OSS, sondern durch den *Umstieg*, weil dadurch im Umgang mit proprietärer Software erworbenes Humankapital teilweise entwertet wird. Wechselkosten, die teilweise strategisch erhöht werden können, könnten ebenso gut als Folgekosten der bisherigen proprietären Lösung aufgefasst werden.
- Durch die zusätzlichen Verfügungsrechte (Freiheit, den Code einzusehen, zu modifizieren und nach Belieben zu verwenden) entsteht für viele Kunden ein Zusatznutzen, der eine höhere Zahlungsbereitschaft rechtfertigt („Total Cost of Benefits“, Wild und Herges vgl. 2000, S. 25 ff.). Dies wird dadurch belegt, dass proprietäre Hersteller z. B. durch Shared-Source-Programme diese Aspekte nachzuahmen versuchen (vgl. Seemayer und Matusow 2005).

### 3 Volkswirtschaftlich schädliche Quersubventionierung?

Es sei nochmal das Argument aufgegriffen, dass bei F/OSS-Geschäftsmodellen die Kunden mit ihrer Zahlungsbereitschaft für ein Güterbündel die Erstellung eines (auch) kostenlos zugänglichen Softwareproduktes „quersubventionieren“. Aus Sicht dieser Kunden ist es irrelevant, wie sich der Gesamtpreis auf die Komponenten verteilt, weil sie nur eine Gesamtzahlungsbereitschaft für die Problemlösung insgesamt haben. Da es aber auch solche Nutzer gibt, die keine Komplementärleistungen benötigen bzw. diese selbst erstellen und daher keinen Preis für F/OSS bezahlen, besteht nun doch ein Unterschied zu proprietärer Software, wo auch „software only“ in der Regel nur gegen Zahlung eines Marktpreises zu haben ist. Da F/OSS an sich kostenlos verfügbar ist, könnte man die Interpretation der „Quersubventionierung“ gelten lassen, weil nur die

Nachfrager nach Komplementärgütern die Kosten der Softwareentwicklung tragen müssen. In diesem Abschnitt wird jedoch gezeigt, dass eine Querfinanzierung via Komplementärgüter keineswegs eine negativ zu beurteilende Verzerrung der Preisstruktur, sondern unter recht allgemeinen Bedingungen ein raffinierter Mechanismus zur wohlfahrtsoptimalen Regulierung ist (für technische Details vgl. Pasche 2005).

Die Nachfrager nach Softwarelizenzen können in zwei Gruppen aufgeteilt werden: Nachfrager, die Komplementärgüter benötigen (Gruppe I) und Nachfrager, die diese nicht benötigen bzw. selbst erstellen und nicht am Markt nachfragen (Gruppe II). Von den Kunden der Gruppe I hat jeder eine Zahlungsbereitschaft für die Problemlösung, bestehend aus Software und Komplementärgut. Aus diesen Zahlungsbereitschaften erhält man für Software und das Komplementärgut jeweils eine Nachfragekurve, die negativ von *beiden* Marktpreisen abhängt. Unterstellt man vereinfachend strikte Komplementarität, so werden beide Güter stets in einem bestimmten Verhältnis nachgefragt. Erhöht sich nun *einer* der Preise, so geht wegen der Komplementarität die Marktnachfrage nach *beiden* Gütern (im selben Verhältnis) zurück. Nun gibt es aber noch die Kunden der Gruppe II, welche ausschließlich Softwarelizenzen nachfragen. Steigt nun der Preis für Lizenzen, so geht die nachgefragte Menge von Gruppe I *und* Gruppe II nach Lizenzen zurück, bei den Komplementärgütern geht aber nur die Nachfrage der Gruppe I zurück. Umgekehrt reagiert bei einer Erhöhung des Komplementärgutpreises zwar die Gruppe I, aber nicht die Gruppe II in ihrer Lizenznachfrage. Man kann also festhalten, dass die Gesamtnachfrage nach Softwarelizenzen stärker auf den Lizenzpreis als auf den Komplementärgüterpreis reagiert. Unter dieser (hinreichenden) Annahme lässt sich nun analytisch herleiten, dass die „Quersubventionierungslösung“ bzw. das F/OSS-Geschäftsmodell wohlfahrtswirtschaftlich einer strikt getrennten Bepreisung von proprietären Softwarelizenzen und Komplementärgütern überlegen ist.

Die Begründung liegt darin, dass bei hohen Entwicklungsfixkosten und Grenzkosten nahe Null ein proprietärer Softwareanbieter ein sog. „natürliches Monopol“ hat: Jede weitere abgesetzte Lizenz senkt die Durchschnittskosten, weshalb es kosteneffizient ist, wenn es nur einen Anbieter gibt. Dieser hat dann aber eine problematische monopolähnliche Stellung. Diese Marktstellung ist nur in gewissem Maß durch Wettbewerber angreifbar, die ein ähnliches, aber eben nicht gleiches Gut anbieten (vgl. Abschnitt 4). Die damit verbundene Marktmacht<sup>5</sup> wird ein rationaler Anbieter ausnutzen und einen relativ hohen Lizenzpreis verlangen. Entsprechend gering wird die Nachfrage nach Lizenzen *und* Komplementärleistungen sein. Eine (fiktive) Regulierungsbehörde, die an einer volkswirtschaftlichen Wohlfahrtsmaximierung interessiert ist, würde den Monopolpreis regulieren. Die ideale, den Effizienzbedingungen genügende Regulierung auf einen Preis in Höhe der Grenzkosten<sup>6</sup> nahe Null hätte

---

5 Die Marktmacht lässt sich auch durch weitere Eigenschaften von Software, vor allem durch nachfrage-  
seitige Netzwerkeffekte begründen (vgl. Gröhn 1999).

6 Jeder Nachfrager kommt dann für die volkswirtschaftlichen Kosten auf, die die letzte, von ihm selbst

allerdings zur Folge, dass der Softwareanbieter seine Entwicklungsfixkosten nicht decken kann. Der Regulierer müsste also die Wohlfahrt maximieren unter der Nebenbedingung, dass die Fixkosten gedeckt werden. Als Gedankenexperiment kann man sich vorstellen, dass dies mit einer Steuer geschieht, die auf das Gut Software und/oder auf das Komplementärgut erhoben wird. Das erzielte Steueraufkommen soll dabei den Entwicklungsfixkosten entsprechen und dem regulierten Softwareanbieter zufließen. Es zeigt sich, dass die Lösung dieses Maximierungskalküls unter den genannten Bedingungen eine *Randlösung* ist: Die optimale Steuer würde ausschließlich auf das Komplementärgut erhoben werden. Dies ist aber im Resultat nichts anderes als ein F/OSS-Geschäftsmodell: Die Software selbst kann (von Nachfragern der Gruppe II) zu Grenzkosten, d. h. fast kostenlos bezogen werden. Die Kunden der Gruppe I finanzieren mit ihrer Komplementärgutnachfrage die Softwareentwicklung mit. Ob nun auf ihrer Rechnung steht, dass sie ein „überbeuertes“ Komplementärgut und dazu kostenlose Softwarelizenzen erhalten haben oder ob für beides ein positiver Preis ausgewiesen wurde, ist dabei völlig unerheblich.

Man könnte einwenden, dass Kunden der Gruppe II wesentlich stärker von diesem Geschäftsmodell profitieren als Gruppe I. Das ist richtig. Dennoch profitiert auch Gruppe I, da die Preissenkung auf dem Softwarelizenzmarkt stärker ausfällt als die Preiserhöhung auf dem Komplementärgütermarkt. Der Grund liegt zum einen darin, dass der monopolistische Preissetzungsspielraum nicht mehr ausgenutzt wird und zum anderen, dass sich die Nachfrage aufgrund der Regulierung insgesamt erhöht, was zu einem „Verdünnungseffekt“ für die umzulegenden Fixkosten führt.

#### 4 Markterfolg für ein ineffizientes Produktionsmodell?

Unabhängig von den unterschiedlichen Theorien darüber, *ob* bzw. *wie* und *warum* das „Produktionsmodell F/OSS“ (in)effizient funktioniert, lässt sich feststellen, *dass* F/OSS im Softwaremarkt existiert: In direkter Konkurrenz zu proprietärer Software behauptet sich F/OSS nicht nur, sondern gewinnt teilweise auch Marktanteile hinzu (vgl. z. B. Evans Data Corporation 2005; Netcraft 2005; OneStat 2005). Zeigt diese empirisch feststellbare Konkurrenzfähigkeit von F/OSS, dass es sich hierbei um ein effizientes Produktionsmodell handeln muss, weil es sonst nicht erfolgreich am Markt bestehen könnte? Oder ist es auf Softwaremärkten möglich, dass auch ineffiziente Formen der Softwareproduktion im Wettbewerb bestehen?

Bei der Beantwortung dieser Fragen soll nicht auf der Ebene des Verstehens oder Bewertens des Phänomens F/OSS argumentiert werden. Daher wird folgende Blackbox-Betrachtungsweise gewählt: Alle, die im Rahmen des „Produktionsmodells F/OSS“ an der Entwicklung einer Software beteiligt sind, werden gedanklich zu einer Organisation zusammengefasst und als Unternehmen A bezeichnet. Daneben befindet sich im betrachteten Markt noch ein weiteres Unternehmen, welches der

---

nachgefragte Gütereinheit verursacht hat.

Einfachheit halber mit *B* bezeichnet wird. Beide Unternehmen bieten jeweils eine Software und dazu passende, komplementäre IT-Dienstleistungen an. Es handelt sich dabei um inkompatible Systeme, d. h. die Software des Unternehmens *A* ist nicht sinnvoll mit der IT-Dienstleistung von *B* kombinierbar und umgekehrt. Welche internen Entlohnungs- und Organisationsstrukturen die Unternehmen aufweisen, ist unbekannt und für die folgenden Überlegungen auch völlig unerheblich. Wichtig ist, *dass* die Unternehmen je zwei Produkte anbieten, unwichtig dagegen, *wie* diese beiden Produkte konkret hergestellt werden.

Die Konsumenten einer Software fragen jeweils unterschiedliche Mengen IT-Dienstleistungen nach. Zur Vereinfachung wird im Folgenden erneut nur zwischen solchen Konsumenten unterschieden, die eine Software und *keine* IT-Dienstleistungen kaufen (Gruppe II) und solchen, die Software *und* IT-Dienstleistungen erwerben (Gruppe I). Auf dem Teilmarkt der Gruppe I bieten *beide* Unternehmen letztlich jeweils ein Komplementärgüter-Paket aus *Software und IT-Dienstleistung* an, und zwar offensichtlich zu Konditionen, die aus Sicht der Konsumenten konkurrenzfähig sind. Das heißt, es gelingt dem Unternehmen *A*, das Komplementärgüter-Paket so zu produzieren, dass es mit Unternehmen *B* bezüglich des angebotenen Preis-Leistungsverhältnisses konkurrieren kann. Dies ist insofern erstaunlich, als dass *A* sein Softwareprodukt *auch* unentgeltlich zur Verfügung stellt, die Software-Entwicklungskosten bei *A* also allein von den Nachfragern der Gruppe I getragen werden (Stichwort „Quersubventionierung“, vgl. Abschnitt 3). Sofern man davon ausgeht, dass die Konsumenten der Gruppe II, welche die Software vom Unternehmen *A* kostenlos erhalten, generell bereit wären, für diese Software auch einen Preis zu bezahlen, verzichtet *A* hier also auf eine Einnahmequelle. Die Vermutung liegt daher nahe, dass das Unternehmen *A* – das Produktionsmodell F/OSS – aus der Freigabe der Software irgendeinen Vorteil zieht, der z. B. die Produktionskosten mindert und so den Einnahmeverlust kompensiert. Dies kann erklären, weshalb sich das Komplementärgüter-Paket von *A* am Markt halten kann, also nicht auf Grund der fehlenden Einnahmen „übersteuert“ angeboten werden muss. Aber auch unabhängig davon, wie man die unterschiedliche Preispolitik der beiden Unternehmen interpretiert, gilt, dass *beide* konkurrierende Unternehmen dem Wettbewerbsdruck standhalten, dass also keines der beiden als ineffizient vom Markt aussortiert wird.

Geht man zunächst einmal davon aus, dass der Wettbewerb auf Softwaremärkten nicht in irgendeiner Weise eingeschränkt wird, dann können Unternehmen – egal welchen „Typs“ – langfristig keine wesentliche Marktmacht ausüben, und ineffiziente Unternehmen werden über kurz oder lang vom Marktmechanismus aussortiert. Der Markterfolg des Unternehmens *A* ist in diesem Fall ein „Effizienzbeweis“, da der Konkurrenzdruck dafür sorgt, dass die Unternehmen ständig bestrebt sein *müssen*, möglichst günstig zu produzieren, die Kundenwünsche zu erfüllen und neue, verbesserte Produkte bzw. Produktvarianten zu entwickeln. Warum es unterschiedliche Produktionsmodelle gibt, wie sie funktionieren und welche mutmaßlichen Effizienzzei-

genschaften theoretisch begründet werden können, ist dabei völlig unerheblich. Der Vorteil eines marktwirtschaftlichen Systems und des „Wettbewerbs als Entdeckungsverfahren“ (von Hayek 1969) ist es ja gerade, dass keine zentrale Institution über die Vor- und Nachteile bestimmter Güter und Produktionsformen entscheidet, sondern der Wettbewerbsprozess selbst die unvorteilhaften Güter und Verfahren „entdeckt“ und aussortiert. Ein dauerhafter Erfolg eines ineffizienten Produktionsmodells bzw. eines ineffizienten Unternehmens in einem letztlich voll funktionsfähigen Markt ist nicht denkbar.

Ineffizienzen bei F/OSS lassen sich *trotx* funktionierender Märkte und effizienter Unternehmen nur dann postulieren, wenn die Ressourcenverschwendung in einem Bereich erfolgt, der dem Marktmechanismus entzogen ist. Die Ineffizienzen wären dann sozusagen „unsichtbar“, weil sie sich nicht in monetären Größen niederschlagen. Da Unternehmen A die auftretenden Ineffizienzen also nicht als Kosten tragen muss, sind die Preise verzerrt und dies verzerrt den Wettbewerb. Es besteht sogar die Gefahr, dass die nur *scheinbar* wettbewerbsfähige F/OSS die effizient arbeitenden proprietären Anbieter verdrängt. Die Ressourcenverschwendung findet dann also im Bereich der freiwilligen F/OSS-Entwicklung statt. Dabei ist es unerheblich, dass Software unentgeltlich programmiert oder weitergegeben wird, es kommt allein darauf an, dass mehr (Opportunitäts-)Kosten als nötig verursacht werden. Auf die Plausibilität der Annahme von Ineffizienz durch „freiwillige“ Tätigkeiten wird im Abschnitt 5 gesondert eingegangen.

Verwirft man die Annahme von Ineffizienz durch „freiwillige“ Tätigkeiten und will dennoch die These des ineffizienten F/OSS-Produktionsmodells aufrechterhalten, so lässt sich die Existenz von F/OSS nur damit erklären, dass der Wettbewerb auf Softwaremärkten systematisch eingeschränkt ist. Aus ökonomischer Sicht weist Software durchaus einige Eigenschaften auf, die die These eines beschränkt funktionsfähigen Wettbewerbs stützen können (vgl. Gröhn 1999; Fichert 2002; Kooths et al. 2003; Quah 2003; Pasche und von Engelhardt 2004; Mundhenke 2005). Zu diesen Eigenschaften gehören u. a.:

1. Software ist ein *Erfahrungsgut im herkömmlichen Sinne*, weil sich einige Eigenschaften typischerweise erst im Verlauf der Nutzung herausstellen, Kaufentscheidungen also immer Entscheidungen unter Unsicherheit sind. Dadurch werden in gewissem Maße suboptimale Entscheidungen getroffen, so dass z. B. eine Lösung gewählt wird, die zunächst günstiger *erscheint*, sich aber langfristig als die teurere Variante herausstellt. Softwareanbieter haben hier die Möglichkeit durch gezielte Informationspolitik die Erfahrungsguteigenschaft strategisch auszunutzen. Einmal getroffene Entscheidungen für eine Software werden zudem ein gewisses Beharrungsvermögen aufweisen, d. h. es werden lieber neue Versionen der bereits bekannten Software gekauft, als noch nicht bekannte Alternativen ausprobiert. Neuere Softwareprodukte, die keiner etablierten Produktlinie angehören, haben dann einen systematischen Nachteil im

Wettbewerb um die Konsumenten. Testberichte, TCO-Studien usw. können einerseits als (erfolgreiche) Versuche interpretiert werden, die Unsicherheit und damit die Fehlentscheidungen auf ein Minimum zu reduzieren. Andererseits sind sie deswegen ein geeignetes Medium strategischer Informationspolitik.

2. Die Wechselbereitschaft der Nutzer wird auch durch die *Erfahrungsgut-Eigenschaft* im einem weiteren, *zweiten Sinne* gemindert: Durch den täglichen Gebrauch der Software wird – neben allgemeinen Fähigkeiten – ganz spezielles, (produkt-)spezifisches Humankapital aufgebaut. Je unterschiedlicher die konkurrierenden Softwareprodukte hinsichtlich GUI, Programm- und Steuerungslogik, Befehlssystematik usw. sind, umso mehr Humankapital wird bei einem Wechsel der Software entwertet und muss neu aufgebaut (d. h. neu erlernt) werden. Dies mindert die Wechselbereitschaft bzw. aus Nutzersicht die Austauschbarkeit von zwei Softwareprodukten und damit die Intensität der direkten Konkurrenz dieser beiden Produkte.
3. Auch die Tatsache, dass viele Softwareprodukte *Netzwerküter* sind, mindert die Austauschbarkeit aus Nutzersicht: Häufig steigt der Nutzen einer Software, je mehr installierte Einheiten es gibt, da dies z. B. den Erfahrungsaustausch erleichtert, Vorteile der Datenkompatibilität zum Tragen kommen, sich die Wahrscheinlichkeit erhöht, dass Komplementärprodukte entwickelt werden usw. Bei inkompatiblen Netzwerken verursacht ein Wechsel Kosten in Form entgangener Netzwerkvorteile. Dies mindert die Wettbewerbsintensität und gibt dem Inhaber des dominierenden Netzwerkes/Standards eine gewisse Marktmacht. Netzwerkeffekte führen ggf. zu *Pfadabhängigkeiten* und *Lock-In-Effekten*. So kann es zu einer dauerhaften Dominanz der inferioren (d. h. der vergleichbar schlechteren) Technologie kommen: Auch wenn sich alle Nutzer mit einem Wechsel zum neuen Softwarestandard besser stellen würden, so ist der Umstieg für den einzelnen Nutzer nachteilig, wenn lediglich ein kleiner Teil der Nutzer den Wechsel vollzieht, da dann der Vorteil der besseren Software vom Nachteil des viel kleineren Netzwerkes überkompensiert wird. Kann ein kollektiver Wechsel nicht verbindlich „verabredet“ werden, hat jeder Nutzer den Anreiz abzuwarten, bis genügend andere Nutzer den neuen Standard übernommen haben. Wenn aber alle (oder der überwiegende Teil der) Nutzer abwarten, dann kommt es nie zu einer hinreichend großen Nutzerzahl im neuen System und ein System-Wechsel findet nicht statt (sog. *Pinguin-Effekt*<sup>7</sup>). Erwartungen über die zukünftige Entwicklung von Netzwerken spielen also eine große Rolle, und Unternehmen könnten versuchen, durch geschickte Informationspolitik

---

7 Dieser Begriff leitet sich ab von hungrigen Pinguinen, die anstatt zu fischen auf einer Eisscholle ausharren, weil aus Angst vor möglichen Raubtieren keiner als Erster ins Wasser springen möchte (vgl. Farrell und Saloner 1987, S. 13 f.).

sowie mit geeigneten PR-/Werbemaßnahmen die Erwartungen der Nutzer zu beeinflussen und damit selbsterfüllende Prophezeiungen zu generieren.

Wenn Softwaremärkte also typischerweise ganz bestimmte Funktionsdefizite aufweisen,<sup>8</sup> kann damit das Überleben eines ineffizienten Unternehmens *A* im Markt erklärt werden. Ist dies der Fall, so haben allerdings auch Produzenten von proprietärer Software (Unternehmen *B*) die Möglichkeit, sich Ineffizienzen zu leisten: Lassen sich auf Softwaremärkten systematisch gravierende Funktionsdefizite erkennen, so profitieren *alle* am Markt vertretenen Unternehmen von diesem geminderten Wettbewerbsdruck und werden diesen Spielraum auch ausnutzen. Ob sich dies dann in Monopolgewinnen, nicht-optimaler Produktionsorganisation oder sonstigen Ineffizienzen äußert, mag vom „Typus“ des Unternehmens abhängen, ist jedoch für die Bewertung des Ergebnisses letztlich unerheblich. Es gilt also erneut, dass der „Wettbewerb als Entdeckungsverfahren“ neben der proprietären Variante mit F/OSS ein Produktionsmodell entwickelt hat, das sich unter den Bedingungen, die auf Softwaremärkten herrschen, zu behaupten vermag. Wer systematisch Funktionsdefizite auf Softwaremärkten erkennt, kann dies also nicht als Argument gegen F/OSS benutzen, sondern ist vielmehr angehalten, aus dieser Erkenntnis Handlungsempfehlungen für die Wirtschaftspolitik abzuleiten, mit der Maßgabe, den Wettbewerb im Softwaremarkt insgesamt zu intensivieren, wie beispielsweise Fichert (2002) vorschlägt.

## 5 Lassen sich Ineffizienzen durch „freiwillige“ Tätigkeiten begründen?

Jeder – durch welche Motive auch immer begründete – Beitrag zu einem F/OSS-Projekt wird aufgrund rationaler Entscheidungen geleistet (vgl. Luthiger 2004). Im Rahmen kommerzieller Geschäftsmodelle ist dies unmittelbar einsichtig, denn der Programmierer bezieht ein Faktoreinkommen und der Unternehmer erzielt Einnahmen, welche die Kosten mindestens decken. Aber auch bei freiwilligen Beiträgen, die in der Freizeit geleistet werden, liegen letztlich rationale Entscheidungen vor. Ist der Programmierer intrinsisch motiviert, weil er Spaß am Programmieren hat, seine Fähigkeiten und Kenntnisse erweitern will, von der Open-Source-Idee überzeugt ist oder Reputation erwerben will, so reflektiert die Entscheidung, ohne Entgelt einen F/OSS-Beitrag zu erbringen, dass er dennoch einen positiven Nettonutzen aus dieser Tätigkeit zieht. Ansonsten ist sein Verhalten (zumindest im traditionellen Sinn) ökonomisch nicht erklärbar. Dieser Nettonutzen ergibt sich aus dem Nutzen der

---

8 Die Frage, wie wahrscheinlich bzw. wann und in welchem Ausmaß der Wettbewerb auf Softwaremärkten tatsächlich eingeschränkt ist, soll an dieser Stelle nicht beantwortet werden. Es sei zum einen auf die oben genannten Quellen verwiesen. Zum anderen sehen die Autoren hier noch weiteren Forschungsbedarf und die Notwendigkeit der Entwicklung einer empirisch robusten, differenzierten Theorie der Ökonomik von Softwaremärkten.

Programmiertätigkeit abzüglich der (volkswirtschaftlichen) Kosten, die der Programmierer trägt. Dazu gehören z. B. die Opportunitätskosten der entgangenen Zeit oder das entgangene Einkommen, das er im Rahmen kommerzieller Softwareentwicklung (oder auch durch andere Tätigkeiten) in dieser Zeit hätte erzielen können. All diese Nutzen und Kosten sind bereits abgewogen und in seine Entscheidung „eingepreist“. Trägt er freiwillig und unentgeltlich zu dem F/OSS-Projekt bei, so existieren keine weiteren volkswirtschaftlichen Kosten, die erst über einen Preismechanismus kompensiert werden müssten. Eine Weitergabe des Gutes in Höhe der Grenzkosten von Null entspricht daher den volkswirtschaftlichen Effizienzbedingungen.

Eine Verzerrung des Systems relativer Preise durch freiwillige Tätigkeiten kann man nur dann begründen, wenn diese Tätigkeiten negative externe Effekte hätten. Dazu müsste Dritten irgendeine Form von Kosten aufgebürdet werden, für die sie nicht über das Preissystem kompensiert werden (vgl. Lancaster 1996). Ein bekanntes Beispiel für negative externe Effekte sind etwa Umweltbelastungen, die durch die Produktion oder den Konsum bestimmter Güter entstehen und dadurch die Nutzung der Umwelt durch Dritte beeinträchtigen. Es ist nicht ersichtlich, dass es bei der Programmierung von F/OSS zu negativen Externalitäten kommt. Im Gegenteil, der Erwerb von Fähigkeiten und Kenntnissen (Humankapital) könnte die Produktivität des freien Programmierers auch bei seiner Erwerbsarbeit erhöhen. Dadurch würde der kommerzielle Sektor von Humankapitalinvestitionen profitieren, für die er selbst nicht aufkommen musste. Viel wichtiger ist aber, dass durch die Wiederverwendbarkeit von offen gelegtem Code in verschiedenen Kontexten und somit die Offenlegung bestimmter Problemlösungstechniken *positive* externe Effekte (sog. *Wissensspillovereffekte*) entstehen. Den F/OSS-Produktionsmodellen gelingt es, auf die Durchsetzung des Ausschlussprinzips zu verzichten und dennoch den Anreiz aufrechtzuerhalten, F/OSS zu entwickeln. Dadurch wird ein quasi-öffentliches Gut erstellt, nämlich der in F/OSS verkörperte, öffentlich zugängliche Wissenskapitalstock. Dies ist bei proprietärer Produktionsweise nicht in diesem Maße möglich, denn selbst bei Varianten, wie z. B. Shared-Source-Software, ist der Zugang zu und die Nutzung des Wissens stark eingeschränkt.

## 6 Kann Open-Source-Software die Effizienz der Märkte erhöhen?

Die Funktionsfähigkeit und Effizienz kommerzieller Märkte wird durch das Eindringen von F/OSS in den Markt nicht bedroht. Dafür gibt es keinerlei empirische Hinweise, und auch die genannten theoretischen Argumente sprechen dagegen. Im Gegenteil: Sowohl die qualitativen Eigenschaften von F/OSS als auch die damit verbundenen veränderten Preisstrukturen üben einen Wettbewerbsdruck aus, der aufgrund der besonderen Charakteristika des Gutes Software in kommerziellen Märkten eher unterentwickelt ist. Es gibt eine Reihe von Anhaltspunkten, die für eine *Verbesserung*

zung der Funktionsfähigkeit des Wettbewerbs durch F/OSS sprechen:

- Die Produktionsweise von F/OSS ist ressourcenschonend. Zum einen werden freiwillige Entwicklungsleistungen und somit verstreut vorhandenes und vom Markt nicht voll erschließbares Humankapital genutzt. Zum anderen betreiben kommerzielle F/OSS-Unternehmen durch ihre begrenzten, aber öffentlich zugänglichen Entwicklungsbeiträge ein *cost sharing*, welches in gewisser Hinsicht mit Forschungsk Kooperationen vergleichbar ist. Dies alles schlägt sich in niedrigen Lizenzkosten nieder.
- F/OSS bringt zahlreiche neue Produkte auf den Markt, die aufgrund der zusätzlichen Freiheiten (Verfügungsrechte) dem Kunden einen Zusatznutzen bieten. Die stärkere Differenzierung der Produkte und die größeren Wahlfreiheiten ermöglichen eine engere Anpassung an die Kundenpräferenzen.
- Durch Senkung der Markteintrittsbarrieren, nicht an einzelne Hersteller gebundene Netzwerkexternalitäten und offene Standards wird der Wettbewerb intensiviert. Dieser erhöhte Wettbewerbsdruck diszipliniert zudem auch Hersteller proprietärer Software.

Der Umstand, dass im Zuge des verschärften Wettbewerbs auch zahlreiche Geschäftsmodelle kleiner und mittlerer Unternehmen, welche auf proprietärer Software aufbauen, unter Druck geraten und somit vielleicht Umsatz- und Arbeitsplatzrückgänge möglich sind, kann nicht als Folge einer zerstörerischen Wettbewerbsverzerrung durch F/OSS interpretiert werden. Das ist der normale Gang der Dinge in einer durch Produkt- und Prozessinnovationen gekennzeichneten wettbewerblichen Marktwirtschaft, wie ihn bereits Schumpeter (1942) als „Prozess der schöpferischen Zerstörung“ beschrieben hat.

## 7 Zusammenfassung

Obwohl F/OSS in der Regel kostenfrei zugänglich ist, führt die Bündelung mit notwendigen Komplementärprodukten dazu, dass die Kunden ihre Zahlungsbereitschaft am Markt offenbaren und an den Softwareentwicklungskosten beteiligt werden. Im Fall freiwilliger F/OSS-Beiträge sind die volkswirtschaftlichen Kosten bereits gedeckt, weil es ohne einen positiven Nettonutzen nicht zu diesen freiwilligen Leistungen gekommen wäre. Die Preisstruktur, die sich deutlich von der Preisstruktur proprietärer Lösungen unterscheidet, spiegelt sehr wohl den volkswirtschaftlichen Ressourceneinsatz wider und stellt auch keine Verzerrung dar. Sie ist lediglich Ausdruck einer völlig anderen Produktionsform.

Selbst die Interpretation der Preisstruktur bei F/OSS-Geschäftsmodellen als „Quersubventionierung“ ist kein Argument für eine schädliche Verzerrung der relativen Preise. Es lässt sich im Gegenteil zeigen, dass Geschäftsmodelle, die durch

Institutionen wie F/OSS-Lizenzmodelle abgesichert sind, die Wohlfahrt erhöhen. Oder verkürzt gesagt: Die GPL und andere Lizenzen erfüllen eine ähnliche Funktion wie eine wohlfahrtsmaximierende Regulierungsbehörde.

Wenn das auf F/OSS aufbauende Güterbündel ineffizient hergestellt werden würde, dann ließe sich der zunehmende Markterfolg nur durch Funktionsdefizite von Softwaremärkten erklären. Der Markterfolg von F/OSS bedeutet also entweder, dass die Produktion des Güterbündels durchaus effizient und somit wettbewerbsfähig ist oder aber, dass Softwaremärkte – bedingt durch die besonderen Eigenschaften des Gutes Software – ohnehin an Ineffizienzen leiden, so dass sich mit dem F/OSS-Modell lediglich die Gründe für Ineffizienzen ändern, aber nicht deren Ausmaß. Die These, dass am Markt erfolgreiche F/OSS im Vergleich zur proprietären Variante signifikant ineffizienter produziert wird, lässt sich nur dann begründen, wenn die Ineffizienz im Bereich der nicht monetär entlohnten F/OSS-Entwicklung auftritt. Ineffizienzen aufgrund freiwilliger, nicht über das Preissystem entgoltenen Leistungen sind allerdings nur dann theoretisch begründbar, wenn sie mit negativen externen Effekten verbunden sind. Dies ist bei F/OSS nicht der Fall, es bestehen im Gegenteil sogar positive Externalitäten (*Wissensspillovereffekte*).

Zudem gilt, dass bei F/OSS zusätzliche Ressourcen aktiviert werden und durch die spezielle Produktionsweise *cost sharing* ermöglicht wird, diesem Produktionsmodell also in Teilbereichen sogar Effizienzvorteile zugesprochen werden können. Hinzu kommt, dass bei F/OSS z. B. die Markteintrittsbarrieren geringer sind. Ganz allgemein gilt, dass in einem Markt jedes neue Produkt bzw. jeder neuer Anbieter das Produktangebot erhöht bzw. weiter ausdifferenziert, so dass der Wettbewerb – hier am Softwaremarkt – insgesamt intensiviert wird.

## Literatur

- Berends, P. und van Wegberg, M. (2000), *Competing Communities of Users and Developers of Computer Software. Competition Between Open Source Software and Commercial Software*, NIBOR Working Paper 1, Netherlands Institute of Business Organization and Strategy Research, Maastricht.
- Bitzer, J. (2000), *Erosion of Monopoly Power due to the Emergence of Linux*, DIW Diskussionspapiere 231, Deutsches Institut für Wirtschaftsforschung, Berlin.
- Blankart, C. B. und Knieps, G. (1992), *Netzökonomik*, in E. Böttcher (Hrsg.), 'Ökonomische Systeme und ihre Dynamik', *Jahrbuch für neue politische Ökonomie*, J.C.B. Mohr (Paul Siebeck), Tübingen.
- Boston Consulting Group und Open Source Technology Group (2002), 'The Boston Consulting Group Hacker Survey, Release 0.73', <http://www.osdn.com/bcg/> [28. Jan 2006]. Präsentation auf der O'Reilly Open Source Conference.
- Evans Data Corporation (2005), 'Database Developers Embrace Open Source Database

## Führt Open-Source-Software zu ineffizienten Märkten?

- Solutions, New Evans Data Survey?  
[http://www.evansdata.com/n2/pr/releases/EDCDB05\\_02.shtml](http://www.evansdata.com/n2/pr/releases/EDCDB05_02.shtml) [28. Jan 2006].
- Farrell, J. und Saloner, G. (1987), Competition, Compatibility and Standards. The Economics of Horses, Penguins and Lemmings, in H. L. Gabel (Hrsg.), 'Product Standardization and Competitive Strategy', North Holland, Amsterdam, S. 1–21.
- Fichert, F. (2002), 'Wettbewerbspolitik im digitalen Zeitalter. Öffnung vermachteter Märkte virtueller Netzwerküter', Beitrag zum 3. Workshop 'Ordnungsökonomik und Recht' des Walter Eucken Instituts.  
<http://www.walter-eucken-institut.de/veranstaltungen/workshop2002/Fichert-Paper.pdf> [28. Jan 2006].
- Giera, J. (2004), The Costs and Risks of Open Source, Report, Forrester Research.
- Gröhn, A. (1999), *Netzwerkeffekte und Wettbewerbspolitik – Eine ökonomische Analyse des Softwaremarktes*, J.C.B. Mohr (Paul Siebeck), Tübingen.
- Kooths, S., Langenfurth, M. und Kalwey, N. (2003), *Open-Source-Software – Eine volkswirtschaftliche Bewertung*, Vol. 4 of *MICE Economic Research Studies*, Muenster Institute for Computational Economics, Münster.
- Lancaster, K. J. (1996), *Trade, markets and welfare*, Edward Elgar, Cheltenham.
- Leiteritz, R. (2004), Open Source-Geschäftsmodelle, in B. Lutterbeck und R. A. Gehring (Hrsg.), 'Open Source Jahrbuch 2004. Zwischen Softwareentwicklung und Gesellschaftsmodell', Lehmanns Media, Berlin, S. 153–186.  
<http://www.opensourcejahrbuch.de/2004/> [10. Feb 2006].
- Luthiger, B. (2004), Alles nur Spaß? Zur Motivation von Open-Source-Entwicklern, in B. Lutterbeck und R. A. Gehring (Hrsg.), 'Open Source Jahrbuch 2004. Zwischen Softwareentwicklung und Gesellschaftsmodell', Lehmanns Media, Berlin, S. 107–120.  
<http://www.opensourcejahrbuch.de/2004/> [10. Feb 2006].
- Mundhenke, J. (2005), Ökonomische Eigenschaften von Software – Die Bedeutung von Open-Source-Software für den Wettbewerb auf Softwaremärkten, in B. Lutterbeck, R. A. Gehring und M. Bärwolff (Hrsg.), 'Open Source Jahrbuch 2005 – Zwischen Softwareentwicklung und Gesellschaftsmodell', Lehmanns Media, Berlin, S. 143–160.  
<http://www.opensourcejahrbuch.de/2005/> [10. Feb 2006].
- Netcraft (2005), 'December 2005 Web Server Survey'. [http://news.netcraft.com/archives/2005/12/02/december\\_2005\\_web\\_server\\_survey.html](http://news.netcraft.com/archives/2005/12/02/december_2005_web_server_survey.html) [28. Jan 2006].
- OneStat (2005), 'Mozilla's browsers global usage share is still growing according to OneStat.com'. [http://www.onestat.com/html/aboutus\\_pressbox40\\_browser\\_market\\_firefox\\_growing.html](http://www.onestat.com/html/aboutus_pressbox40_browser_market_firefox_growing.html) [28. Jan 2006].
- Pasche, M. (2005), (Self-)Regulation of a Natural Monopoly via Complementary Goods – The Case of F/OSS Business Models, Diskussionspapier 18/2005, Friedrich-Schiller-Universität, Wirtschaftswissenschaftliche Fakultät, Jena.

*Markus Pasche und Sebastian von Engelhardt*

- Pasche, M. und von Engelhardt, S. (2004), Volkswirtschaftliche Aspekte der Open-Source-Softwareentwicklung, Diskussionspapier 18/2004, Friedrich-Schiller-Universität, Wirtschaftswissenschaftliche Fakultät, Jena.
- Quah, D. (2003), Digital Goods and the New Economy, CEP Discussion Papers 563, London School of Economics, London.
- Schumpeter, J. A. (1942), *Capitalism, Socialism and Democracy*, Harper & Brothers, New York.
- Seemayer, W. und Matusow, J. (2005), Das Microsoft-Shared-Source-Programm aus der Business-Perspektive, in B. Lutterbeck, R. A. Gehring und M. Bärwolff (Hrsg.), 'Open Source Jahrbuch 2005 – Zwischen Softwareentwicklung und Gesellschaftsmodell', Lehmanns Media, Berlin, S. 185–200. <http://www.opensourcejahrbuch.de/2005/> [10. Feb 2006].
- Shapiro, C. und Varian, H. R. (1999), *Information Rules. A Strategic Guide to the Network Economy*, Harvard Business School Press, Boston.
- Wild, M. und Herges, S. (2000), Total Cost of Ownership (TCO) – Ein Überblick, techn. Bericht 1/2000, Universität Mainz.
- von Hayek, F. A. (1969), Der Wettbewerb als Entdeckungsverfahren, in F. A. von Hayek (Hrsg.), 'Freiburger Studien', J.C.B. Mohr (Paul Siebeck), Tübingen, S. 249–265.