Peer-Produktion – der unerwartete Aufstieg einer commonsbasierten Produktionsweise

Christian Siefkes

Das Linux-Prinzip

Im Jahr 1991 hatte der junge finnische Informatikstudent Linus Torvalds eine verblüffende Idee: Er begann auf seinem PC ein Betriebssystem zu schreiben. Zunächst ging es ihm nur darum, Dinge ausprobieren zu können, die mit den damals erhältlichen Betriebssystemen nicht möglich waren. Nach Monaten der Bastelei bemerkte Torvalds, dass er ein auch für andere nützliches System entwickelt hatte. Er kündigte seine Arbeit im Internet an und bat um Rückmeldung, welche Eigenschaften sich die anderen von einem solchen System wünschten. Wochen später stellte er die Software ins Internet. Jeder konnte seinen Code nun herunterladen, verwenden und (bei entsprechenden Programmierkenntnissen) den eigenen Bedürfnissen anpassen.

Die Software stieß auf gewaltiges Interesse, denn die damals verbreiteten Betriebssysteme konnten entweder wenig, oder sie waren teuer. Zudem wurden sie in Firmen entwickelt, auf die die Nutzerinnen und Nutzer keinen Einfluss hatten. Dass jemand ein Betriebssystem per Internet entwickelte, dabei die Nutzer explizit um Mitarbeit bat und die Ergebnisse allen frei zur Verfügung stellte, war eine Sensation. Es dauerte nur zwei Jahre, bis über hundert Leute an dem Projekt mitwirkten, das in Anlehnung an den Gründer »Linux« getauft wurde. Zu dieser Zeit hatte das von Richard Stallman initiierte GNU-Projekt bereits viele freie Betriebssystemkomponenten entwickelt – durch die Kombination mit dem von Torvalds geschriebenen Systemkern entstand daraus ein praktisch nutzbares und komplett freies Betriebssystem.

Heute gehört GNU/Linux – neben Windows und Mac OS – zu den drei verbreitetsten Betriebssystemen. Es wird von Millionen von Menschen verwendet. Noch beliebter als bei privaten Anwendern ist es bei Firmen, die zuverlässige Server brauchen. Auch wo die Leistungsanforderungen besonders hoch sind, wird das System genutzt, so laufen über 90 Prozent der 500 schnellsten Supercomputer unter Linux.[1]

Der Erfolg von GNU/Linux basiert darauf, dass die Software selbst – wie alle Freie Software – ein Gemeingut ist, das alle frei verwenden, weiterentwickeln und

1 | Siehe unter: http://top500.org/stats/list/37/osfam (Zugriff am 17.07.2011).

an andere weitergeben dürfen. Die Freiheiten, die Software zum Gemeingut machen, wurden schon in den 1980er-Jahren von Richard Stallman beschrieben und exemplarisch in der GNU General Public License (GPL) umgesetzt. Sie ist die populärste Lizenz für Freie Software[2] und wird auch für Linux verwendet. Entscheidend für den Erfolg ist zudem die Community, die Gemeinschaft, die die Entwicklung des Betriebssystems koordiniert. Die offene, dezentrale und scheinbar chaotische Art, in der Torvalds und seine Mitstreiter zusammenarbeiten, ist als »Basar«-Modell in die Software-Geschichte eingegangen (Raymond 1999) – im Gegensatz zum hierarchischen, sorgfältig geplanten »Kathedralenstil«, der nicht nur den Bau mittelalterlicher Kathedralen, sondern auch einen Großteil der in Firmen entwickelten Software prägte.

Bei Freier Software gibt es keine strenge Trennung zwischen Nutzung und aktiver Beteiligung. Viele Beteiligte nutzen die Software nur, doch einige tragen gelegentlich oder sogar regelmäßig und intensiv zu ihrer Weiterentwicklung bei. Wer was tut, entscheiden die Beitragenden selbst. Es gibt keine Beteiligungsverpflichtung, aber auch wenig Hindernisse. Wer gegen die üblichen Praktiken oder Empfehlungen verstößt, muss damit rechnen, von den anderen »geflamed«, das heißt in rauen Worten auf den Verstoß hingewiesen zu werden. Etwas Schlimmeres passiert selten, härtere Sanktionen, wie der formelle Ausschluss aus dem Projekt, kommen kaum vor.

Gemeingüter, Beiträge und freie Kooperation

Die Erfolgsgeschichte von GNU/Linux zeigt die für die sogenannte »Peer-Produktion« wesentlichen Merkmale auf. Peer-Produktion basiert auf Gemeingütern: Ressourcen und Güter, die allen zustehen und gemäß selbstdefinierten Regeln gemeinsam oder anteilig genutzt werden.[3] Zu den Regeln, die sich die Freie-Software-Community gegeben hat, gehören insbesondere »vier Freiheiten«, die im Beitrag von Benjamin Mako Hill ausführlich beschrieben werden.[4]

Stallmans GPL, die beliebteste Freie-Software-Lizenz, fordert, dass diese Freiheiten auch für alle Weiterentwicklungen der Software erhalten bleiben. Wenn ich demnach GPL-Software ändere und veröffentliche, muss ich sie selbst unter GPL stellen. Dieses Prinzip wird »Copyleft« genannt, da es sich das normale Urheberrecht (Copyright) zunutze macht, um dessen Funktion »umzudrehen«: Statt den Autorinnen und Autoren, die aus Gemeingut schöpfen, exklusive Verwertungs- und Kontrollrechte zu geben, stellt es sicher, dass die Software für immer und in allen Fassungen Gemeingut bleibt.

Peer-Produktion baut auf Gemeingütern auf, und sie erzeugt neue Gemeingüter oder pflegt und verbessert die vorhandenen. Andere Ressourcen wie die

2 | Zu alternativen Lizenzen und deren Vereinbarkeit miteinander vgl. auch den Beitrag von Mike Linksvayer in diesem Buch (Anm. der Hg.).

3 | Michel Bauwens schreibt in diesem Buch ausführlicher zur Peer-to-Peer-Produktion (Anm. der Hg.).

4 | Vgl. auch GNU-Projekt (2002): Die Definition Freier Software, online unter: http://www.gnu.org/philosophy/free-sw.de.html (Zugriff am 17.07.2011).

verwendeten Computer sind bei Peer-Projekten typischerweise in privater Hand, doch sie werden zum Erreichen der Projektziele benutzt, nicht zum Erzielen eines finanziellen Gewinns.[5]

Im Gegensatz zur Produktion für den Markt findet Peer-Produktion in der Regel nicht für den Verkauf, sondern für den Gebrauch statt. Projekte haben ein gemeinsames Ziel, und alle Teilnehmenden tragen auf die eine oder andere Weise zu diesem Ziel bei, weil sie die Ziele des Projekts teilen, weil sie genießen, was sie tun, oder weil sie der Community etwas zurückgeben wollen. Bei marktwirtschaftlichen Aktivitäten wird dagegen getauscht, meist eine Ware gegen Geld.[6]

Anders als in klassischen Unternehmensstrukturen und in planwirtschaftlichen Systemen gibt es in Peer-Projekten keine Befehlsstrukturen. Das heißt keineswegs, dass die Projekte unstrukturiert wären (meist gibt es Maintainer oder Administratorinnen, die das Projekt auf Kurs halten und entscheiden, ob Beiträge integriert oder zurückgewiesen werden), aber niemand kann befehlen und niemand ist gezwungen zu gehorchen. Der unübersetzbare Begriff der »Peers« bezieht sich auf diese freiwillige Kooperation zwischen Gleichberechtigten, die sich niemandem unterordnen müssen. In einem offenen Prozess entwickeln die Projektbeteiligten die Regeln und Organisationsformen der Zusammenarbeit selbst.

Freie Kultur und Freies Design

Was bei GNU/Linux funktionierte, wird in ähnlicher Form mittlerweile in unzähligen anderen Projekten praktiziert. Das wohl bekannteste Beispiel ist die 2001 gegründete freie Enzyklopädie Wikipedia. Ihre deutsche Ausgabe umfasst mittlerweile mehr als 1,3 Millionen Artikel, die englische mehr als 3,8 Millionen. Linux und die Wikipedia stehen exemplarisch für zwei Communitys – die Freie-Software-Bewegung (auch »Open-Source-Bewegung« genannt) und die Freie-Kultur-Szene. Beide sind viel größer als ihre jeweiligen Flaggschiffe. Es gibt Hunderttausende Freie-Software-Programme und Millionen von Werken (Texte, Bilder, Musik, selbst Filme), die unter Creative-Commons-Lizenzen veröffentlicht werden.[7]

Freie-Design-Projekte (oft auch »Open Hardware« genannt) entwerfen gemeinsam materielle Produkte und stellen dabei Objektbeschreibungen, Konstruktionspläne und Materiallisten zur freien Verfügung. Im Bereich elektronischer Hardware ist zum Beispiel das italienische Arduino-Projekt sehr bekannt geworden. Es wird von vielen anderen Projekten genutzt und erweitert. Offenes Möbeldesign betreiben Ronen Kadushin und das SketchChair-Projekt. Im Open Architecture Network und im Projekt Architecture for Humanity entsteht Architektur, die sich

5 | Commons sind in der Regel Mischformen von Privateigentum und Gemeineigentum, von privater und gemeinsamer Nutzung; siehe dazu beispielsweise die Beiträge von Mayra Lafoz Bertussi und Liz Alden Wily in diesem Buch (Anm. der Hg.).

6 | Stefan Meretz befasst sich in diesem Buch ausführlicher mit der Differenz zwischen der Logik des Marktes und der Logik der Gemeingüter.

7 | Die Freiheitsgrade der verschiedenen Creative-Commons-Lizenzen sind abgestuft, nicht alle gewähren die »vier Freiheiten«; mehr dazu schreibt Mike Linksvayer in diesem Buch (Anm. der Hg.).

an den Bedürfnissen der Bewohnerinnen und Bewohner und nicht an den Profitinteressen von Baufirmen oder der Selbstdarstellung von Designern orientieren soll. OpenWear ist eine kollaborative Kleidungsplattform, die Menschen ermutigt und unterstützt, selber zu Produzenten zu werden. Ähnliches betreiben Pamoyo in Berlin und Threadless in Chicago. Das Freifunk-Projekt baut frei zugängliche Funknetze auf. Das Open Prosthetics Project entwickelt frei nutzbare Arm- und Beinprothesen. Es wurde von einem ehemaligen Soldaten gestartet, der im Krieg eine Hand verloren hatte und mit den kommerziell erhältlichen Prothesen nicht zufrieden war. Das Projekt hat auch den Anspruch, eine bessere medizinische Versorgung der Menschen zu erreichen, die sie sich normalerweise nicht leisten können (zum Beispiel in den Entwicklungsländern).

Wer produzieren will, braucht Produktionsmittel. Viel Aufmerksamkeit hat der 3D-Drucker RepRap erregt, der seine eigenen Bauteile selbst »ausdrucken« kann. Fab@Home ist ein weiterer 3D-Drucker, ebenso der MakerBot, um den herum sich eine große Community gebildet hat. Thingiverse ist die Austauschplattform für 3D-Designs. Der Bau von Produktionsmitteln für die »personal fabrication« ist das Ziel der Projekte Contraptor und CubeSpawn, die computerkontrollierte (= CNC) Maschinen zur Verarbeitung von Holz und Metallen entwerfen.[8]

Bei Freien Designs kann man wie bei Freier Software und anderen freien Projekten darauf bauen, dass die Offenheit zu großer Vielfalt führt, da jeder eigene Anpassungen oder Erweiterungen beitragen kann. Dies erhöht die Chancen, dass eine dem eigenen Geschmack oder den eigenen Bedürfnissen entsprechende Variante bereits von anderen entworfen und online gestellt wurde. So können auch Spezialbedürfnisse abgedeckt werden, die sich für kommerzielle Anbieter nicht »rechnen«.

Die Entstehung communitybasierter Infrastrukturen

Mit Designs und Bauplänen allein lassen sich jedoch noch keine Dinge produzieren – dafür bedarf es auch materieller Produktionsmittel und Ressourcen. Immerhin sind im Zuge der technischen Entwicklung viele Produktionsverfahren günstiger und zugänglicher geworden. Heute können Hobbyisten und Peer-Projekte mit günstig erworbenen oder selbstgebauten Maschinen produzieren. Vor wenigen Jahrzehnten wären dafür noch kapital- und personalintensive Fabriken nötig gewesen.

Natürlich kann und wird nicht jeder alle zur Produktion notwendigen Dinge im Keller haben. Wichtig sind vielmehr Projekte, in denen sich zum Beispiel Einwohnerinnen und Einwohner eines Dorfs oder Stadtteils zusammentun, um eine gemeinsame Infrastruktur aufzubauen. Beispiele dafür gibt es. So haben die Einwohner der südafrikanischen Stadt Scarborough mit der Scarborough Wireless User Group ein dezentrales »Mesh«-Netzwerk eingerichtet, das ihnen den Zugang zum Internet und zum Telefonnetz ermöglicht. Die benötigten WLAN-Router werden von einzelnen Bürgern gekauft und dem Netz zur Verfügung gestellt – es gibt niemanden, dem das ganze Netz oder ein Großteil davon gehören würde, und damit

8 | Dank an Stefan Meretz für die Zusammenstellung vieler der hier genannten Projekte.

niemanden, der es abschalten oder zensieren könnte. Die benötigte Software und ein Teil der Hardware werden als Freie Software und Open Hardware entwickelt.

Zugleich entstehen erste offene, communityorganisierte Produktionsstätten. Weltweit existieren bereits über 50 Fab Labs, ein halbes Dutzend davon in Deutschland. Fab Labs sind offene Werkstätten, die den Anspruch haben, »beinahe alles« produzieren zu können. So weit ist es noch nicht, doch allerhand nützliche Dinge wie Möbel, Kleidung, Platinen und Computerzubehör lassen sich dort bereits herstellen. Bislang arbeiten die Fab Labs oft mit proprietären Maschinen, deren Design nicht offengelegt ist, so dass sie nicht nachgebaut oder verbessert werden können. Doch es gibt Anstrengungen, diese Abhängigkeit zu überwinden und eine von Grund auf commonsbasierte produktive Infrastruktur aufzubauen: ein Netzwerk Freier Produktionsstätten, dessen Ausstattung zu 100 Prozent aus Open Hardware und Freier Software besteht und in den zusammenarbeitenden Werkstätten selbst reproduziert werden kann. Auf diese Weise würde der kapitalistische Markt Schritt für Schritt überflüssiger, weil die Menschen immer mehr Dinge, die sie brauchen, auf der Grundlage von Commons in gemeinsamer Peer-Produktion herstellen können.

Zwei Konzepte von Fülle

Ist eine solche Ausweitung der Peer-Produktion von der immateriellen auf die materielle Ebene nicht aufgrund der Endlichkeit der natürlichen Ressourcen und der begrenzten Aufnahmekapazität der Erde zum Scheitern verurteilt? Es wäre unmöglich, sieben Milliarden Autos herzustellen und zu fahren, da allein das benötigte Benzin die vorhandenen Ölvorräte in kurzer Zeit aufbrauchen und die freigesetzten CO_2-Emissionen die Erderwärmung katastrophal beschleunigen würden (Exner et al. 2008). Jedoch sollte es ohne weiteres möglich sein, Fahrräder für alle zu produzieren, ohne die Grenzen der verfügbaren Biokapazität zu sprengen. Auch andere nicht allzu ressourcenintensive Dinge sollten in ausreichender Zahl produziert werden können, um die Bedürfnisse zu befriedigen.

Digitale Peer-Produktion im Internet hat eine erstaunliche Fülle an Software und Inhalten hervorgebracht – einen Reichtum, von dem wir alle zehren. Eine vergleichbare Fülle für alle in der materiellen Welt erscheint unmöglich, wenn man bei Fülle an »Verschwendung« denkt.

Man kann sich unter »Fülle« aber auch etwas anderes vorstellen, nämlich: »Genau was ich brauche, wenn ich es brauche«.[9] Mit Dingen, die man schnell wegwirft, kann man nicht mehr Bedürfnisse befriedigen als mit Dingen, die man länger behält. Die commonsbasierte Peer-Produktion ermöglicht es, einer solchen bedürfnisorientierten Fülle für alle näherzukommen. Dort kommt es dann etwa nicht auf Autos an, sondern auf die Möglichkeit, andere Orte zu erreichen und mit anderen Menschen in Verbindung zu treten.

Da ohne natürliche Ressourcen keine materielle Produktion möglich ist, bleibt allerdings auch das Potential der Peer-Produktion stark begrenzt, sofern nicht auch

9 | Zum Thema Fülle siehe das Gespräch zwischen Brian Davey, Wolfgang Hoeschele, Roberto Verzola und Silke Helfrich in diesem Buch (Anm. der Hg.).

die Nutzung der Naturressourcen gemäß ihren Prinzipien geregelt wird. Ebenso wie die digitale Peer-Produktion Wissen und Software als Gemeingüter betrachtet, muss die materielle Peer-Produktion die verwendeten Ressourcen und Produktionsmittel als Commons pflegen und gestalten, sie also nachhaltig nutzen und in ihrem Zustand erhalten oder verbessern. Dabei gilt es, Modi zu finden, die sicherstellen, dass niemand zu kurz kommt und alle Bedürfnisse (produktive wie konsumtive) ernst genommen werden.

Das ist eine große Herausforderung, doch die Peer-Produktion hat mit ihren (völlig unerwarteten) Erfolgsgeschichten wie GNU/Linux und Wikipedia gezeigt, dass sie Großes leisten kann. Und Beispiele für die langfristige Nutzung von natürlichen Ressourcen und die erfolgreiche Gestaltung selbstgeschaffener Infrastrukturen finden sich in der langen Geschichte der Commons zur Genüge. Für die weitere Stärkung der commonsbasierten Peer-Produktion ist es daher wichtig, dass Commoners aus allen Bereichen – ob digital, ökologisch oder traditionell – weiter aufeinander zugehen. Sie können viel voneinander lernen.

Literatur

Exner, Andreas/Lauk, Christian/Kulterer, Konstantin (2008): Die Grenzen des Kapitalismus. Wie wir am Wachstum scheitern, Wien.

Raymond, Eric (1999): Die Kathedrale und der Basar, online unter: http://gnuwin.epfl.ch/articles/de/Kathedrale/ (Zugriff am 02.10.2011).

Siefkes, Christian (2010): Peer-Produktion: »Wie im Internet eine neue Produktionsweise entsteht«, in: *Widerspruch – Münchner Zeitschrift für Philosophie,* Nr. 52, S. 37-51.

Siefkes, Christian (2011): »Das gute Leben produzieren«, in: *Streifzüge,* Nr. 51, S. 18-23.

Christian Siefkes (Deutschland) ist Programmierer und lebt in Berlin. Sein zentrales Forschungsinteresse gilt dem emanzipatorischen Potential freier Software und anderen Formen der commonsbasierten Peer-Produktion. Er bloggt auf http://www.keimform.de. Zu seinen Veröffentlichungen gehören *From Exchange to Contributions* (2007) und *The Emergence of Benefit-driven Production* (2011).