

# Wie Pech und Schwefel

In der Wissensgesellschaft entstehen Produkte auch jenseits von Unternehmen – ein Essay von Manfred Langen.

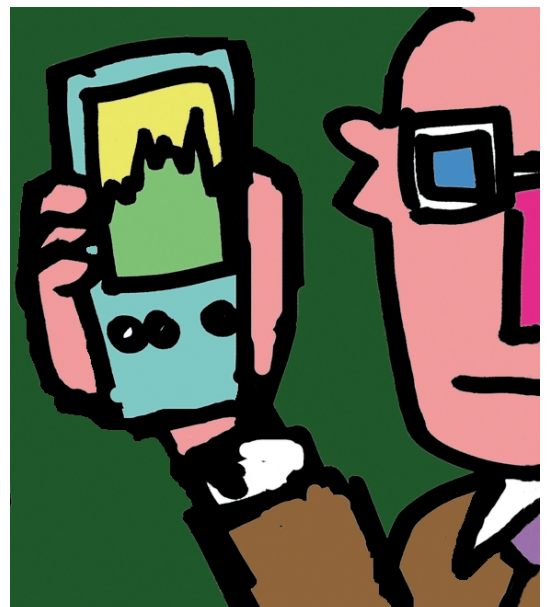
***Open Source Software-Entwickler sind quer über den Erdball aktiv. Ohne direkten Auftrag- oder Geldgeber arbeiten sie in virtuellen Teams zusammen. Sie kennen sich meist nicht persönlich und treffen sich so gut wie nie. Sie kooperieren trotzdem und helfen sich gegenseitig. Viele Köpfe sorgen für höchste Qualität. Was treibt sie an, was hält sie zusammen? Fragen, die bisher noch wenig untersucht oder beachtet wurden. Ein neues Management-Paradigma wird sichtbar. Alle Akteure verstehen sich als Freiwillige, die sich auf ein gemeinsames Ziel hin selbst verpflichtet haben und an der Steuerung ebenso beteiligt sein wollen wie am Erfolg.***

**W**ettbewerbsfähige Produkte – hergestellt von einem Verbund freiwilliger Software-Entwickler? Noch vor zehn Jahren hätten Wirtschaftsexperten kräftig den Kopf geschüttelt. Denn echte Produktqualität, so hieß es damals unisono, kann nur im Umfeld eines ordentlichen Unternehmens entstehen. Die Zeiten aber haben sich geändert. Software-Entwickler arbeiten heute ohne direkten Auftrag- oder Geldgeber quer über den Erdball zusammen. Ihre gemeinsame Losung lautet Open Source Software.

Diese Entwicklung hat zu einer Aufspaltung in der IT-Welt geführt: Auf der einen Seite stehen einige superprofitable Software-Firmen wie SAP, Oracle oder Microsoft, die Renditen von 25 bis über 40 Prozent erwirtschaften, auf der anderen Seite steht ein Freiwilligen-Netzwerk, das in kollektiver Schaffensfreude die Nutzung ihrer Werke kostenfrei zur Verfügung stellt. Open Source Software ist nicht nur in technologischer Hinsicht ein wichtiges Thema geworden, nein, es ist natürlich von höchstem Interesse, wie Motivation und Koordination in diesen losen Projektverbänden funktionieren – in einem globalen Experimentierlabor für virtuelle Unternehmen von morgen.

Wer jetzt erstaunt den vermeintlich neuen Trend hinterfragt, dem sei gesagt, dass Open Source Software bereits auf eine 20-jährige Historie zurückblickt – für die schnelllebige Computerwelt eine beachtlich lange Zeit. Dabei wurden viele qualitativ hochwertige Ergebnisse erzielt, wie zum Beispiel Linux, Apache und Mozilla. Erreicht wurde es, obwohl die klassischen Rahmenbedingungen fehlen: Denn es existiert kein Budget. Es existiert kein Auftraggeber. Es gibt keine Zielvereinbarungen auf feste Termine. Projektstrukturen entstehen ad hoc und lösen sich gegebenenfalls wieder auf. Die Software-Ersteller erhalten kein Entgelt für ihren Beitrag.

Stattdessen beobachtet man: international verteilte Arbeitssysteme mit Selbstorganisation und informeller Planung. Selbstbestimmung bei den Zielen. Kostenfrei verfügbare Kollaborationsplattformen im Internet. Freiwillige, gegenseitige Hilfe. Software in Produktqualität zum kostenlosen Download. Offenbar gibt es explizite und implizite Faktoren, durch die



alle Beteiligten kooperativ zusammenwirken, sich permanent selbst organisieren und auf ein gemeinsames Ziel hin ausrichten.

Was aber sind die Gründe dafür? Auffallend ist zunächst, dass Open Source-Entwickler mit einem gewissen Stolz ihr Ergebnis allen zur Verfügung stellen. Dieses Qualitätsbewusstsein für das eigene Produkt rangiert ganz oben. Eine weitere entscheidende Größe, warum Open Source so gut funktioniert, scheint im Zusammenspiel von kollektiven Werten, Denkmustern und Verhaltensweisen zu liegen. Alle Beteiligten fühlen sich als Kokreative, man arbeitet gemeinsam für das größere Ganze. Diese Kultur steht übrigens auch für ein Paradoxon der Softwareproduktivität: Effizienz wird dadurch erreicht, indem man darauf verzichtet, Effizienz explizit einzufordern.

Ein bekanntes Beispiel im Internet ist *sourceforge.net*. Ein Blick auf den dortigen Server zeigt aktuell mehr als 79.000 Open Source-Projekte an. Natürlich sind darunter auch viele kleine Projekte oder solche, die über die Planungsphase nie hinauskommen. Die Fülle zeigt aber deutlich, dass sich Open Source Software nicht auf das populäre Linux beschränkt. Das Besondere an Sourceforge ist, dass nicht nur die aktuelle Software-Version zum Download bereitgestellt wird – vielmehr befindet sich die gesamte Entwicklungsumgebung für das verteilte Projektmanagement im Internet.

Somit liegt nicht nur die Software offen, sondern auch der gesamte Entwicklungsprozess inklusive aller Diskussionen. Alles fein säuberlich im Internet abgelegt und im Bedarfsfall recherchierbar. Durch dieses Vielaugen-Prinzip finden eine ständige Qualitätsüberwachung und eine schnelle Fehlerbeseitigung statt, was der Open Source Software Robustheit und Sicherheit verleiht. Doch warum funktioniert diese Zusammenarbeit? Während in gängigen Community-Modellen immer betont wird, wie wichtig persönliche Treffen für ein gegenseitiges Vertrauen sind, finden sie in der Open Source-Szene, wenn überhaupt, nur selten statt. Noch ein Paradoxon?

Eigentlich nicht. Ein Blick auf die wissenschaftlichen Erkenntnisse zur Evolution von Kooperation hilft weiter. Die Spieltheorie hat am Beispiel des Gefangenendilemmas gezeigt, dass Kooperation auch ohne Kenntnis des Mitspielers eine sinnvolle und stabile Strategie sein kann. Die erfolgreichste Strategie TIT FOR TAT beruht alleine auf Reziprozität mit einem ersten Good-Will-Schritt. Die einzige Voraussetzung ist, dass die Interaktion langfristig stattfindet und die Spieler immer wieder aufeinander treffen können. Reziprozität scheint also der Schlüsselfaktor der Kooperation zu sein, wenn man Mitspieler nicht kennt und überdies keine übergeordnete Instanz existiert, die Ausbeutung bestraft.

Ein Vertrauen über persönliche Treffen ist also eine nicht notwendige Voraussetzung. Bei einer Open Source Community gibt es ein generelles Vertrauen in die Prinzipien von Open Source Software, die auf der Grundlage von Transparenz aufbauen. Es bestätigt sich weiterhin, dass künstliche Anreize durch Incentive-Systeme in einer Gruppe von Gleichberechtigten überflüssig sind. Es ist überwiegend die intrinsische Motivation, die eine Kooperation zwischen untereinander fremden Personen, die weltweit verteilt sein können, herbeiführt.

Eine Studie des MIT über das Apache Usenet-Hilfesystem hat ergeben, dass eine Symmetrie im Geben und Nehmen eine entscheidende Größe für die Motivation der Befragten war, in Open Source Software-Projekten mitzuarbeiten. Diese Symmetrie muss nicht unmittelbar gegeben sein, sondern wird längerfristig im Sinne einer Tauschökonomie betrachtet. So bekommt derjenige, der Software schreibt, Feedback über die Funktionstüchtigkeit beziehungsweise korrigierte Codesegmente zur Fehlerbeseitigung. Wer in den Foren einen hilfreichen Beitrag findet, ist häufig motiviert, auch jemand anderem einen hilfreichen Tipp zu geben. Solange ein Nehmen und wieder

*Offen und weit verzweigt.*

*Geben und Nehmen.*

Zurückgeben beziehungsweise Geben und wieder Zurückbekommen stattfindet – ganz im Sinne der TIT FOR TAT-Strategie –, hat die Lebensfähigkeit des Projektes eine gute Prognose.

Kein Wunder, dass sich eine große Gruppe von Open Source-Interessierten unter Studenten findet. Für sie ist die Möglichkeit, Fähigkeiten und Fertigkeiten an einem interessanten Projekt weiterentwickeln zu können, ein starkes Motiv. Und offenbar ist es in Open Source Software-Projekten so, dass man beim Lernen auch Spaß haben kann. Weshalb Linus Torvalds sein Buch auch unter dem Titel *Just for fun* publiziert hat. Betrachtet man hingegen die quälenden Bemühungen, das Thema E-Learning voranzubringen, so hat sich im Open Source-Umfeld bereits eine virtuelle Lernplattform etabliert, ohne dass jemand sie gezielt als solche geplant hat.

Ein weiterer Antrieb, sich als Entwickler an Open Source-Projekten zu beteiligen, ist, als Autor bekannt zu werden. Da eine Software schneller größere Verbreitung findet, wenn sie kostenlos ist, wiegt für viele das Bekanntwerden den Verzicht auf ein direktes Entgelt auf. Obwohl man natürlich dagegenhalten könnte, dass die Reputation langfristig zu besseren Karrieremöglichkeiten und damit wieder in Geld umgesetzt werden kann. Diese Art von Berechnung scheint jedoch für viele nicht im Vordergrund zu stehen. Es geht meist gar nicht um die Anerkennung durch eine Firma, in der eine Kompensation in Form von Geld und Karriere angeboten wird, sondern um die Anerkennung unter den Experten für Software-Entwicklung. Die Experten erkennen die Kompetenzträger an ihrem Werk – eine Selbstdarstellung in Form von schönen Reden und bunten Folien ist überflüssig.

Ein anderes von Entwicklern häufig genanntes Motiv ist, in Open Source Software-Projekten das tun zu können, was man wirklich möchte. Wenn das Ergebnis dann bei anderen Beachtung findet, ist es Anerkennung dafür, was man gemacht hat, und nicht dafür, dass man es so gemacht hat, wie es jemand aufgetragen hat. Daraus kann man schließen, dass Eigenständigkeit und Selbstbestimmung der Kreativität sehr förderlich sind, während eine polarisierte Kunden-Lieferanten-Situation eher blockierend wirkt. Im Open Source-Umfeld ist der „Kunde“ nicht entmündigter Konsument, sondern kann unmittelbar Einfluss nehmen und aktiv an der Wertschöpfung teilhaben.

*Selbstbestimmung in der Community.*

Die genannten Motive sind die Bindeglieder in einer Open Source Community und bilden die Basis für deren Kooperationsverhalten. Es besteht ferner eine Abneigung gegen die Restriktionen bei der Nutzung kommerzieller Software durch Lizenzregelungen, die als mehr oder weniger kompliziert mit undurchsichtigen rechtlichen Konsequenzen angesehen werden. Was aber weniger als „Glaubensmodell“ oder als „Gegenoffensive“ zu Microsoft zu verstehen ist. Man ist ganz einfach davon überzeugt, dass sich „Software-Monokulturen“ als problematisch erwiesen haben und die damit verbundenen Abhängigkeiten mehr als hinderlich sind.

Der Quellcode soll daher offen zugänglich sein, weil diese Form der Transparenz Voraussetzung für Qualität und schnelleren technischen Fortschritt ist. Die Verfügbarkeit des Codes ermöglicht es, unmittelbar selbst Fehler zu beheben und zusätzlich benötigte Funktionalität einzubauen, ohne auf die nächste angekündigte Version warten zu müssen. Luftige Marketingversprechungen haben wenig Chancen in dieser Kultur, da die Karten offen auf den Tisch – in diesem Fall auf den Server – gelegt werden.

Mit Open Source Software ist der Beweis erbracht, dass das produktive Zusammenwirken geistiger Ressourcen ohne nennenswerten Geldmitteleinsatz möglich ist – sogar in großen Projekten. Allerdings gelingt dies in einem Kontext, der von dem eines Unternehmens stark abweicht. Für ein Unternehmen stellen sich aufgrund der beobachteten Phänomene viele Fragen:

*Freiwillige mit einem gemeinsamen Ziel.*

Ist Software gar kein Produkt, sondern eher eine Dienstleistung? Haben wir

in Unternehmen, organisationspsychologisch betrachtet, die falschen Strukturen für kreative Tätigkeiten? Ist eine mangelnde Kooperationsbereitschaft eine Folge der hierarchischen Aufbauorganisation, die zum Beispiel eine Konkurrenz um die Besetzung der nächsthöher bezahlten Stelle induziert? Kann ein klassisches Unternehmen sich überhaupt in eine virtuelle Organisation transformieren oder ist dafür ein Neuanfang notwendig?

Antworten finden wir bei einem der bekanntesten Managementtheoretiker. Peter Drucker schrieb 1998 in *Management's new paradigm*: „Es ist nicht Aufgabe der Manager, anderen zu sagen, was zu tun ist, sondern sie als Freiwillige zu sehen, die sich auf ein gemeinsames Ziel hin selbst verpflichtet haben und an der Steuerung ebenso beteiligt sein wollen wie am Erfolg.“ Drucker war es auch, der 1993 als Erster den Begriff der Wissensgesellschaft prägte und die steigende Bedeutung von Wissensarbeitern betonte. Offenbar ist die Kultur der Technologieexperten anders als die der Ertragsoptimierung. Die Skalierbarkeit des Profits durch Lizenzvertrieb ist im Modell der Open Source Software eingeschränkt oder ganz unterbunden. Das Dienstleistungsgeschäft durch Experten bleibt jedoch bestehen.

Ein weiteres Spannungsfeld entsteht dadurch, dass möglicherweise die Loyalität der Wissensarbeiter zu ihrer Profession größer ist als zum aktuellen Arbeitgeber. Wie kann also ein Unternehmen eine ausreichende Bindung seiner Wissensarbeiter erreichen, eine Frage, die sich für virtuelle Organisationen noch viel vehementer darstellt als ohnehin schon für die heutigen Firmen?

Zum Abschluss ein Tipp: Der erste umfassende Versuch in Deutschland, das Open Source-Phänomen fachübergreifend darzustellen, ist das *Open Source Jahrbuch 2004*. Das umfangreiche Werk mit 453 Seiten steht gemäß der Open Source-Philosophie als „open content“ im Internet unter <http://ig.cs.tu-berlin.de/Think-Ahead.ORG/> bereit.

*Manfred Langen* ist in der Siemens Corporate Technology für Wissensmanagement verantwortlich und hat mit seinem Fachzentrum die Siemens Intranet Portal Software unter [sipf.sourceforge.net](http://sipf.sourceforge.net) veröffentlicht.

**E-Mail-Adresse des Autors:**

[manfred.langen@siemens.com](mailto:manfred.langen@siemens.com)

Mit einer Illustration von *Limo Lechner*.