

Benno Luthiger

# Geschäfte machen und Spaß haben

## Spaßsuchende Programmierer und profitsuchende Firmen im Open-Source-Bereich

**Viele Software-Entwickler suchen den Spaß am Programmieren, wenn sie sich in einem Open-Source-Projekt engagieren. Zugleich drängen immer mehr Firmen mit eindeutigen Geschäftsinteressen in diesen Bereich. Das muss kein Widerspruch sein, wie der Autor anhand seiner Dissertation zur Motivation von Open-Source-Programmierern zeigt.**

Programmieren macht Spaß. Open-Source-Software ist so gesehen das nützliche Nebenprodukt einer Tätigkeit, die ausgeübt wird, weil sie lustvoll ist. Diese einfache Erklärung tönt auf den ersten Blick überzeugend, weist aber beim genaueren Hinsehen zwei gewichtige Schwachpunkte auf. Sie vermag erstens nicht die Qualität diverser Open-Source-Produkte zu erklären. Zweitens unterstellt sie irrationale Software-Entwickler.

Ein rationaler Software-Entwickler weiß, dass mit dem Verkauf von Software Geld verdient werden kann und er schätzt vermutlich eine Option, bei welcher „Spaß haben“ und „Geld verdienen“ gleichzeitig realisiert werden kann, höher ein als eine Option, bei der nur Spaß möglich ist. Open-Source-Software ist aber frei erhältlich, also kann mit dem Verkauf solcher Software kein Geld verdient werden. Den rationalen Software-Entwickler vorausgesetzt, wäre demnach zu erwarten, dass überhaupt keine Open-Source-Software entsteht.

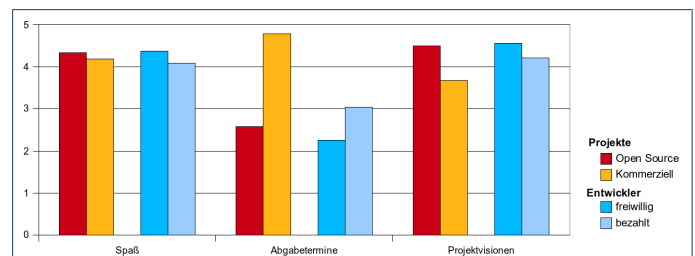
### Spaßsucher handeln rational

Damit Spaß die Erzeugung von Open-Source-Software auch unter der Annahme von rationalen Programmierern motivieren kann, braucht es ein zusätzliches Element. Stellen wir uns einen Software-Entwickler vor, der die Alternativen hat, in seiner Freizeit in einem Open-Source-Projekt zu arbeiten oder sein Pensum als Programmierer in der Software-Firma auszuweiten. Wenn der Ertrag an Spaß bei beiden Optionen gleich groß ist, wird dieser Programmierer zweifellos die zweite Option wählen, weil er dadurch den zusätzlichen Nutzen eines größeren Einkommens hat. Wenn allerdings der Spaß beim bezahlten Programmieren geringer ist, so wird ein Open-Source-Engagement dieses Software-Entwicklers verständlich. Das Schlüsselement, damit Spaß als Motivationsfaktor für Open-Source-Software funktionieren kann, ist also, dass die gleiche Tätigkeit (das Programmieren) in einem unterschiedlichen Kontext (Open Source/kommerzielle Software) unterschiedlich viel Spaß macht.

In der FASD-Studie (Fun and Software Development) [1] wurde der Spaß, den Software-Entwickler haben, sowohl in Open-Source-Projekten als auch unter kommerziellen Bedingungen untersucht. Die Umfrage wurde von 1.330 Open-Source-Programmierern (über die Open-Source-Plattformen SourceForge, GNU/Savannah und BerliOS) sowie 114 Programmierern in sechs Schweizer Software-Firmen ausgefüllt. Mit dieser Studie konnte die Hypothese bestätigt werden: Software-Entwickler, die in ihrer Freizeit in Open-Source-Projekten arbeiten, empfinden mehr Spaß am Programmieren als solche in kommerziellen Software-Projekten. Interessant ist, dass der gleiche Effekt auftritt, wenn freiwillige Open-Source-Entwickler mit solchen verglichen werden, die für ihre Open-Source-Arbeit bezahlt werden.

Damit stellt sich die Frage, welche Unterschiede in den Entwicklungsmodellen den Unterschied beim Spaß am Programmieren verursachen. Werden die Entwicklungsmodelle verglichen, so

ist das Programmieren in Software-Firmen durch monetäre Anreize (die Programmierer verdienen mit ihrer Arbeit), durch formale Hierarchien und durch Abgabetermine gekennzeichnet. Wenn sich Software-Entwickler in ihrer Freizeit in Open-Source-Projekten engagieren, so ist ihre Tätigkeit durch Freiwilligkeit und ihre Projektarbeit durch eine klare Projektvision charakterisiert. Die Freiwilligkeit des Open-Source-Engagements ist deshalb von Bedeutung, weil es dadurch den Open-Source-Programmierern möglich wird, mit einem „Trial-and-Error-Verfahren“ das Projekt herauszufinden, das ihnen die optimale Herausforderung bietet.



Im Rahmen der FASD-Studie wurden unterschiedliche Projektsituationen verglichen. Das Ergebnis: Open-Source-Entwickler haben mehr Spaß, Open-Source-Projekte haben deutlich weniger Abgabetermine und etwas häufiger eine Projektvision.

### Projektvision und optimale Herausforderung

Wie die weiteren Resultate der FASD-Studie zeigen, sind es nicht die Abgabetermine in kommerziellen Software-Projekten, die den Spaß am Programmieren beeinträchtigen, sondern die Projektvisionen und die optimale Herausforderung, die das Programmieren in Open-Source-Projekten so lustvoll machen. Dieses Resultat ist bemerkenswert, bedeutet es doch, dass Programmieren im kommerziellen Softwarebereich im Prinzip ebenso viel Spaß machen könnte wie in einem Open-Source-Projekt. Bezahlte Software-Entwickler müssen sich wohl oder übel formale Hierarchien und Abgabetermine gefallen lassen. Würden sich diese Faktoren stark negativ auf die Freude am Programmieren auswirken, so wäre Programmieren unter kommerziellen Bedingungen zwangsläufig immer weniger lustvoll als im Open-Source-Umfeld. Da sich aber diese beiden Faktoren anscheinend nicht negativ auswirken, sondern der spaßfördernde Effekt zustande kommt, weil Open-Source-Projekte bezüglich Projekt-Visionen und optimaler Herausforderungen besser abschneiden, gibt es keinen prinzipiellen Grund dafür, dass Programmieren weniger Spaß macht, wenn man dafür bezahlt wird.

Es ist nachvollziehbar, dass die Projektsituation in kommerziellen Software-Projekten den Programmierern weniger Möglichkeiten gibt, ihr Potenzial optimal in das Projekt einzubringen. Auch ist es begreiflich, wenn Projekt-Verantwortliche in Software-Firmen darauf verzichten, eine Projektvision so zu skizzieren, dass sie von den Programmierern nachvollzogen werden kann. Dies bes-

ser zu machen wäre aufwändig, aber nicht unmöglich. Aus diesem Grund, so lässt sich aus den Untersuchungen schließen, ist es nicht unmöglich, dass ein Software-Projekt im kommerziellen Umfeld genauso viel Spaß macht wie eines, welches unter Open-Source-Bedingungen stattfindet.

### Firmen rechnen anders

Die Zahlen belegen eindeutig, dass professionelles Engagement im Open-Source-Bereich kein Einzelfall ist, sondern dass mit Hilfe von Open-Source-Software überzeugende Geschäftsmodelle realisiert werden können. Hier soll nicht auf die Vielfalt der möglichen Geschäftsmodelle eingegangen werden, sondern auf die Auswirkungen, die das breite Eindringen von kommerziellen Verwertungsstrategien in den Open-Source-Bereich zur Folge haben können.

Software-Firmen können Programmierer beauftragen, in Open-Source-Projekten zu arbeiten, weil sie sicherstellen wollen, dass gewisse Funktionen, die für das Geschäftsmodell der Firma wichtig sind, in der notwendigen Zeit mit der benötigten Qualität zur Verfügung stehen. Zu diesem Zweck kann eine Firma einen Programmierer, der schon lange im Projekt arbeitet und mit diesem vertraut ist, temporär für einen Job anstellen. Die Firma kann aber auch einen eigenen Programmierer für die Arbeit im Open-Source-Projekt abstellen. In beiden Fällen wird eine solche Arbeit den Programmierern mit großer Wahrscheinlichkeit nicht so viel Spaß machen wie ein Engagement auf freiwilliger Basis. Da die Arbeit bezahlt ist, muss sich die Firma nicht ausführlich um eine Projektvision kümmern und das Element der Freiwilligkeit fällt ebenfalls weg. Diese Faktoren könnten dazu führen, dass mit dem Eindringen der Geschäftswelt in den Open-Source-Bereich ein Verlust an Spaß verbunden ist.

Es lassen sich jedoch auch andere Szenarien denken. Wenn eine Firma Personen für Arbeiten in einem Open-Source-Projekt bezahlt, die zwar wichtig sind, aber wenig Spaß machen und daher häufig vernachlässigt werden (z. B. Dokumentation, Testen der Applikation, Qualitätssicherung), könnte dies zur Folge haben, dass die übrigen Programmierer an einem qualitativ verbesserten Projekt arbeiten. Es ist wahrscheinlich befriedigender an einem Projekt zu arbeiten, das ein qualitativ hochwertiges Produkt herstellt, als an einem, bei dem die Arbeit zwar Spaß macht, das Endprodukt aber wenig taugt. Diese Überlegung ist grundlegend für die Annahme, dass das Engagement von Firmen durchaus positive Auswirkungen auf den Spaß im Open-Source-Projekt haben kann.

### Firmen fördern Software-Qualität

Auf der Ebene der Software-Qualität können Firmen, die sich mit einer Geschäftsidee in einem Open-Source-Projekt engagieren, Positives zum Projekt beitragen. Software-Qualität kann als Kombination von Code-Qualität und Benutzerfreundlichkeit (Usability) betrachtet werden. Die Code-Qualität umfasst beispielsweise die Eleganz und Einfachheit des Codes, dessen Les- und Wartbarkeit. In Bezug auf Code-Qualität wird Open-Source-Software in aller Regel ein gutes Zeugnis ausgestellt. Schlechter fällt der Befund dagegen aus, wenn die Benutzerfreundlichkeit (z. B. der Lernaufwand zum Beherrschen der Applikation, die Effizienz des Gebrauchs, Fehlerhäufigkeit und die subjektive Zufriedenheit beim Gebrauch) beurteilt wird.

Wie kommt das gute Abschneiden von Open-Source-Software bezüglich Code-Qualität zustande und warum bestehen Mängel in Bezug auf deren Benutzerfreundlichkeit? Um diese Fragen zu beantworten, muss man die Dynamik in einem Open-Source-Projekt verstehen. Neben Programmierern, die durch Spaß motiviert werden, sind ganz pragmatisch motivierte Software-Entwickler

anzutreffen. Andere wiederum engagieren sich aus Gründen der Reputation. Pragmatische Programmierer möchten ihre Beiträge möglichst rasch in der Applikation integriert sehen und kümmern sich nicht umfassend um die Code-Qualität. Spaßsucher erfreuen sich vermutlich an elegantem Code, der leicht zu pflegen und zu erweitern ist. Allerdings hängt die Qualität des Codes, der von spaßmotivierten Programmierern beigesteuert wird, weitgehend von ihren Fähigkeiten ab. Die Code-Qualität von Open-Source-Applikationen hängt demnach in erster Linie von Programmierern ab, die sich im Projekt engagieren, um eine Reputation als gute Programmierer aufzubauen. Die Reputation eines Programmierers hängt stark von der Reputation des Projekts ab, an dem er beteiligt ist. Erhält ein Open-Source-Produkt einen schlechten Ruf, zum Beispiel aufgrund mangelhafter Code-Qualität, so schlägt dies unweigerlich auf die Reputation der involvierten Programmierer zurück. Reputationsmotivierte Programmierer haben also ein großes Interesse, dafür zu sorgen, dass das Produkt bezüglich Code-Qualität stimmt.

Warum ist dieses Argument nicht gültig, wenn es um die Benutzerfreundlichkeit von Open-Source-Applikationen geht? Software-Entwickler interagieren in erster Linie mit anderen Programmierern. Wenn ein Programmierer in einem Open-Source-Projekt eine Rückmeldung bekommt, so ist der Absender normalerweise ebenfalls ein Programmierer, der einen Fehler oder einen Verbesserungsvorschlag meldet. Wenn ein einfacher Software-Benutzer mit einem Open-Source-Produkt nicht zurechtkommt, wird er dieses Produkt beiseite legen und in Zukunft vermutlich nicht mehr benutzen. Das Open-Source-Projekt hat somit kaum eine Chance, von diesen fehlgeschlagenen Versuchen zu erfahren und daraus zu lernen.

Genau das ist der Punkt, an dem ein Open-Source-Projekt vom Engagement einer Software-Firma profitieren kann. Firmen, zumindest wenn sie Erfahrung im Software-Geschäft mitbringen, haben den Kontakt zu den Endanwendern und kennen deren Bedürfnisse und Probleme im Umgang mit Software. Firmen handeln im ureigenen Interesse, wenn sie dafür sorgen, dass das Open-Source-Projekt, in dem sie sich engagieren, nicht nur bezüglich der Code-Qualität, sondern auch im Hinblick auf die Benutzerfreundlichkeit gut abschneidet. Open Source ist ein Zusammenspiel verschiedener Interessen und Motive, die sich im optimalen Fall ideal ergänzen. Spaß spielt eine wesentliche Rolle bei der Erzeugung von Open-Source-Software. Das Eindringen von Firmen mit klaren Geschäftsinteressen in den Open-Source-Bereich muss den bisherigen Erfolg der Open-Source-Bewegung nicht zwangsläufig gefährden. Es sollte vielmehr dazu genutzt werden, bestehende Mängel zu beheben, damit in Zukunft noch bessere Open-Source-Software entsteht und verfügbar ist.

### Links und Literatur

☞ [Softlink 1256](#)

- [1] Die Studie Fun and Software Development: <http://www.dissertationen.unizh.ch/2006/luthigerstoll/diss.pdf>

#### DER AUTOR



Benno Luthiger hat Physik und Ethnologie studiert. Seit 1990 verdient er seinen Lebensunterhalt als Software-Entwickler. 2006 hat er seine Dissertation "Spaß und Software-Entwicklung: Zur Motivation von Open-Source-Programmierern" abgeschlossen.