debatte 4 *André Spiegel* Die Befreiung der Information

André Spiegel

Die Befreiung der Information

GNU, Linux und die Folgen

Matthes & Seitz Berlin

André Spiegel, geboren 1969, ist promovierter Informatiker und arbeitet als freier Programmierer, Berater, Autor und Dozent. Seit 1994 ist er in mehreren Projekten der GNU/Linux-Szene als Entwickler aktiv. Er ist Associate Member der Free Software Foundation (FSF) und berät heute mehrere große Unternehmen beim Einsatz von Freier Software. André Spiegel lebt in Berlin.

Online-Ausgabe: www.die-befreiung-der-information.de

Alle Rechte für die kommerzielle Verbreitung:
MSB Matthes & Seitz Berlin Verlagsgesellschaft m.b.H.
Göhrener Str. 7, 10437 Berlin

www.matthes-seitz-berlin.de

Umschlaggestaltung: neo design consulting, Bonn
Druck und Bindung: in Deutschland

ISBN 978-3-88221-879-4 ISBN 3-88221-879-7 **debatte 4**

Inhaltsverzeichnis

Bestandsaufnahme

Computer in jedem Haushalt, zusammengeschaltet über ein Netz, das jeden mit jedem verbindet, über Ländergrenzen und Kontinente hinweg.

Was vielen vor zehn Jahren noch als unrealistisch erschienen wäre, ist heute so sehr Teil des Alltags geworden, dass wir es kaum noch als Besonderheit wahrnehmen. Die Entwicklung hat sich in einer Art Selbstorganisation vollzogen, ohne dass sie von einer einzelnen Instanz geplant oder gesteuert worden wäre – nicht einmal von den großen Konzernen, die lange Zeit den Geschehnissen eher atemlos hinterherliefen, als dass sie entschieden hätten, wohin die Reise gehen soll.

Und so ist eine Situation entstanden, in der Teenager, die über das Netz Musikstücke austauschen, den Vertriebswegen der Musikindustrie weit überlegen sind.

Es ist eine Situation, in der eine Online-Enzyklopädie, an der jeder Internet-Benutzer mit einem einzigen Mausclick mitarbeiten kann, in weniger als vier Jahren der Encyclopædia Britannica ebenbürtig wurde.

Eine Situation, in der jeder E-Mail-Schreiber seine Briefe so verschlüsseln könnte, dass auch die mächtigsten Geheimdienste der Welt davor kapitulieren müssten.

Es ist eine Welt, in der neun von zehn Computerbenutzern die Software, die sie zum Betrieb ihrer Geräte brauchen, von einem einzigen Unternehmer beziehen, der dadurch zum reichsten Mann der Welt wurde.

Gleichzeitig programmiert eine Szene von Enthusiasten ein alternatives Betriebssystem, das dieselben Aufgaben besser erfüllen kann, und stellt es jedem zur freien Verfügung.

Das informationstechnische Erdbeben, das diese Entwicklungen auslösen, ist so gewaltig, dass ganze Industriezweige um ihren Umsatz, wenn nicht sogar um ihre Existenz zu fürchten beginnen – und zwar mit einigem Recht. Die Folge ist, dass diese Industrien, sobald sie den Ernst der Lage erkennen, alle Hebel in Bewegung setzen, um die Entwicklungen aufzuhalten oder in ihrem eigenen Interesse umzulenken. Da die Veränderungen aber sehr weitreichend sind und da sie allesamt eher dem einzelnen Individuum als den großen Konzernen zugute kommen, sind drakonische Maßnahmen erforderlich, um sie wieder zu kontrollieren. Viele der Gesetze, die heute auf Betreiben großer Unternehmen erlassen werden, sind kaum anders als mit Orwell'schen Begriffen zu beschreiben. Werden sie durchgesetzt, dann könnten sich die bereits sichtbaren, positiven Effekte der digitalen Revolution in ihr reines Gegenteil verkehren. Das hat eine völlig neue Form von Bürgerrechtsbewegungen auf den Plan gerufen, die sich für die »digitalen Rechte« der Individuen einsetzen – Rechte, von deren Existenz die Betroffenen oft kaum etwas ahnen und mit deren Hintergründen sie erst vertraut gemacht werden müssen.

Wenn in diesem Buch von einer »digitalen Revolution« die Rede ist, sind damit zwei klar umrissene Faktoren gemeint. Der erste ist die rasante Entwicklung der Digital- und Computertechnik seit etwa der Mitte des 20. Jahrhunderts. Die Idee, Informationen wie zum Beispiel Texte, Musikstücke

oder Bilder durch Folgen von »Nullen und Einsen«, also Spannungsmuster in elektronischen Schaltkreisen, darzustellen, erwies sich als so mächtig, dass diese Technik innerhalb weniger Jahrzehnte den Weg auf jeden Schreibtisch, in jeden Haushalt, in jedes Hi-Fi-Regal und in jede Handtasche oder Jackentasche fand. Die Entwicklung verlief exponentiell: Schon in den siebziger Jahren hatte einer der Gründer der Firma Intel, Gordon E. Moore, vorausgesagt, dass sich die Anzahl der Schaltelemente auf einem Chip regelmäßig verdoppeln würde. Das Zeitintervall dafür wurde später auf etwa 18-24 Monate geschätzt. In den vergangenen dreißig Jahren erfüllte sich diese Voraussage mit bemerkenswerter Genauigkeit – und zwar zum Teil deshalb, weil jeder Hersteller immer wieder befürchtete, die Konkurrenz würde den nächsten Verdopplungsschritt termingerecht schaffen, und sich darum zum »Gleichziehen« gezwungen sah. Mehrfach mussten dazu völlig neue Technologien entwickelt werden, die wenige Jahre vorher noch nicht bekannt waren. Das Mooresche Gesetz wurde also zu einer Art selbsterfüllender Prophezeiung. In der Folge sind die Geräte, die wir heute mit uns herumtragen, um Größenordnungen leistungsfähiger als diejenigen, die in den siebziger Jahren noch ganze Laboretagen füllten. Ein typischer Mikroprozessor ist heute etwa tausendmal schneller als in den achtziger Jahren, die typische Größe des Arbeitsspeichers hat sich ebenfalls etwa vertausendfacht, und die Kapazität einer Computerfestplatte ist sogar um den Faktor zehntausend bis einhunderttausend größer geworden.

Für sich alleine hätte diese Entwicklung der Computertechnik jedoch kaum zu einer derartigen Umwälzung der Informationsprozesse in der Gesellschaft geführt, wie wir

sie heute beobachten. Es bedurfte noch einer zweiten, mindestens ebenso wichtigen Entwicklung, nämlich derjenigen eines weltumspannenden Netzwerkes, das diese Geräte in sehr effizienter Weise miteinander verbindet. Das Internet, wie wir es heute kennen, ging hervor aus einem Datennetz, das seit den siebziger Jahren mehrere US-amerikanische Universitäten miteinander verband, dem sogenannten ARPANET. Seinen Namen erhielt dieses Netz von der *Advanced Research Project Agency (ARPA)*, einer Behörde des US-Verteidigungsministeriums, welche die entsprechenden Forschungsprojekte zum Teil finanziert hatte. Zweifelsohne spielten also auch militärische Interessen eine Rolle bei der Entwicklung. Die oft gehörte Erklärung, das Internet sei entwickelt worden, um ein Netzwerk aufzubauen, das gegebenenfalls auch einen Atomschlag überstehen könnte, ist hingegen nur ein Gerücht, wenn auch ein sehr hartnäckiges. Tatsächlich ging es vor allem darum, die teure und begrenzte Computerleistung möglichst effizient einzusetzen, indem man sie auch Forschern an anderen Universitäten zur Verfügung stellte.[1]

Die Idee solcher vergleichsweise unregulierter Datenschnellverbindungen bewies eine derartige Zugkraft, dass sich das Netz schnell erweiterte. Bereits 1973 entstanden die ersten transatlantischen Verbindungen zu Universitäten in Norwegen und Großbritannien. Anfang der achtziger Jahre wurde mit dem TCP/IP-Protokoll eine Art *lingua franca* der Computernetze eingeführt, so dass es fortan möglich war, beliebige Arten von schon bestehenden Netzwerken zusammenzuschalten und über die Grenzen dieser Netzwerke hinweg Nachrichten auszutauschen. Man bezeichnete diese netzwerkübergreifende Kommunikation

mit dem Begriff »inter-networking«, und für das entstandene »Netz der Netze« setzte sich bald die Bezeichnung »das Internet« durch.

Waren sie einmal vorhanden, konnte man über die Datenleitungen beliebige Anwendungen realisieren. In der Frühzeit des Internet war das vor allem die elektronische Post (E-Mail) sowie ein System dezentral betriebener Diskussionsforen, das sogenannte USENET. In ein allgemein zugängliches Medium verwandelte sich das Internet dann Anfang der neunziger Jahre, als Tim Berners-Lee am Europäischen Kernforschungszentrum in Genf (CERN) das *World Wide Web (WWW)* erfand. Durch diese Technik wurde es möglich, Informationen in Form optisch aufbereiteter und formatierter Seiten zur Verfügung zu stellen, diese Seiten durch *Hyperlinks* miteinander zu verbinden und mithilfe von *Suchmaschinen* in Sekundenbruchteilen aufzufinden. Das Internet wurde damit zu einem globalen Informationsraum, der ohne besondere Vorkenntnisse betreten werden konnte. Fast gleichzeitig mit dem Aufkommen des World Wide Web begannen sich darum sowohl kommerzielle Anbieter als auch Privatbürger für das Netz zu interessieren – zunächst gegen beträchtlichen Widerstand seitens der Universitäten, die ihre Forschungsnetze durch die freie Wirtschaft gefährdet sahen. Allerdings war bis Mitte der neunziger Jahre bereits eine sehr vielfältige Netzkultur entstanden, die weit über rein akademische Interessen hinausging. Das Internet war nur noch nominell eine reine »Forschungseinrichtung«. Gegen Ende der neunziger Jahre kam es dann zum explosionsartigen Anwachsen der Teilnehmerzahl, als das Internet den Weg in die Privathaushalte fand.

Bemerkenswert an diesem Phänomen ist seine wirtschaftliche Grundlage. Der Verfasser erinnert sich noch gut an einen Moment Anfang der neunziger Jahre, als ihm ein Universitäts-Kollege über die Schulter schaute. Es war gerade ein transatlantischer Datentransfer im Gange.

»Und wer bezahlt das?«, fragte er entgeistert.

»Niemand. Es ist einfach da.«

Die Antwort ist weniger naiv, als sie klingt. Zwar ist es richtig, dass transatlantische Kabel und die übrige Infrastruktur des Netzwerks viel Geld kosten. Diese Kosten werden aber, anders als zum Beispiel beim Telefonnetz, nicht auf die einzelne Verbindung, den einzelnen Datentransfer umgelegt. Finanziert wird das Netz vielmehr indirekt, und das aus den unterschiedlichsten Quellen: Zum Teil sind es Steuergelder, zum größten Teil aber eine in der Summe längst nicht mehr nachvollziehbare Verflechtung zahlloser Betreibergesellschaften und Diensteanbieter. Das Internet gleicht in dieser Hinsicht viel eher dem Straßennetz als dem Telefonnetz: Es ist eine Ressource, die von der Gesellschaft als ganzer bereitgestellt und finanziert wird – auch wenn sich die einzelnen Teile in den unterschiedlichsten privaten und kommerziellen Händen befinden. Da es sich um eine Meta-Struktur – ein Netz der Netze – handelt, besitzt niemand die ultimative Kontrolle darüber, und die Finanzierung ist weitgehend entkoppelt von der Funktion des Systems.

Leistungsfähige Informationsprozessoren in den Händen jedes einzelnen Bürgers, weltumspannende Datenleitungen, die jedem ununterbrochen zur Verfügung stehen – es ist

klar, dass sich der Stellenwert der Information und die Art und Weise, wie die Gesellschaft mit ihr umgeht, dadurch nachhaltig verändern werden. Tatsächlich haben sich viele dieser Veränderungen bereits ereignet, haben unhintergehbare Fakten geschaffen, die wiederum neue Prozesse auslösen. Dieses Buch dokumentiert die wichtigsten von ihnen.

Die ersten, die mit den neuen Technologien in Berührung kamen, waren die Programmierer. Es ist darum auch nicht verwunderlich, dass das Hinterfragen der informationstechnischen Spielregeln der Gesellschaft unter ihnen seinen Anfang nahm. Während zu Beginn der achtziger Jahre viele Unternehmen damit begannen, Software als ein Produkt zu betrachten, das man verkaufen konnte, war der Programmierer Richard Stallman davon überzeugt, dass Software der Gesellschaft dann am meisten nützen würde, wenn sie frei wäre, also jedem ungehindert zur Verfügung stünde. Er begann darum das sogenannte GNU-Projekt, dessen Ziel es war, jedem Computerbenutzer »Freie Software« zur Verfügung zu stellen, die alle seine Bedürfnisse erfüllte, ohne dass er sich deshalb in die Abhängigkeit irgendwelcher Unternehmen begeben müsste. Um die Freiheit der GNU-Software zu garantieren, stellte Stallman sie unter eine Lizenz, die das Urheberrecht in geschickter Weise ausnutzt, um andere Programmierer oder Unternehmen zu ermuntern – oder auch zu zwingen –, ihre Software ebenfalls frei verfügbar zu machen. Aus dem GNU-Projekt ging schließlich das Betriebssystem hervor, das heute unter dem Namen »Linux« bekannt ist. Die zugrundeliegenden Ideen haben jedoch die Software-Industrie noch weit darüber hinaus verändert; Schlagworte wie »Open Source«, die heute nicht nur Programmierern geläufig sind, stam-

men ebenfalls aus dem Umfeld von GNU und Linux. Das auf diese Einleitung folgende Kapitel beschreibt diese Entwicklungen im Detail.

Während Programmierer wie Richard Stallman die Grundlagen für eine freie Verfügbarkeit von Information schufen, kam es zu einer scheinbar gegenläufigen Entwicklung im Bereich der *Kryptographie*, also der Wissenschaft, die sich mit dem Verschlüsseln und Entschlüsseln von Informationen beschäftigt. Mehrere mathematische und technische Durchbrüche sorgten seit den siebziger Jahren dafür, dass derjenige, der Informationen vor unbefugtem Zugriff schützen will, dies heute mit einer nie dagewesenen Effektivität tun kann – gleichgültig, ob es sich dabei um einen Medienkonzern handelt, der die unerwünschte Verbreitung seiner Produkte verhindern will, oder um einen Bürgerrechtler, der an einem repressiven Regime vorbei mit der Außenwelt kommunizieren will. Kryptographische Verfahren sind darum ein wichtiger Schauplatz der Auseinandersetzungen um den Status der Information in der Gesellschaft, und das dritte Kapitel dieses Buches wird sich ausführlich mit ihnen beschäftigen.

Der für die Öffentlichkeit sichtbarste Kampf um die Hoheit über die Information spielt sich heute in der Welt der klassischen Medien ab, besonders in den Bereichen Musik, Film und dem geschriebenen Wort. Am augenfälligsten ist dabei die Krise, in der sich die Musikindustrie befindet. Die Kapazität der Netzwerke ist heute ausreichend, um Audio-Daten in akzeptabler Qualität vollkommen widerstandslos und ohne nennenswerte Kosten zu übertragen. Die Vertriebsstrukturen der Musikindustrie – das Herstellen von Tonträgern, ihre Lagerung und Aus-

lieferung zu den Händlern – werden damit überflüssig. Freie Internet-Tauschbörsen für Musikdateien sind darum zu einer signifikanten Erscheinung geworden, und die Industrie geht, um ihr Geschäftsmodell zu verteidigen, in großem Stil dagegen vor. Tausende von Tauschbörsen-Benutzern werden verklagt, gleichzeitig versucht man, dafür zu sorgen, dass die Öffentlichkeit mittelfristig keine universell einsetzbaren Computer mehr kaufen kann. Sie sollen ersetzt werden durch Geräte, die nur noch solche Operationen erlauben, mit denen die großen Konzerne einverstanden sind.

Die Filmbranche befindet sich in einer vergleichbaren Situation, unterschieden allerdings durch die Tatsache, dass die Netzwerke dem Transport von Video-Daten noch nicht wirklich gewachsen sind. Ein abendfüllender Spielfilm besteht aus etwa tausendmal mehr Daten als ein dreiminütiges Musikstück. Allerdings ist seine Produktion auch etwa um denselben Faktor teurer. Die Filmindustrie hat darum ein vielleicht noch größeres Interesse als die Musikindustrie, die freie Verbreitung der Daten möglichst zu verhindern. Anders als die Musikindustrie muss sie dabei auch keinerlei Rücksicht auf etablierte Formate wie zum Beispiel das der Audio-CD nehmen. Sie setzt darum in viel stärkerem Maße bereits heute kryptographische Verfahren ein, um ihre Produkte und damit ihr Geschäftsmodell zu schützen. Gleichzeitig macht sie dabei immer wieder die Erfahrung, dass jeder dieser Versuche fast augenblicklich von der verteilten, dezentralen Intelligenz tausender Programmierer ausgehebelt und überwunden wird. Auch hier ist darum zunehmend die Forderung zu hören, das Problem aus der Welt zu schaffen, indem man der Öffentlichkeit den univer-

sellen Computer wegnimmt und ihn durch Geräte ersetzt, die unter der Kontrolle der Industrie stehen.

Das geschriebene Wort und seine physische Erscheinungsform, das Buch, ist demgegenüber von der digitalen Revolution bislang erstaunlich unberührt geblieben. Die Gründe dafür dürften vor allem in der technischen Überlegenheit des Mediums Buch liegen, dessen Vorteilen die digitale Welt bislang nichts Vergleichbares entgegenzusetzen hat. Beantwortet werden muss jedoch die Frage, ob und wie das in Büchern gespeicherte Wissen in den zunehmend digitalen Informationsraum unserer Kultur integriert werden kann.

Die hier skizzierten Entwicklungen in der Welt der klassischen Medien sind Gegenstand des vierten, fünften und sechsten Kapitels, die sich mit der Musikindustrie, der Filmbranche und mit der Zukunft des Buches beschäftigen. Die digitale Revolution bringt jedoch Möglichkeiten mit sich, die über die reine Übermittlung von Information von wenigen an viele, wie sie charakteristisch für die klassischen Medien ist, hinaus gehen. Das Internet erlaubt die hochgradig dezentrale Kommunikation und damit Kooperation zwischen Individuen; es sind diese Mechanismen, die ein Projekt wie das GNU/Linux-Betriebssystem überhaupt möglich machen. In den letzten Jahren sind mehrere höchst erstaunliche Projekte entstanden, die nach denselben Prinzipien funktionieren; das bekannteste von ihnen ist die Internet-Enzyklopädie Wikipedia. Die Mechanismen solcher weltweit verteilten Kooperationsprojekte stehen im Zentrum des siebten Kapitels.

GNU und Linux – Grundlagen einer Revolution

Computer sind universelle Maschinen: Alles, was sie tun, wird durch *Software* bestimmt. Es sind letzten Endes Spannungsmuster im Arbeitsspeicher des Computers, endlose Folgen von »Eins« und »Null«, von »Spannung ist da« und »Spannung ist nicht da«, die von der zentralen Prozessoreinheit als Anweisungen interpretiert werden. Aber schon von »Anweisungen« oder »Interpretation« zu sprechen, ist eigentlich ein kognitiver Trick. Tatsächlich sorgen die Spannungspegel an bestimmten Stellen der Maschine lediglich dafür, dass wiederum an anderen Stellen Schalter umklappen und Strom fließt oder kein Strom fließt. Aus endlosen Wiederholungen dieses Prinzips ist alles andere aufgebaut – und zwar in Form so vieler übereinander liegender, aufeinander aufbauender Schichten, dass es heute wenig einzelne Menschen auf der Welt geben dürfte, die einen handelsüblichen PC durch alle diese Schichten hindurch verstehen. Man kann es auch mit den Worten eines IBM-Mitarbeiters ausdrücken, der bei der Einweihung einer neuen Großrechenanlage verkündete: »Also, zählen kann die Anlage zwar nur bis zwei – aber das kann sie unheimlich schnell.«

Damit ein Computer überhaupt etwas Sinnvolles tun kann, ist eine gewisse Grundsoftware erforderlich, die zum Beispiel dafür sorgt, dass der Prozessor Zugriff auf Tastatur und Maus bekommt, der Bildschirm zum Leben erwacht und eine Netzwerkverbindung aufgebaut werden kann.

Diese Grundsoftware, man nennt sie das *Betriebssystem*, wird beim Einschalten des Rechners vom großen, aber langsamen Speicher der Festplatte in den kleinen, aber sehr viel schnelleren Arbeitsspeicher kopiert.

Auf der Software-Struktur, die das Betriebssystem bereitstellt, können dann die eigentlichen Anwendungsprogramme wie zum Beispiel ein Web-Browser oder ein Chat-Programm ausgeführt werden. In der Praxis verschwimmt allerdings die Unterscheidung zwischen Betriebssystem und Anwendungsprogrammen zunehmend. Wer ein Betriebssystem wie Microsoft Windows kauft – und man kann heute, wie wir noch sehen werden, beim Kauf eines PC in der Regel gar nicht vermeiden, Microsoft Windows zu kaufen –, der bekommt eine Software-Grundausstattung, in der die gängigsten Anwendungen wie zum Beispiel ein Web-Browser oder ein Abspielprogramm für Musikdateien bereits enthalten sind. Dem Käufer erscheint alle diese Software wie ein integraler Bestandteil des Geräts; nur gewisse Komponenten wie zum Beispiel ein Textverarbeitungsprogramm muss er noch nachträglich hinzufügen. Dass es bei allen diesen Bausteinen – bis hinunter zum eigentlichen Betriebssystem – überhaupt Wahlmöglichkeiten geben könnte, ist eine Einsicht, für die sowohl die Kunden als auch die Kartellämter lange Jahre gebraucht haben.

Programmierer schreiben ihre Programme – also zum Beispiel Betriebssysteme oder Anwendungssoftware – in besonderen, künstlichen Sprachen, den sogenannten *Programmiersprachen*. Beispiele dafür sind C, BASIC, LISP, COBOL oder Java. Die Anweisungen der Programmier-

sprache können aber nicht direkt von einem Computer ausgeführt werden; sie müssen zuerst in Folgen von Nullen und Einsen, also letztlich in Spannungspegel im Speicher des Computers, übersetzt werden. Diese Übersetzung selber wird wiederum von einem Programm durchgeführt, dem sogenannten *Compiler*. (Oft ist der Compiler in derselben Programmiersprache geschrieben, die er dann später übersetzt, was zu einem interessanten Henne-/Ei-Problem führt.)

Man nennt den Programmtext, den ein Programmierer zur Lösung eines bestimmten Problems schreibt, auch *Quelltext* (engl. *source code*). Die Zielsprache der Übersetzung ist die *Maschinensprache* des verwendeten Computers; das Produkt der Übersetzung nennt man *ausführbares Programm* (engl. *object code, binary code* oder *executable code*).

In diesem Sachverhalt liegt das technische Grundproblem, mit dem die Auseinandersetzung über den Status der Information ihren Anfang nahm. Es besteht in den folgenden zwei Punkten:

1. Ausführbare Programme – also endlose Folgen von Nullen und Einsen – sind für Menschen unverständlich und nicht beherrschbar (aus genau diesem Grund programmiert man in höheren Programmiersprachen und nicht direkt in Nullen und Einsen).

2. Es gibt keine Möglichkeit, die Maschinensprache in die Programmiersprache *zurück* zu übersetzen. Der Grund dafür ist, dass bei der Übersetzung in die Maschinensprache viel Information verloren geht: zum Beispiel Kommentare des Programmierers, die er in den Pro-

grammtext eingefügt hat, oder die sinngebenden Namen von Unterprogrammen und Variablen.

Wer heute ein Betriebssystem kauft, oder eine Anwendung wie zum Beispiel ein Textverarbeitungsprogramm, der bekommt vom Hersteller eine ausführbare Version des Programms. Der Quelltext hingegen, den man braucht, um die Wirkungsweise des Programms zu verstehen oder es zu verändern, ist in der Regel ein gut gehütetes Firmengeheimnis. Und selbst wenn er das nicht ist – viele Hersteller betrachten den Quelltext zumindest als ihr »intellektuelles Eigentum« und behalten sich das alleinige Recht vor, ihn zu verändern und weiterzuentwickeln.

Manche Programmierer sind damit nicht einverstanden.

Eine Frage der Ethik

In den siebziger und achtziger Jahren war das Artificial Intelligence Lab des Massachussetts Institute of Technology (MIT) eines der wichtigsten Zentren der Computertechnik. Es war auch eine Art Oase der Freaks: Die Programmierer, die sich untereinander »Hacker« nannten, leisteten ihre Pionierarbeit auf den frühen Großrechnersystemen in einer unkomplizierten, kooperativen, fast familiären Atmosphäre. Zahllose Programme wurden geschrieben, verändert, weiterentwickelt, und die Tatsache, dass jedem der Quelltext aller Programme zur Verfügung stand, war unabdingbar und selbstverständlich. Bisweilen nahm der Teamgeist auch drastische Formen an: Kam zum Beispiel ein Professor auf die Idee, eines der wenigen Terminals für sich allein zu bean-

spruchen und in seinem Büro einzuschließen, dann trat die Kunst des »lock hacking« in Aktion – mithilfe möglichst kreativer Methoden öffnete man das Schloss des Büros und führte das Terminal wieder der Allgemeinheit zu.

Einer dieser Hacker war Richard M. Stallman. Vielleicht mehr noch als für seine Kollegen war die Beschäftigung mit Computern für ihn zum Lebensinhalt geworden. Im Labor gab es – für den Fall, dass jemand bis spät in die Nacht durchprogrammierte – ein Bett, und Stallman war nicht nur der häufigste Benutzer dieses Betts, er hatte zeitweise gar keine Wohnung mehr außerhalb des Labors.

In den achtziger Jahren begannen sich die Verhältnisse zu verändern, zum Teil dadurch, dass viele der Hacker mit hohen Summen von der Industrie abgeworben wurden, zum Teil dadurch, dass proprietäre Software von außerhalb des MIT ins Labor gelangte. Zum Beispiel stiftete die Firma Xerox dem Labor einen der ersten modernen Laserdrucker, lieferte die Treibersoftware aber nur als ausführbares Programm mit. Als der Drucker Schwierigkeiten machte, versuchte Stallman, an den Quelltext des Treibers zu kommen, um ihn so abzuändern, dass er richtig mit den Rechnern des Labors zusammenarbeitete. Aber Xerox gab den Code nicht heraus.

Ein Kollege von Stallman an der Carnegie Mellon University hatte den Code allerdings bekommen, jedoch als Bestandteil eines besonderen Vertrages mit Xerox und unter der Auflage eines *Non-Disclosure Agreements (NDA)* – er durfte den Code nicht an Dritte weitergeben. Stallman erfuhr davon und besuchte den Kollegen bei der nächsten Gelegenheit. Er betrat sein Büro und fragte nach einer Kopie des Codes.

Wenn man Stallman heute davon erzählen hört, gewinnt man den Eindruck, dass dies der Moment war, der sein Leben verändern sollte. Der Kollege antwortete: »Ich habe versprochen, dir den Code nicht zu geben.«

Stallman war völlig perplex. Ohne ein Wort zu sagen, verließ er wütend das Büro. Er konnte es, wie er heute sagt, noch nicht gleich formulieren, aber mit der Zeit wurde ihm klar, was ihn an der Situation so empört hatte: Es war die Tatsache, dass ihm der Zugriff auf eine offenbar vorhandene und nützliche Information verweigert wurde, und nicht nur das: Jemand hatte sogar *versprochen*, weder mit ihm noch mit irgendjemandem sonst zu kooperieren.

Was Stallman erlebt hatte, war eine Folge dessen gewesen, dass die Computertechnik sich von einem Gegenstand der akademischen Forschung zu einem Wirtschaftszweig entwickelte und die Industrie sich anschickte, ihre eigenen Spielregeln zu etablieren. Viele Leute gerieten zu dieser Zeit in sehr ähnliche Situationen wie Stallman, aber was ihn von diesen anderen unterschied, war, dass er sich nicht darüber beruhigen konnte. Er kam zu dem Schluss, dass ein Konzept wie das Non-Disclosure Agreement schlicht *unethisch* sei und dass er, um nicht gezwungen zu sein, unter solchen Bedingungen zu arbeiten, vielleicht sogar seinen Beruf aufgeben müsse. Dann aber kam ihm die Idee, wie er Programmierer bleiben und doch seinem moralischen Anspruch gerecht werden könnte: Er würde selber ein Betriebssystem schreiben und es jedem, der es benutzen wollte, frei zur Verfügung stellen, so dass niemand mehr von irgendeiner Software-Firma abhängig wäre.

Wahrscheinlich trägt diese Geschichte Züge der Stilisierung. Stallmans Biograph, Sam Williams, hat festgestellt,

dass weder Stallman noch sein damaliger Kollege an der Carnegie Mellon University sich an die Details des Gesprächs über die Druckersoftware erinnern können – einschließlich der Tatsache, ob das Gespräch überhaupt stattgefunden hat.[2]

Stallman begann jedenfalls, sein Betriebssystem zu programmieren, und zwar von Grund auf, einschließlich des Compilers und des Texteditors, die nötig waren, um den Rest des Systems überhaupt entwickeln zu können. Als Vorbild wählte er das bereits existierende Unix (»nicht mein ideales Betriebssystem, aber es ist nicht ganz schlecht«), und nannte sein eigenes System GNU, was für »GNU is Not Unix« steht – ein selbstreferentielles Akronym.[3]

Um zu verhindern, dass das MIT irgendwelche Rechte an seiner Software haben würde, ging Stallman schließlich im Jahr 1984 zu seinem Chef und erklärte seine Kündigung.

»Sind Sie da wirklich sicher?«

Stallman beharrte. Sein Chef hatte offenbar ein Gespür dafür, dass hier etwas Besonderes vorging, denn er nahm die Kündigung an und fragte dann weiter: »Wollen Sie Ihr Büro behalten?«

Verblüfft akzeptierte Stallman das Angebot. Das Labor am MIT wurde so für die nächsten Jahre zum Hauptquartier seiner Bewegung, obwohl er sowohl finanziell als auch rechtlich von der Universität unabhängig war. Als eine Art Dachorganisation gründete er die *Free Software Foundation (FSF)*, die später als gemeinnütziger Verein anerkannt wurde. Seinen Lebensunterhalt bestritt Stallman – in Dingen des täglichen Lebens ohnehin recht anspruchslos – durch den Verkauf gebundener Handbücher sowie

dadurch, dass er sich den Vertrieb seiner Software als Dienstleistung bezahlen ließ: »Schicken Sie mir 150 Dollar, und ich schicke Ihnen ein Magnetband mit meiner Software darauf.« In der Zeit vor der flächendeckenden Verbreitung des Internet ließ sich damit ein ausreichendes Einkommen erzielen, obwohl es jedem Benutzer natürlich freigestellt blieb, sich die Programme gegebenenfalls über andere Kanäle umsonst zu besorgen.

Vom Programmieren zum Asteroiden-Bergbau

Zu Beginn des GNU-Projekts schrieb Richard Stallman ein Manifest, um Mitstreiter zu gewinnen und für seine Idee zu werben.[4] Stallman erklärt darin, dass er es als seine moralische Pflicht betrachte, das GNU-System zu entwickeln, denn die »Goldene Regel« verlange, dass derjenige, dem ein Programm gefällt, es mit denen teilen müsse, denen es auch gefallen könnte. Es sei ihm daher unmöglich, ein Non-Disclosure Agreement oder eine Software-Lizenzvereinbarung zu unterschreiben. Um weiterhin Computer benutzen zu können, werde er einen Grundstock an freier Software zusammenstellen, der ihm erlaube, ohne jede Art von proprietärer Software auszukommen. Viele Programmierer seien bereit, ihm zu helfen, denn das Weitergeben von Programmen sei der fundamentale Akt der Freundschaft zwischen Programmierern.

Stallman fährt fort, indem er »einige leicht zu widerlegende Einwände gegen das GNU-Projekt« aufzählt. Der Kern dieser Einwände ist, dass dieses Modell unmöglich ökonomisch funktionieren könne, ja, dass die Program-

mierer, wenn sie ihre Programme verschenken, schlicht verhungern müssten. Stallmans Antwort auf diesen Einwand besteht aus drei Teilen.

Erstens, so führt er aus, sei natürlich niemand gezwungen, Programmierer zu sein: »Die meisten von uns sind nicht in der Lage, Geld zu verdienen, indem sie sich an irgendeine Straßenecke stellen und Grimassen schneiden. Das heißt aber nicht, dass wir darum dazu verdammt sind, uns an Straßenecken zu stellen, Grimassen zu schneiden und zu verhungern. Wir machen etwas anderes.«

Den Einwand, dass ohne finanziellen Anreiz aber niemand mehr programmieren würde, lässt Stallman nicht gelten: »Programmieren übt auf manche Menschen eine unwiderstehliche Faszination aus, üblicherweise auf die, die es am besten können. Es gibt [auch] keinen Mangel an professionellen Musikern, [...] obwohl die wenigsten von ihnen hoffen können, damit je ihren Lebensunterhalt zu verdienen.«

Zweitens aber stimme die implizite Annahme des Fragenden nicht, nämlich dass ein Programmierer niemals auch nur einen Cent verdienen könne, wenn er sich nicht das Recht bezahlen lässt, seine Software zu benutzen. Andere Modelle seien vorstellbar, wenn man nur nach ihnen suche: Programmierer könnten beispielsweise Benutzerberatung anbieten, oder ganz allgemein Dienstleistungen im Zusammenhang mit der Software (Unterricht, Installation, Wartung etc.). Die Entwicklung neuer Systeme könnte durch Interessengruppen finanziert werden, deren Mitglieder Beiträge bezahlen, die dann eigens engagierten Programmierern als Arbeitslohn zukommen. Auch eine Software-Steuer sei vorstellbar, die dann auf einzelne Programmierer bzw. Unternehmen umgelegt würde. Benutzer

könnten aber auch von sich aus beschließen, bestimmte Projekte zu unterstützen, etwa weil sie die Resultate nach ihrer Fertigstellung verwenden wollen. Diese Beträge ließen sich dann von der Steuer absetzen.

Letzten Endes jedoch, und damit beschließt Stallman sein Manifest, gehe es um eine viel größere Perspektive. Er schreibt: »Programme frei zu machen ist langfristig ein Schritt in Richtung einer Welt ohne Ressourcenknappheit, wo niemand besonders hart arbeiten müssen wird, nur um seinen Lebensunterhalt zu verdienen. Die Menschen werden die Freiheit haben, sich mit Dingen zu beschäftigen, die Spaß machen, zum Beispiel Programmieren, nachdem sie die nötigen zehn Stunden pro Woche mit unumgänglichen Arbeiten verbracht haben wie Gesetzgebung, Familienberatung, Reparatur von Robotern und Asteroiden-Bergbau. Es wird nicht nötig sein, dass man vom Programmieren leben kann.«

Freie Software und die GPL

Stallman führte für sein Betriebssystem den Begriff *Freie Software* ein (engl. *free software*).[5] Der Begriff war von Anfang an als eine politische Idee gedacht. Freie Software, so definierte Stallman, ist Software, die dem Benutzer bestimmte Freiheiten gibt. Im einzelnen sind das die folgenden (sie werden, wie unter Programmierern üblich, bei null beginnend durchnummeriert):

Freiheit Nr. 0 Die Freiheit, das Programm auszuführen, jederzeit, zu jedem Zweck.

Freiheit Nr. 1 Die Freiheit, das Programm zu verändern, um es den eigenen Bedürfnissen anzupassen (Zugriff auf den Quelltext ist eine Voraussetzung dafür).

Freiheit Nr. 2 Die Freiheit, das Programm weiterzugeben, »um seinem Nachbarn zu helfen«.

Freiheit Nr. 3 Die Freiheit, auch veränderte Versionen des Programms weiterzugeben.

Der Ausdruck »Freie Software« legte allerdings auch den Grund für ein sehr hartnäckiges Missverständnis, und das insbesondere auf Englisch: Gemeint ist nicht, dass die Software »umsonst« wäre, »frei« im Sinne von »gratis«, oder dass »Freie Software« nichts mit Geschäft, Broterwerb oder Ökonomie zu tun hätte. Zwar ist es in aller Regel tatsächlich so, dass man Freie Software kostenlos bekommen kann, aber schon Stallmans Beispiel des Vertriebs von Magnetbändern zeigt, dass eine Vielzahl von kommerziellen Geschäftsmodellen um die Freie Software herum vorstellbar ist. (Tatsächlich funktioniert heute ein nicht unerheblicher Teil der Software-Industrie nach ganz ähnlichen Prinzipien.)

Als Eselsbrücke, um die beiden Bedeutungen von »frei« auseinander zu halten, hat sich das englische »Think of *free speech*, not *free beer*« eingebürgert. Natürlich war es nur eine Frage der Zeit, bis ein paar freundliche Spaßvögel das zum Anlass nahmen, »Freies Bier« herzustellen, also ein Bier, das nicht etwa umsonst ist, sondern dessen Rezept frei verfügbar ist.[6]

Das Gegenteil von »Freier Software« ist damit nicht etwa »kommerzielle Software«, sondern vielmehr Software,

für die eine oder mehrere der oben genannten Freiheiten nicht gelten. Man spricht dann von »unfreier Software« oder »proprietärer Software«, d.h. Software, die einen »Eigentümer« hat, dessen alleiniger Kontrolle sie unterworfen ist.

Um den rechtlichen Status der Programme des GNU-Projekts zu schützen, entwickelte Stallman außerdem eine besondere Lizenz, die *General Public License (GPL)*.[7] Die GPL gibt dem Benutzer eines Programms die vier oben genannten Freiheiten, aber sie geht noch einen Schritt weiter: Sie verlangt außerdem, dass weiterentwickelte Versionen des Programms, oder davon abgeleitete, neue Programme, ebenfalls unter der GPL lizenziert werden müssen.

Durch diese Klausel wird die Lizenz von einer bloß passiven Erklärung von Freiheitsrechten zu einem politischen Instrument. Sie bewirkt, dass Unternehmen den unter GPL lizenzierten Code nicht in ihre proprietären Produkte einbauen können, sondern dass sie im Gegenteil dazu ermuntert (oder gezwungen) werden, ihren eigenen Code ebenfalls als Freie Software zu veröffentlichen. Das Ziel, das Stallman damit erklärtermaßen verfolgt, ist, das Konzept der proprietären, unfreien Software vollständig abzuschaffen.

Das Urheberrecht wird hier also gewissermaßen zur Waffe gegen sich selbst umgeschmiedet. Stallman spricht daher auch humorvoll von *Copyleft*, also einem umgekehrten Copyright. Es ist eine Art formalisierter Ausdruck eines Prinzips, das auch unter dem Namen »Share-And-Share-Alike« bekannt ist, also in etwa: »Ich teile mit dir, teile du auch mit mir.«

Kritiker argumentieren, dass Freie Software unter der GPL also gar nicht wirklich »frei« sei, weil immerhin eines ausdrücklich verboten ist – die Software nämlich »unfrei«

zu machen. Manche sehen hier einen sich selbst widersprechenden Radikalismus am Werk. Andere halten dagegen, dass es nur recht und billig sei, wenn jemand, der etwas der Allgemeinheit zur Verfügung stellt, nicht möchte, dass sein Beitrag in proprietären Produkten verschwindet. Als Folge dieser Auseinandersetzung sind auch andere, »permissivere« Lizenzmodelle entwickelt worden, die gerade auf den Copyleft-Aspekt verzichten. Am einfachsten liegt die Sache bei sogenannter *Public-Domain-Software*, worunter man Software versteht, für die keinerlei Urheberrecht geltend gemacht wird, also auch keine Lizenz erforderlich ist. Um rechtliche Probleme zu umgehen (in Deutschland beispielsweise kann ein Autor auf das Urheberrecht gar nicht verzichten), werden allerdings in der Regel ausdrückliche Lizenzen verwendet, die jede Art von Verwendung der Software erlauben, aber zum Beispiel die Haftung des Autors für Fehlfunktion ausschließen (ob solch eine Klausel rechtlich wirksam ist, steht wiederum auf einem anderen Blatt). Es sind insbesondere große Konzerne, die solche Software inzwischen gerne als Technologie-Pool verwenden, aus dem sie sich nach Belieben für ihre eigenen Produkte bedienen.

Nichtsdestoweniger ist heute die GPL die in Freien Software Projekten mit Abstand am meisten verwendete Lizenz.

Just for fun

Im Lauf der achtziger Jahre stellten Richard Stallman und einige andere Programmierer einen Großteil des GNU-Systems fertig. Die Programme, die dabei entstanden, brachten es in der Fachwelt zu hohem Ansehen: Sie waren

technisch besser als ihre Gegenstücke aus den proprietären Unix-Varianten etwa von AT&T, Sun Microsystems oder Hewlett-Packard. Was noch fehlte, war der sogenannte *Kern* des Betriebssystems (engl. *kernel*). Der Kern ist die zentrale Schaltstelle in einem Betriebssystem, ein eigenes, sehr komplexes Programm, das die übrige Software, die auf dem Computer läuft, verwaltet und ausführt. Ohne einen eigenen Kern war der bereits funktionierende Teil des GNU-Systems gewissermaßen noch ein Luftschloss und konnte nur auf einem der schon existierenden, proprietären Unix-Systeme ausgeführt werden.

Das änderte sich, als der finnische Student Linus Torvalds im Jahr 1991 einen eigenen Kern schrieb. Torvalds, damals Anfang zwanzig, brauchte diesen Kern für Experimente mit Betriebssystemen und entwickelte ihn, wie er heute sagt, *just for fun*.[8] Um ein vollständiges Betriebssystem zu bekommen, bediente er sich bei den frei verfügbaren Programmen des GNU-Projekts, hatte aber selber keinerlei Beziehung zu GNU oder zur FSF. In Anlehnung an seinen Vornamen gab er dem Kern den Namen *Linux* und veröffentlichte ihn unter der General Public License (GPL), um ihn der Allgemeinheit zur Verfügung zu stellen. (Die Verwendung der GPL war eine eher beiläufige Entscheidung, Torvalds verfolgte mit seiner Arbeit keine politische Agenda.)

Damit war die kritische Masse erreicht. Zum ersten Mal war es möglich, einen Computer vollständig mit Freier Software zu betreiben. Zudem hatte Torvalds seinen Kern für die Intel x86-er Architektur geschrieben, d.h. für gewöhnliche, handelsübliche PCs. Für diese Architektur hatte es bis dahin nur unbedeutende, proprietäre Unix-

Varianten gegeben, nun aber existierte eine Alternative zum ansonsten marktbeherrschenden Betriebssystem Windows von Microsoft – wenngleich zunächst, und noch für etliche Jahre, nur technisch versierte Programmierer etwas mit dem neuen Betriebssystem anfangen konnten.

Linux erfreute sich dennoch steigender Beliebtheit unter Eingeweihten, und schließlich wurde das GNU-Projekt darauf aufmerksam. Richard Stallman zeigte sich irritiert. Jemand anders schien die Initiative übernommen zu haben und hatte das GNU-System zu einem vollständigen Betriebssystem gemacht. Was Stallman daran besonders kränkte, war, dass sich auch der griffige Name »Linux« für das System allgemein durchgesetzt hatte, wobei völlig unter den Tisch fiel, dass dieses neue Betriebssystem zum größten Teil aus den Programmen des GNU-Projekts bestand. Der Linux-Kern machte darin nur einen vergleichsweise kleinen, wenn auch entscheidenden Bestandteil aus. Ganz abgesehen von der fehlenden Anerkennung für die technische Leistung des GNU-Projekts, sah Stallman vor allem die Gefahr, dass die politischen Ideen, um deren Willen er das Projekt begonnen hatte, in Vergessenheit geraten könnten. Er forderte darum öffentlich, dass das System umbenannt werden müsse. Sein Vorschlag, *Lignux*, wurde von der inzwischen recht großen Fangemeinde jedoch nur mit Gelächter beantwortet. Linus Torvalds erklärte, dass ihm der Name des Systems gar nicht so wichtig sei – und letztlich blieb alles beim Alten. Stallman besteht seinerseits darauf, das System *GNU/Linux* zu nennen, aber diese Sprachregelung hat sich in der breiten Öffentlichkeit, die das Betriebssystem inzwischen genießt, nicht durchsetzen können.

Die Bestandteile eines GNU/Linux-Systems stammen inzwischen aus sehr vielen verschiedenen Projekten. Viele davon stehen unter der General Public License der Free Software Foundation, viele aber auch unter anderen, weniger politisch gefärbten Lizenzmodellen. Die eigentliche Software des GNU-Projekts stellt darunter nicht mehr den mengenmäßig größten Anteil dar, ist aber nach wie vor der größte Einzelbeitrag zum Gesamtsystem. Das GNU-Projekt hat inzwischen auch einen eigenen Betriebssystem-Kern unter dem Namen *Hurd* veröffentlicht, der aber kaum praktische Bedeutung hat.

In der Öffentlichkeit gilt Linus Torvalds als der Schöpfer von »Linux«, obwohl er selbst durchaus betont, dass er nur einen kleinen Teil beigesteuert hat und dass er auch in seinem eigenen Teilsystem, dem Linux-Kern, vor allem die Arbeit vieler anderer Programmierer koordiniert hat.

Public Relations

Eine weitere prominente Figur in der Szene ist Eric S. Raymond. Bereits seit den achtziger Jahren hatte er bei Projekten im Umfeld des GNU-Systems mitgearbeitet und sich einen Namen als fähiger Programmierer gemacht. Allerdings war er zunehmend uneins mit Richard Stallmans radikalen politischen Ansichten. Wie viele andere Programmierer betrachtete er das Schreiben Freier Programme nicht als eine moralische Verpflichtung, sondern war vielmehr fasziniert von der Möglichkeit, über das Internet mit hunderten von Kollegen zusammenzuarbeiten. Er verfolgte den Siegeszug des Linux-Kerns und fand,

dass er vor allem dadurch zu erklären war, dass Torvalds die neuen Kommunikationsmöglichkeiten des Internet viel stärker ausgenutzt hatte als Stallman im GNU-Projekt. Während die GNU-Programmierer vergleichsweise hinter verschlossenen Türen an ihrem Code arbeiteten, um nach zum Teil erheblicher Wartezeit einer staunenden Außenwelt ihre Ergebnisse zu präsentieren, hatte Torvalds seinen Linux-Kern fast sofort veröffentlicht. Mit seinem umgänglichen Charakter und kommunikativen Talent hatte er es geschafft, schnell eine große Menge von Partnern im Netz zu finden, und arbeitete mit diesen Leuten intensiv zusammen. Oft wurden an einem einzigen Tag mehrere neue Versionen des Linux-Kerns veröffentlicht.

Raymond fand, dass dieses Modell der hochgradig dezentralen Kommunikation zu besseren technischen Ergebnissen führte als das zentralisierte Entwicklungsmodell, wie es etwa die großen Software-Konzerne betreiben – ganz davon abgesehen, dass es einfach mehr Spaß machte. Da Raymond die Konzerne nicht als moralische Gegner betrachtete, versuchte er sich vorzustellen, wie man das freie Entwicklungsmodell in die Geschäftswelt tragen könnte, um damit Geld, möglicherweise sogar *viel* Geld zu verdienen.

In der zweiten Hälfte der neunziger Jahre nahm Raymond gewissermaßen die Rolle des Public-Relations-Experten der Bewegung an und versuchte, die großen Konzerne dafür zu interessieren. Die Firma Netscape war die erste, die darauf ansprang. Durch den »Browser-Krieg« gegen den Giganten Microsoft und dessen Internet Explorer zermürbt, war die Geschäftsleitung offen für Raymonds Ideen und entschloss sich im Jahr 1998, den Quelltext

ihres Browsers offen zu legen. Man hoffte, dass Massen von Programmierern sich aus den Tiefen des Internet darauf stürzen und das Produkt in ungeahnte technische Höhen katapultieren würden.

Die Signalwirkung dieser Entscheidung innerhalb der Computer-Industrie war nicht zu unterschätzen. Es zeichnete sich ab, dass Freie Software durchaus ein ernstzunehmendes Geschäftsmodell sein konnte. Raymond sprach dramatisch von einem »Schuss, der auf der ganzen Welt zu hören war«.

Auf den technischen Ertrag des Projektes musste man freilich eine ganze Weile warten. Die unabhängigen Programmierer ließen keinen Stein auf dem anderen und schrieben den Browser praktisch von Grund auf neu. Sie verwendeten dabei den bisherigen, Netscape-internen Codenamen des Projekts, *Mozilla*. Erst im Sommer 2002 erschien die offizielle Version 1.0 dieses Browsers, der wiederum von Netscape unter dem Namen *Netscape 7.0* vermarktet wurde. Mozilla unterschied sich von Microsofts Internet Explorer vor allem dadurch, dass der Browser die Standards des World Wide Web Consortiums (W3C) in vorbildlicher Weise einhielt, wogegen der Internet Explorer dafür bekannt war und ist, diese Standards regelmäßig zu ignorieren und so einen Microsoft-spezifischen Defacto Standard zu schaffen.

Mozilla war jedoch im Lauf der Entwicklung zu einem recht großen und schwerfälligen Programm geworden. Erst als man aus dem Mozilla-System einen neuen, bewusst klein und schlank gehaltenen Browser herauslöste, begann dessen Siegeszug. Der neue Browser erhielt den Namen *Firefox* und wurde mit großem Enthusiasmus von Frei-

willigen aus der Szene verbreitet. Das Echo davon drang bis in die Mainstream-Medien, so dass der Name Firefox heute den meisten Computerbenutzern ein Begriff ist. Der Marktanteil von Firefox, gemessen an der Zahl der Besucher auf bestimmten Websites*, liegt heute (Januar 2006) bei etwa 15-25%. Die ungebrochene Dominanz des Internet Explorer (70-80%) erklärt sich vor allem dadurch, dass dieser Browser von vornherein zum Windows-Betriebssystem gehört, während Firefox vom Benutzer selbstständig heruntergeladen und installiert werden muss.[9]

Open Source

Die Vorgänge im Umfeld der Netscape-Entscheidung von 1998 hatten dazu geführt, dass sich eine Gruppe von prominenten Vertretern der Szene formierte, die für eine verstärkte Zusammenarbeit mit der Industrie eintraten. Zu ihnen gehörten neben Eric Raymond auch der Verleger Tim O'Reilly und der Programmierer Bruce Perens. Stallman hingegen wurde von der Gruppe mehr oder weniger bewusst gemieden. Man hatte den Eindruck, dass sein moralistischer Anspruch und die Betonung der Idee der Freiheit bei Verhandlungen mit Geschäftsleuten nicht gut ankamen. Es schien an der Zeit, der Bewegung ein anderes Gesicht zu geben. Bei einer Art Gipfeltreffen, zu dem Tim

* Ein Web*site* ist ein bestimmter »Ort« im World Wide Web, von engl. *site* = der Ort, der Platz. Oft wird das mit der Web*seite* verwechselt, engl. *web page*, also einer einzelnen, im Web-Browser dargestellten Seite. Ein Web*site* besteht in der Regel aus mehreren Web*seiten*. In diesem Buch wird es darum immer *der* Website, aber *die* Webseite heißen.

O'Reilly eingeladen hatte, kam man überein, den Begriff »Freie Software« durch das weniger verfängliche »Open Source« zu ersetzen und rief die »Open Source Bewegung« ins Leben.[10]

Der unmittelbare Erfolg schien die Idee zu bestätigen. Schnell setzte sich der Begriff »Open Source« in weiten Teilen der Programmierer-Szene und in der Öffentlichkeit durch. In der Industrie ist »Open Source« heute ein fest etablierter Begriff, während der Ausdruck »Freie Software« in vielen Fällen auf Unverständnis stößt.

Es überrascht nicht, dass Richard Stallman mit diesem Namenswechsel nicht einverstanden ist. Er betont, dass der Begriff »Open Source« eine völlig andere Agenda hat als die Freie Software Bewegung, die er begründete. Der Fokus der Open Source Bewegung liegt darauf, technisch möglichst gute Software herzustellen, und sie argumentiert, dass der dezentrale, offene Entwicklungsprozess im Internet der beste Weg dazu ist. Die Lizenzmodelle, so heißt es, müssten jedoch den ökonomischen Realitäten Rechnung tragen. So findet Raymond beispielsweise nichts dabei, selber auch proprietäre Software zu schreiben oder mit proprietären Projekten oder Produkten zu kooperieren. Auf lange Sicht werde sich das Open Source Modell ohnehin evolutionär durchsetzen.

Stallman erklärt dagegen, dass der Gedanke der Freiheit oberste Priorität hat: Die Abhängigkeit der Benutzer von den Software-Herstellern soll gebrochen werden; wegen eben dieser Abhängigkeit ist es moralisch verwerflich, Quelltexte geheim zu halten, anstatt sie der Menschheit zur Verfügung zu stellen. Die eigene, Freie Software muss daher auch keineswegs technisch besser sein als die proprietären

Produkte – wenn sie das ist, dann ist das ein schöner Nebeneffekt, aber die Hauptsache ist, dass es Freie Software ist.

Stallman betrachtet die Open Source Bewegung daher als eine separate Bewegung, mit der er sich nicht identifiziert. Er betont ferner, dass es sich hierbei nicht um eine »klassische« Spaltung in einen realpolitischen und einen fundamentalistischen Flügel handelt. Er schreibt:

»Radikale Gruppen der sechziger Jahre hatten den Ruf der Spalterei: Organisationen brachen auseinander wegen Meinungsverschiedenheiten in strategischen Details und hassten einander dann. Sie stimmten in den grundlegenden Prinzipien überein und widersprachen sich nur in den praktischen Empfehlungen, aber sie betrachteten sich als Feinde und bekämpften sich bis aufs Messer. Das ist zumindest das Bild, das man heute von ihnen hat, ob es nun korrekt ist oder nicht.

Das Verhältnis zwischen der Freien Software Bewegung und der Open Source Bewegung ist gerade das Gegenteil davon. Wir sind in den grundlegenden Prinzipien uneins, aber haben mehr oder weniger dieselben praktischen Empfehlungen. Also können wir in vielen konkreten Projekten zusammenarbeiten, und tun das auch. Wir betrachten die Open Source Bewegung nicht als Feind. Der Feind ist die proprietäre Software.«[11]

Die Kathedrale und der Basar

Abgesehen von seinen Verbindungen zur Industrie ist Eric Raymond auch bekannt geworden durch seine Essays. Die bekanntesten von ihnen sind *The Cathedral and the Bazaar*

und *Homesteading the Noosphere* (etwa: »Die Besiedelung des Reichs der Ideen«).[12] Raymond versucht darin, historisch und konzeptionell aufzuarbeiten, was er die »Hacker-Kultur« nennt, also die Szene der über das Internet kooperierenden Programmierer, aus denen unter anderem die GNU/Linux-Bewegung hervorgegangen ist. Wie funktioniert diese Subkultur? Was motiviert ihre Anhänger? Warum ist sie so erfolgreich?

Der Ausdruck »Hacker« weckt in der Öffentlichkeit unbehagliche Assoziationen. Man stellt sich darunter hochbegabte Computerfreaks vor, die in Großrechnersysteme von Banken, Versicherungen und der NASA einbrechen und Chaos verbreiten. Die Hacker-Kultur, die Raymond beschreibt, distanziert sich von diesen Dingen nachdrücklich und möchte für solche Einbrecher lieber das Wort »Cracker« verwendet wissen. Der Ausdruck »Hacker« ist demgegenüber ein Ehrentitel. Er steht für jemanden, der vom Programmieren fasziniert ist und dem es Spaß macht, gut darin zu sein. Anders als die oft aus der Illegalität operierenden, destruktiven »Cracker« arbeiten die »Hacker« konstruktiv und sind stolz darauf, neue, bessere, auch benutzerfreundlichere Systeme zu bauen.

In *Homesteading the Noosphere* stellt Raymond die These auf, dass die Hacker-Kultur, obwohl scheinbar völlig offen und unorganisiert bis zur Anarchie, in Wirklichkeit einem strengen, unausgesprochenen Verhaltenskodex folgt.[13] So gibt es in fast jedem Projekt einen eindeutig festgelegten Projektleiter oder Moderator, der die letzte Entscheidungsgewalt darüber hat, welche Änderungen in das Projekt übernommen werden, welche neuen Funktionen hinzugefügt werden und welche nicht. Er ist es, der

das Projekt nach außen repräsentiert und der in der Regel auch entscheidet, wann eine neue Version der Software veröffentlicht wird. Der Führungsstil des Projektleiters ist eine Frage seiner Persönlichkeit, er kann von autoritär bis integrativ reichen und muss sich dadurch bewähren, dass es ihm gelingt, ein Team von Freiwilligen um sich zu versammeln und unter ihnen als Autorität akzeptiert zu werden. Oft ist der Projektleiter der ursprüngliche Autor der Software, der mit der Zeit eine Gruppe von Interessierten um das Projekt herum aufgebaut hat. Die Position des Projektleiters kann jedoch auch wechseln; in der Regel ist das ein Vorgang, der in wechselseitigem Einvernehmen geschehen muss und der sorgsam dokumentiert und bekannt gegeben wird, um die Autorität des neuen Projektleiters zu etablieren. »Wenn du das Interesse an einem Projekt verlierst, ist deine letzte Pflicht, es einem kompetenten Nachfolger zu übergeben«, schreibt Raymond.

Programmierer, die von außen an das Projekt herantreten, können in der Regel nicht einfach so »mitmachen«. Zu Beginn reichen sie meist einzelne, kleinere Änderungsvorschläge ein, die dann von einem Programmierer aus dem engeren Kreis geprüft und gegebenenfalls in den eigentlichen Programmcode eingefügt werden. Arbeitet ein Außenstehender längere Zeit mit und erwirbt er das Vertrauen der anderen Projektmitglieder, dann wird ihm schließlich der Schreibzugriff auf die zentrale Version des Programmcodes freigeschaltet, so dass er selbstständig Änderungen einfügen kann. Der »Neuling« muss sich bis dahin so in die Arbeitsprozesse des Teams eingewöhnt haben, dass er ein Gefühl dafür hat, welche Änderungen er in eigener Initiative durchführen kann und welche vorher

im Team diskutiert werden müssen. Sämtliche Änderungen werden außerdem automatisch protokolliert und können bei Bedarf rückgängig gemacht werden, wobei es jedoch praktikabler und zeitsparender ist, strittige Punkte vorher zu klären.

Die Zuständigkeiten und Rechte in einem Projekt gründen sich, wie Raymond schreibt, auf das Prinzip »Autorität folgt Verantwortung«. Ein Programmierer wird nur dann beispielsweise den Schreibzugriff bekommen oder über den weiteren Verlauf des Projektes mitbestimmen können, wenn er nicht nur gute Ideen hat und technische Kompetenz beweist, sondern sich auch um weniger beliebte Arbeiten wie Fehlersuche und Dokumentation kümmert und sich an der Kommunikation mit den Benutzern des Projekts in den Mailinglisten und Diskussionsforen beteiligt.

Kommt es zu Meinungsverschiedenheiten, sollen diese nach Möglichkeit einvernehmlich geregelt werden. Ob dies gelingt, hängt zu großen Teilen von der sozialen Kompetenz und dem diplomatischen Geschick des Projektleiters ab. Schlägt die Einigung fehl, dann ziehen sich einzelne Programmierer möglicherweise aus dem Projekt zurück, schlimmstenfalls aber droht eine Spaltung des Projekts, im Englischen »fork«, also »Gabelung« genannt. Eine Fraktion der Programmierer, die mit dem Verlauf des Projekts nicht einverstanden ist, nimmt sich bei einem »fork« den Code und beginnt mit ihm ein neues, oft ähnlich benanntes Projekt nach den eigenen Vorstellungen. Rechtlich gesehen ist das natürlich jederzeit möglich (es ist eine der Grundvoraussetzungen der Freien Software), faktisch aber besteht ein starker sozialer Druck dagegen, weil ein »fork«

meistens zu inkompatiblen Software-Versionen führt, schlimmstenfalls die Zahl der zur Verfügung stehenden Programmierer halbiert und viel zusätzliche Arbeit verursacht. Die wenigen »forks«, die es in prominenten Projekten der Szene über die Jahre gegeben hat, sind allen Beteiligten in bitterer Erinnerung. Manchmal allerdings ist ein »fork« auch die einzige Möglichkeit, einen überfälligen Generationswechsel herbeizuführen oder einen inkompetenten Projektleiter abzulösen. Die Entscheidung wird dann meist dadurch bestätigt, dass sich fast alle Benutzer spontan dem neuen Projekt zuwenden und das alte schnell an Bedeutung verliert.

Raymond schließt aus diesen unausgesprochenen Verhaltensregeln, dass die Programmierer, ohne es sich vielleicht einzugestehen, vom Streben nach sozialer Anerkennung motiviert sind. Durch ihre Arbeit können sie sich innerhalb der Szene und bei den Benutzern einen Ruf erwerben. Als Beleg für diese Theorie führt Raymond an, dass die Liste der Namen derjenigen, die an einem Projekt mitgearbeitet haben, mit sehr großer Sorgfalt behandelt wird – sie ist gewissermaßen das Allerheiligste eines Projekts, und einen Namen aus dieser Liste herauszulöschen, wäre ein Sakrileg.

Eine Erklärung für diese Struktur, so Raymond, könnte in dem aus der Soziologie und Ethnologie bekannten Phänomen der *Schenkkulturen* (engl. *gift cultures*) bestehen. In einer Schenkkultur definiert sich der soziale Status nicht durch das, was man besitzt, sondern vielmehr durch das, was man verschenkt. Man findet solche Schenkkulturen z.B. auf tropischen Inseln, wo wegen günstiger klimatischer Bedingungen keinerlei Ressourcenknappheit

herrscht. Aber auch manche Schichten der westlichen Gesellschaften funktionieren so, denkt man etwa an das Showgeschäft oder allgemein an Schichten mit sehr hohem Wohlstand.

Es steht zu vermuten, dass sich auch eine Bewegung wie die GNU/Linux-Gemeinschaft so erklären lässt: In den westlichen Gesellschaften, zumal bei deren technischen Eliten, ist möglicherweise ein Grad des Wohlstandes erreicht, der die traditionellen Modelle von Karriere und Besitzstandswahrung obsolet werden lässt, so dass sich stattdessen auch hier eine Art Schenkkultur herausbildet.

Trolltech und das Qt-Problem

Konfrontationen zwischen der GNU/Linux-Bewegung und anderen Modellen der Software-Entwicklung sind keine Seltenheit. Der Verlauf solcher Konflikte verrät eine Menge über den inneren Zusammenhalt und die Prinzipientreue der Bewegung, oft sehr zur Überraschung auch eingeweihter Beobachter.

Einer der ersten dieser Konflikte entstand um das Jahr 1998 im Rahmen des KDE-Projektes. KDE verfolgt das Ziel, GNU/Linux mit einer grafischen Oberfläche auszustatten, die der von Microsoft Windows oder auch MacOS vergleichbar ist. (Unix-Systeme verfügten lange Jahre nur über sehr rudimentäre grafische Schnittstellen, die Nicht-Fachleuten kaum zuzumuten waren.) Eine entscheidende Komponente von KDE ist die Qt-Bibliothek, ein Software-Paket, das grafische Elemente wie Schaltflächen, Menüs und Eingabefelder bereitstellt. Qt war von der nor-

wegischen Firma Trolltech entwickelt worden, und zwar als proprietäres Produkt für die Windows- und Mac-Welt. Trolltech versprach sich vom Einsatz unter GNU/Linux eine zusätzliche Verbreitung der Software, und so wurde Qt den KDE-Entwicklern unter einer besonderen Lizenz umsonst zur Verfügung gestellt. Die Lizenz funktionierte nach dem Prinzip »Angucken, aber nicht Anfassen«: Zwar war der Quelltext für die KDE-Entwickler zugänglich, aber Trolltech behielt sich unter anderem das alleinige Recht vor, neue Versionen der Bibliothek herauszubringen.

Qt war also keine Freie Software, und darüber kam es zum Streit. Enthusiastische KDE-Entwickler verwiesen auf die technischen Vorteile von Qt, zu denen es damals keine Alternative in der Freien Software Szene gab. Außerdem könne man Qt schließlich umsonst benutzen und den Quelltext einsehen, und das hielten viele für ausreichend. Oft war auch der Vorwurf zu hören, dass man ein gutes Projekt nicht durch unsinnige Lizenzstreitigkeiten gefährden dürfe. Andere bestanden jedoch darauf, dass Qt eben keine Freie Software sei und dass es darum nicht einmal erlaubt sei, die KDE-Programme (die unter der GPL lizenziert waren) überhaupt mit der Qt-Bibliothek zu verbinden. Als Folge entstanden zwei neue Projekte: Eines, unter dem Namen *Harmony*, begann damit, die Qt-Bibliothek vollständig nachzubauen, um sie schließlich unter der GPL zu veröffentlichen. Ein anderes Projekt, genannt *GNOME*, sollte eine völlig neue grafische Oberfläche entwickeln, die ohne Qt auskam und zum offiziellen Desktop für das GNU-System werden sollte. Einige der FSF nahestehende GNU/Linux-Anbieter begannen bereits damit, KDE vollständig aus ihrem Angebot zu entfernen.

Besonders die Entscheidung für das GNOME-Projekt wurde wiederum scharf kritisiert, denn man fürchtete, das Erscheinungsbild von GNU/Linux würde damit uneinheitlich werden, ganz abgesehen von dem doppelten Aufwand, den eine solche Parallelentwicklung bedeuten würde. Nichtsdestoweniger machten beide Projekte schnell Fortschritte, und die Firma Trolltech sah sich bald zum Einlenken gezwungen, wollte sie ihren Status innerhalb der GNU/Linux-Szene nicht völlig verlieren. Die Qt-Lizenz wurde in mehreren Schritten zu einer echten Freien Software Lizenz abgeändert und mit der GPL kompatibel gemacht. Das Harmony-Projekt war damit überflüssig und wurde eingestellt. KDE wurde als gleichberechtigte Oberfläche auch in FSF-nahe GNU/Linux-Varianten aufgenommen, gleichzeitig aber wurde die Entwicklung von GNOME weiter betrieben. Im Ergebnis gibt es heute zwei grafische Oberflächen für GNU/Linux-Systeme, nämlich KDE und GNOME, die einander im Funktionsumfang weitgehend gleichen. Viele Programme und Komponenten sind zwischen den Oberflächen ohne weiteres austauschbar.

Das BitKeeper-Debakel

Ein ähnlicher Konflikt, aber mit signifikant anderem Ausgang, ergab sich in jüngerer Zeit um die Versionsverwaltung des Linux-Kerns. Unter einer »Versionsverwaltung« versteht man ein System, das aufzeichnet, wer wann welche Änderung am Code eines Projekts gemacht hat und warum. Man benötigt dafür ein eigenes, sehr komplexes

Programm, das die Aktivitäten zahlreicher, meist über mehrere Kontinente und Zeitzonen verteilter Entwickler koordinieren muss. Gerade die Entwicklung des Linux-Kerns ist in dieser Hinsicht besonders anspruchsvoll, weil es sich um eines der prominentesten und aktivsten Freien Software Projekte handelt.

Etwa um das Jahr 2002 zeigte sich, dass Linus Torvalds mit den bis dahin existierenden, freien Werkzeugen zur Versionsverwaltung seiner Aufgabe nicht mehr gerecht werden konnte. Es waren einfach zu viele Änderungsvorschläge und Weiterentwicklungen, die bei ihm eingereicht wurden und die er zu integrieren hatte. Torvalds entschied sich darum, ein System namens *BitKeeper* zu nutzen, das von Larry McVoy, selber einem Mitglied des Linux-Teams, entwickelt worden war. BitKeeper wird als ein proprietäres, also kostenpflichtiges und geschlossenes System vertrieben, was McVoy damit begründete, dass er ohne Lizenzgebühren die Entwicklung nicht hätte finanzieren können. Für das Linux-Projekt würde er allerdings eine Ausnahme machen und das System kostenlos zur Verfügung stellen.

Die Entscheidung führte zu einem Aufschrei in der Szene. Besonders Richard Stallman kritisierte heftig, ein freies Projekt, und noch dazu ein so prominentes wie den Linux-Kern, in irgendeiner Weise von unfreier Software abhängig zu machen. Auch viele Linux-Entwickler zeigten sich irritiert. Torvalds hingegen argumentierte, dass es kein anderes, freies System gäbe, das dieselbe Aufgabe erfüllen könnte. Und tatsächlich hatte sich seine Produktivität, gemessen an der Zahl der Änderungen, die er in den Linux-Kern integrieren konnte, nach dem Umstieg mehr als verdoppelt.

Aber die »Ehe« hielt nur drei Jahre. Andere Entwickler waren nach wie vor unzufrieden damit, von einem proprietären System abhängig zu sein. Schließlich begann einer von ihnen damit, durch *reverse-engineering* das Datenformat von BitKeeper zu bestimmen und es in einem freien Programm nachzubauen. Das wiederum betrachtete McVoy als klaren Regelverstoß: »Ihr könnt gerne mit mir in Konkurrenz treten, aber nicht als Trittbrettfahrer. Löst die Probleme selber, und konkurriert ehrlich. Konkurriert nicht, indem ihr euch meine Lösung anguckt.«[14]

Es kam zu keiner Einigung, und so zog Larry McVoy schließlich sein Angebot der kostenlosen BitKeeper-Nutzung zurück. Das Linux-Team musste sich nach einer anderen Versionsverwaltung umsehen, und Richard Stallman veröffentlichte einen Artikel, in dem er seine ablehnende Haltung in allen Punkten bestätigt sah: »Zum ersten Mal in meinem Leben möchte ich mich bei Larry McVoy bedanken.«[15]

Kleine Distributionskunde

Weltweit gibt es heute mehrere zehntausend Freie Software Projekte, von denen viele auf eigens eingerichteten Serverfarmen wie *sourceforge.net* beheimatet sind. Die Grundsoftware, die man für ein funktionierendes GNU/Linux-System braucht, stammt aus einigen hundert dieser Projekte. Es ist darum ein eigener Arbeitsschritt erforderlich, um diese Bausteine so auszuwählen und zusammenzufügen, dass ein vollständiges, benutzbares System entsteht. Man nennt ein solches Paket aller relevanten Software eine

Distribution. Zahlreiche Projekte und Unternehmen haben sich die Zusammenstellung, die Pflege und den Vertrieb solcher Distributionen zur Aufgabe gemacht; sie sind es, über die ein Anwender mit GNU/Linux in Berührung kommt. Zu den bekanntesten gehören *Red Hat*, *SuSE* und *Mandriva* (ehemals *Mandrake*). Hinter ihnen stehen kommerzielle Unternehmen, die das gebündelte GNU/Linux auf CD-ROM bzw. DVD verkaufen, einschließlich entsprechender Dokumentation und dem Angebot einer Kunden-Hotline.

Eine Sonderstellung nimmt die Distribution des *Debian*-Projekts ein, weil sie ausschließlich von Freiwilligen zusammengestellt und betreut wird, insofern also vielleicht am deutlichsten dem Geist der Bewegung entspricht. Debian ist eine der wenigen Distributionen, die durchgehend die Bezeichnung »GNU/Linux« (statt einfach »Linux«) verwendet; das Projekt verfügt außerdem über sehr strenge Richtlinien, welche Software aufgenommen werden darf und welche nicht (die meisten anderen Distributionen fügen auch proprietäre, closed-source Software hinzu).

Debian gilt jedoch gleichzeitig als recht technisch orientiert und für Laien wenig geeignet. Die *Ubuntu*-Distribution ist in jüngerer Zeit angetreten, um diesen Mangel zu beheben. Das Projekt wird von dem südafrikanischen Unternehmer Mark Shuttleworth gesponsert, der bereits durch seinen Touristen-Flug zur Internationalen Raumstation von sich reden machte. Ubuntu basiert auf der Infrastruktur des Debian-Projekts und wird heute oft als die am leichtesten zu installierende und benutzerfreundlichste GNU/Linux-Variante genannt.

»Erst ignorieren sie dich, dann lachen sie dich aus, dann bekämpfen sie dich, und dann gewinnst du.« – Dieser Ausspruch von Gandhi, der vielen Graswurzelbewegungen ins Stammbuch geschrieben wurde, zeigt auch eine gewisse Gültigkeit für die GNU/Linux-Bewegung und ihre Auseinandersetzung mit den proprietären Software-Giganten, allen voran die Firma Microsoft.

Allerdings nicht unbedingt in der angegebenen Reihenfolge. Schon als »Micro-Soft« (damals noch in anderer Schreibweise) im Jahr 1975 gegründet wurde, hatte sich Bill Gates mit der Szene der Hacker auseinander zu setzen, die Software als ein Allgemeingut betrachteten und Programme mit großer Selbstverständlichkeit untereinander kopierten. In einem heute berühmten »Offenen Brief an die Hobbyisten« versuchte Bill Gates sich bereits 1976 von dieser Idee abzugrenzen. Er schreibt darin: »Wer kann es sich leisten, professionelle Arbeit umsonst zu tun? Welcher Hobbyist kann drei Mann-Jahre ins Programmieren stecken, alle Fehler finden, sein Produkt dokumentieren und es dann umsonst weggeben?«[16]

Die Sätze finden sich heute, kommentarlos, auf dem Umschlag von Stallmans Biographie.

Es sollte über zwei Jahrzehnte dauern, bis die Szene der freien Programmierer wieder auf dem Radarschirm des Unternehmens erschien, das inzwischen zu einem Weltkonzern geworden war. Nach außen hin wurde das GNU/Linux-Phänomen lange Zeit keiner Erwähnung für nötig befunden, aber eine Microsoft-interne Studie zeigte, dass man sich sehr wohl damit auseinander setzte. Das Me-

morandum wurde im Oktober 1998 Eric Raymond zugespielt, der es unter dem Namen »Halloween-Dokument« veröffentlichte und kommentierte.[17]

Die Studie war, wie sich später belegen ließ, tatsächlich für ranghohe Microsoft-Manager geschrieben worden. Der Verfasser räumte darin ein, dass die Open Source Bewegung eine reale Bedrohung für den Konzern darstellte und dass die Qualität dieser Software proprietären Produkten gleichkam oder sie sogar übertraf. Besonders viel Potential habe GNU/Linux dann, wenn die Kommunikation zwischen Rechnern und Programmen nach unabhängigen, weltweit vereinbarten Standards abliefe. Als Gegenmaßnahme empfahl der Verfasser darum recht unverhohlen, dass Microsoft seine Marktmacht ausnutzen sollte, um solche Standards zu untergraben. Man sollte zum Schein auf sie einschwenken, um dann eigene, proprietäre Zusätze hineinzubringen, so dass nur noch Microsoft-Produkte mit diesen erweiterten »Standards« funktionieren würden. »De-commoditizing« hieß dieses Vorgehen im Microsoft-Sprachgebrauch. Eine andere Formel, die ebenfalls intern bei Microsoft geprägt und dann von der unabhängigen Fachwelt übernommen wurde, ist sprechender: »Embrace, Extend, Extinguish«, also etwa: »Mitmachen, Erweitern, Auslöschen«.

Als auf dem Höhepunkt des Dot-com Booms im Jahr 2000 die Open Source Idee salonfähig geworden war, musste sich Microsoft auch öffentlich mit der Bewegung auseinander setzen. Unrühmlich in die Geschichte eingegangen ist die Äußerung von Vizepräsident Jim Allchin, der am 14. Februar 2001 bekanntgab: »Ich bin Amerikaner, ich glaube an den Amerikanischen Weg. Es erfüllt

mich mit Sorge, wenn die Regierung Open Source Projekte unterstützt, und ich glaube nicht, dass wir unsere Politiker gut genug unterrichtet haben, so dass sie die Bedrohung verstehen.«[18]

Allchin bezog sich darin auf Projekte, in denen Software für US-amerikanische Behörden oder das Militär auf Open Source Basis entwickelt wurde, finanziert durch Steuergelder. Mit der »Bedrohung« meinte er offenbar das Copyleft-Prinzip der GPL, also die Tatsache, dass ein Unternehmen GPL-Software nur dann in eigene Produkte einbauen darf, wenn diese Produkte dann selber unter der GPL veröffentlicht werden.

Dass Allchin dies als »unamerikanisch« hinzustellen versuchte, löste freilich nur allgemeines Kopfschütteln aus. Im Juni desselben Jahres war es Microsoft-CEO Steve Ballmer, der noch einmal nachlegte:

»Unter dem Gesichtspunkt des intellektuellen Eigentums ist Linux ein Krebsgeschwür, das alles infiziert, womit es in Berührung kommt.«[19]

In weiten Teilen der Industrie hatte sich jedoch inzwischen die Überzeugung durchgesetzt, dass die freie Verfügbarkeit des Quelltextes große Vorteile brachte, vor allem auf dem Gebiet der Software-Sicherheit. Je mehr unabhängige Beobachter das Innenleben eines Programms untersuchen können, desto größer die Wahrscheinlichkeit, dass auch hartnäckige Fehler und Sicherheitslücken gefunden werden. Auch das Vertrauen der Benutzer in die Software steigt, wenn sich auf diese Weise belegen lässt, dass die Programme keine versteckten Hintertüren enthalten. Gerade dieser letzte Punkt war es, der besonders Regierungen und andere öffentliche Stellen zunehmend miss-

trauisch dagegen machte, dass ihre gesamte software-technische Infrastruktur von einer einzigen amerikanischen Firma stammte und man der Software wie einer *black box* vertrauen musste.

Microsoft entschloss sich darum zu einer Art Appeasement-Politik. In einer groß inszenierten Rede vor der Stern School of Business an der New York University kündigte Microsoft-Chefdenker Craig Mundie ein »Shared Source« Programm an, das die genannten Einwände ausräumen sollte. Ausgewählte Teile des Windows-Betriebssystems würden zur öffentlichen Kontrolle freigegeben, wobei sich Microsoft aber sämtliche Rechte an dem Code vorbehielt.

Es war das übliche »Angucken, aber nicht Anfassen«. Bruce Perens, einer der Mitbegründer der Open Source Initiative, setzte sich dafür ein, dass Microsoft eine klare und eindeutige Antwort aus der Szene bekommen würde. Mit viel Diplomatie gelang es ihm, ein Dokument zu verfassen, das sowohl von Richard Stallman als auch Eric Raymond, Linus Torvalds und noch einigen anderen unterschrieben wurde. Es heißt darin:

»Wir verbuchen einen neuen Triumph für Open Source und Freie Software: Wir sind ein so ernsthafter Konkurrent für Microsoft geworden, dass ihre Manager öffentlich bekanntgeben, dass sie Angst haben. Die einzige Bedrohung, die wir für Microsoft darstellen, ist jedoch das Ende der monopolistischen Geschäftspraktiken. Microsoft ist eingeladen, sich als ein gleichberechtigter Partner zu beteiligen, eine Rolle, in der sich heute viele befinden, von Individuen bis hin zu transnationalen Unternehmen wie IBM und HP. Gleichberechtigung ist aber nicht das, wonach Microsoft sucht [...] Sie hoffen, den Nutzen der Frei-

en Software zu bekommen, ohne diesen Nutzen mit denen zu teilen, die daran mitarbeiten, ihn herzustellen. [...] Microsoft, es ist Zeit, mit uns zusammenzuarbeiten.«[20]

Der Kampf um den Desktop

Als sogenanntes Desktop-System, also bei Arbeitsplatzrechnern und privaten Computern, hat GNU/Linux zwar an Einfluss gewonnen, aber sich noch nicht nennenswert durchsetzen können. Studien, die zum Beispiel auf Besucherzahlen bei bestimmten Websites beruhen, sehen das Betriebssystem seit mehreren Jahren relativ konstant bei einem weltweiten Benutzeranteil von 3% – fast derselbe Wert, den auch MacOS von Apple erreicht, während die übrigen 94% auf Windows entfallen. Zwar sind solche Statistiken immer mit viel Vorsicht zu genießen, aber unbestritten dürfte zumindest die Aussage sein, dass sowohl GNU/Linux als auch MacOS im einstelligen Prozentbereich liegen, und Microsoft nahe bei 90%.

Der Grund für diese ungebrochene Microsoft-Dominanz dürfte vor allem in der Monopol-Politik des Konzerns liegen. Hersteller, die Windows mit ihren PCs ausliefern wollen, werden von Microsoft dazu verpflichtet, *ausschließlich* Windows anzubieten (einschließlich des Aufklebers »Designed for Windows« auf dem Gehäuse). Das Betriebssystem, das mit 50-100 Euro im Endpreis zu Buche schlägt, ist also aus Sicht der Kunden integraler Bestandteil des Geräts.

Die so erreichte Markt-Dominanz erhält sich dann selbst aufrecht: Hersteller von Peripheriegeräten wie

Druckern oder Grafik-Karten betrachten ihre Arbeit oft als erledigt, wenn sie ihrem Produkt eine CD-ROM mit einem Windows-Treiber beilegen können. Das wäre noch nicht weiter tragisch, da die freie Programmierer-Szene in der Regel sehr schnell bei der Hand ist, entsprechende Treiber für GNU/Linux-Systeme zu bauen. Möglich ist das allerdings nur, wenn der Hersteller die Spezifikation seines Produktes offen legt, so dass die freien Programmierer Zugang zu den technischen Details haben. Gerade das haben viele große Hersteller in der Vergangenheit immer wieder mit den erstaunlichsten Argumenten verweigert. Erst langsam beginnen die Firmen zu begreifen, dass GNU/Linux-Treiber (die sie oft noch nicht einmal selber entwickeln müssen) ihrem Geschäft eher förderlich sind als schaden. In der Praxis werden heute alle gängigen PC-Komponenten von GNU/Linux unterstützt; Probleme gibt es nur noch gelegentlich bei sehr neuen Entwicklungen, die erst seit wenigen Wochen oder Monaten auf dem Markt sind und deren Hersteller nicht mit der freien Entwickler-Szene zusammenarbeiten.

Ein weiterer Grund für die Microsoft-Dominanz auf dem Desktop sind schließlich die proprietären Dateiformate, allen voran das der Textverarbeitung *Word*. Die Struktur solcher *.doc*-Dateien ist hoch komplex und wird von Microsoft nicht offiziell bekannt gegeben – zum Teil wohl deshalb, weil das Unternehmen selber nicht über eine vollständige Spezifikation des Formates verfügt. Faktisch ist es allein dadurch definiert, dass das Programm *Microsoft Word*, ein über Jahrzehnte gewachsener Moloch aus Millionen Programmzeilen, solche Dateien lesen und schreiben kann. Es hat darum lange Jahre gedauert, bis freie Pro-

grammierer das Format durch Ausprobieren so weit analysiert hatten, dass sie ihre eigenen Textverarbeitungsprogramme damit kompatibel machen konnten.

Technische Gründe, warum die Benutzer, insbesondere Laien, auf Windows angewiesen wären, werden darum zusehends bedeutungslos. Die grafischen Oberflächen moderner GNU/Linux-Systeme, basierend auf GNOME oder KDE, sind der von Microsoft Windows äquivalent und entsprechend intuitiv zu bedienen. Für das, was die meisten Benutzer üblicherweise mit ihren Rechnern tun – Surfen im Web, E-Mail, Chat sowie das Verfassen von Texten – gibt es entsprechende Anwendungssoftware, die der unter Windows mindestens ebenbürtig ist: Der Web-Browser *Firefox*, die E-Mail-Programme *Thunderbird* oder *Evolution*, der Instant Messenger *Gaim* sowie der Office-Suite *OpenOffice.org*, der von Textverarbeitung über Tabellenkalkulation bis zu Präsentationssoftware alle Komponenten enthält, die Microsoft Office ebenfalls anbietet, und zudem in der Lage ist, Dateien auch in den Microsoft-Formaten zu lesen und zu erzeugen.

Auch jenseits dieser Standard-Programme gibt es kaum eine Aufgabe, für die nicht entsprechende Freie Software existierte, sei das Bildbearbeitung, Finanzbuchhaltung, Videoschnitt, 3D-Modellierung oder Astronomie.

Eine veränderte Welt

Die Schwierigkeiten, Microsoft die Monopolstellung auf dem Desktop-Markt streitig zu machen, dürfen nicht darüber hinwegtäuschen, dass sich die Software-Industrie

durch die Ideen von Freier Software und Open Source längst grundlegend gewandelt hat. Es sind vor allem die Computer hinter den Kulissen, zum Beispiel bei Banken und Versicherungen, aber auch die Server des World Wide Web, die inzwischen zu einem signifikanten Anteil mit Freier Software funktionieren. So werden seit einigen Jahren schon mehr als 60% aller Websites durch den freien Webserver *Apache* realisiert. In der Szene der Web-Anbieter hat sich inzwischen das Kürzel »LAMP« etabliert, was für »Linux, Apache, MySQL, Perl« steht und die typische Technologiepalette bezeichnet, mit der viele Websites programmiert werden. Alle diese Komponenten sind Freie Software.

Auch wenn es um ganze Betriebssysteme geht, ist das proprietäre Modell auf dem Rückzug. Die Firma Sun Microsystems, derzeit Marktführer bei kommerziellen Unix-Systemen, hat im Jahr 2006 ihr komplettes Betriebssystem *Solaris* unter einer Open Source Lizenz veröffentlicht, um die Vorteile des offenen Entwicklungsprozesses zu nutzen (ihren Umsatz erzielt die Firma Sun vornehmlich durch den Verkauf von Hardware und durch Support-Verträge mit großen Kunden, nicht aber durch die Lizenzierung von Software). Auch Apple, neben GNU/Linux der einzige Konkurrent für Microsoft auf dem Desktop-Markt, hat zumindest das Fundament des Betriebssystems MacOS durch Freie Software ersetzt, nämlich durch das nicht ganz so weit verbreitete, aber »liberal« lizenzierte BSD-Unix. Auch mehrere wichtige Anwendungsprogramme auf dem Macintosh, zum Beispiel der Web-Browser *Safari*, stammen inzwischen aus freien Projekten und werden von Apple in Zusammenarbeit mit der Szene weiterentwickelt.

Lediglich das »Kronjuwel« des Apple-Betriebssystems, die hochwertige grafische Oberfläche, ist nach wie vor ein proprietäres Produkt.

Zumindest was die öffentliche Inszenierung betrifft, ist es jedoch der Branchenriese IBM gewesen, der die vielleicht deutlichste Hinwendung zu Freier Software und Open Source vollzogen hat. Im Januar 2001 erklärte IBM-Präsident Sam Palmisano das GNU/Linux-System zur Referenzplattform für alle zukünftigen Entwicklungen des Konzerns.[21] Man habe eingehende strategische Debatten geführt und sei zu dem Schluss gekommen, dass IBM nur durch Zusammenarbeit mit der weltweiten Programmierer-Szene in der Lage sein würde, den Herausforderungen der Zukunft zu begegnen. In der Folge investierte IBM dreistellige Millionenbeträge in den Aufbau entsprechenden Know-hows innerhalb des Konzerns sowie in die Umstellung der meisten eigenen Software-Produktlinien, um sie unter GNU/Linux einsetzen zu können. IBM tritt seither oft als Service-Dienstleister auf, der seinen Kunden das freie Betriebssystem »verkauft« – und zwar nicht die Software selbst, sondern das entsprechende Know-how und die technische Unterstützung. Ein prominentes Beispiel dafür sind die IT-Infrastrukturen mehrerer großer Unternehmen an der New Yorker Wallstreet, die unter der Leitung von IBM auf GNU/Linux umgestellt wurden. Manche IBM-Programmierer werden von dem Konzern auch dafür bezahlt, in freien Projekten mitzuarbeiten, und zwar natürlich in solchen, an denen das Unternehmen ein besonderes Interesse hat.

Obwohl IBM also mit einiger Überzeugungskraft als »Partner der Szene« auftritt, sind die Ziele und Geschäfts-

modelle des Konzerns keineswegs mit denen der Bewegung deckungsgleich. Nach wie vor sind fast alle Software-Produkte, die IBM herstellt, proprietär und müssen vom Kunden mit sehr hohen Lizenzgebühren bezahlt werden – auch dann, wenn IBM sich relativ unverhohlen bei freien Projekten, die permissiv lizenziert sind, »bedient«, also Software aus dem freien Technologie-Pool abschöpft und in eigene Produkte einbaut, die dann proprietär, also ohne Quelltext, an die Kunden verkauft werden. Aus genau diesem Grund verwendet IBM in der Regel auch keinen Code, der unter der GPL lizenziert ist, weil das Copyleft-Prinzip genau dieses Vorgehen verbieten würde.

Auch andere Aspekte der Firmenpolitik von IBM stehen im Widerspruch zu Überzeugungen in der Szene. Dazu gehört beispielsweise das Engagement des Konzerns für eine neue Generation von Kopierschutztechniken, von denen in späteren Kapiteln dieses Buches die Rede sein wird.

Sei nicht böse

Ein anderer Konzern, der seinem Image zufolge eigentlich das Zeug zu einem Partner der Szene haben sollte, glänzt bislang durch eine fast auffällige Abwesenheit: Google.

»Don't be evil« – »sei nicht böse« war das Motto, mit dem Larry Page und Sergey Brin im Jahr 1996 ihr Unternehmen gründeten. Wegen der Unaufdringlichkeit seiner Suchmaschine schnell zum meistbesuchten Site des World Wide Web aufgestiegen, besitzt Google heute eine weltweit verteilte Infrastruktur von mehreren zehntausend Computern. Alle diese Computer laufen unter GNU/Linux –

die technischen Einzelheiten sind jedoch geheim. Darin besteht denn auch der Hauptkritikpunkt der Freien Software Bewegung an dem Unternehmen, nämlich dass der Konzern zwar von den Errungenschaften der Szene profitiert, aber nur sehr wenig an sie zurückgegeben hat. Auch die Anwendungsprogramme, die Google in den letzten Jahren mit erstaunlicher Produktivität und Innovationskraft auf den Markt bringt (Google Mail, Google Desktop, Google Earth, um nur einige zu nennen) sind allesamt proprietäre Software, deren Quelltext nicht offen gelegt wird. Und es gibt noch einen weiteren Bereich, in dem Google keineswegs mit den Überzeugungen der Szene konform geht, so wie übrigens auch der IBM-Konzern nicht: die Frage der Patentierbarkeit von Software.

Der patentierte Mausclick

Es ist der Albtraum jedes unabhängigen Programmierers, dass ihm aus heiterem Himmel ein Brief auf den Tisch flattert, der ihn davon in Kenntnis setzt, dass seine Software ein obskures Patent verletzt, von dem er nie etwas gehört hat, und ihm daher ein Prozess oder exorbitante Lizenzgebühren drohen. Ein solcher Fall ereignete sich im Jahr 1994, als die Firma Unisys überraschend bekanntgab, ein Patent auf das weit verbreitete Grafikformat GIF (Graphics Interchange Format) zu besitzen.[22] Für Programme, die GIF-Bilder herstellen konnten, mussten nun Lizenzgebühren an Unisys bezahlt werden. Auch Betreiber von Websites, die GIF-Bilder lediglich *verwendeten*, hätten unter Umständen belangt werden können, und zwar dann, wenn

sich nachweisen ließ, dass die Bilder von Programmen stammten, die nicht von Unisys lizenziert waren. Es kam zu einem Sturm des Protests. Mehrere Gruppen riefen dazu auf, dem GIF-Format den Rücken zu kehren und stattdessen frei verfügbare Formate wie JPEG oder PNG zu verwenden. Die Umstellung war keineswegs leicht, da GIF-Bilder einige Eigenschaften hatten, die nur von dem damals recht neuen Format PNG erreicht werden konnten. Es dauerte jedoch mehrere Jahre, bis die einschlägigen Web-Browser und Grafikprogramme das PNG-Format verlässlich unterstützten. Zwar sind die Patente auf GIF im Jahr 2004 ausgelaufen, aber die Affäre hat das Format nachhaltig in Misskredit gebracht und anderen Verfahren Vorschub geleistet.

Ob Software überhaupt patentiert werden kann oder sollte, ist umstritten. Bevor Software zu einem weltweiten Massenmarkt wurde, stellte man Computer-Programme mathematischen Formeln oder Algorithmen gleich und hielt sie als solche für nicht patentierbar. Das änderte sich im Laufe der achtziger Jahre zunächst schleichend, als vor allem das US-amerikanische Patentamt immer mehr faktische Software-Patente zuließ. Im Jahr 1995 wurde diese Praxis durch eine patentamtliche Richtlinie bestätigt, allerdings nie in einem Gesetz festgeschrieben. In der Europäischen Union, wo Software-Patente bislang ausdrücklich unzulässig sind, wird die Frage intensiv debattiert.

Die Schwierigkeiten bei Software-Patenten beginnen damit, dass die Einschätzung immaterieller Artefakte und Verfahren, wie es Computer-Programme sind, die Patentämter oft überfordert. Das gilt besonders dann, wenn sich die zugrundeliegende Technik so extrem schnell verändert

wie in der IT-Branche. In der Folge sind häufig »Trivialpatente« vergeben worden, d.h. Patente für Verfahren, die von Programmierern im Zuge ihrer ganz gewöhnlichen Arbeit laufend neu »erfunden« werden. Die *Electronic Frontier Foundation (EFF)*, eine der aktivsten Organisationen im Kampf gegen Software-Patente, hat eine Liste der zehn offensichtlichsten und aus ihrer Sicht damit gefährlichsten Trivialpatente zusammengestellt – zu ihnen gehört das *Einkaufen per Mausclick*, das *Bezahlen im Internet per Kreditkarte* und auch der *Hyperlink*.[23] Es ist diese Trivialität vieler Patente sowie die schier unüberschaubare Zahl existierender Patente überhaupt, die es einem Programmierer praktisch unmöglich machen, sicherzustellen, dass er keines davon verletzt. Software-Entwicklung wird damit zu einem Gang durch ein Minenfeld.

Es kommt hinzu, dass die Patente sich häufig nicht in den Händen der eigentlichen »Erfinder« befinden. Eine besondere Art von Geschäftemachern – die Szene nennt sie verächtlich »Patenthaie« – hat sich darauf spezialisiert, »schlafende« Patente aufzukaufen oder aber neue Patente so clever durch das Anerkennungsverfahren zu schleusen, dass die Behörden nicht bemerken, dass es längst existierende Technologien gibt, die ebenfalls unter das Patent fallen würden. Das Patent wird dann einige Jahre lang in der Schreibtischschublade aufbewahrt, bis die Idee sich möglichst weit verbreitet hat (oft, weil sie viele Male unabhängig neu »erfunden« wurde). Dann wird es durch gezielte Patentklagen zu Geld gemacht. Oft richten sich diese Klagen gegen kleine Unternehmen oder Privatpersonen, die sich keine teuren Anwälte leisten können. Auch dubiose Patente führen so manchmal zum »Erfolg«.

Die großen Mega-Corporations wie IBM, Microsoft oder Hewlett-Packard verfügen ihrerseits über tausende von Software-Patenten und halten ihre Forschungsabteilungen auch dazu an, offensiv neue Patente zu erwerben. Verwendet werden sie aber vor allem zur *Abschreckung*: sollte Konzern A auf die Idee kommen, Konzern B wegen einer Patentverletzung zu belangen, könnte Konzern B sofort mit einer Reihe von Gegenklagen kontern. Beide halten sich auf diese Weise durch ihre Patent-Portfolios wechselseitig in Schach. Ein kleines Unternehmen oder ein unabhängiger Programmierer hat diese Möglichkeit nicht. Es wäre darum ein Leichtes für die großen Konzerne, kleinere Mitbewerber durch gezielte Patentklagen auszuschalten.

Organisationen wie die FSF und die EFF kämpfen darum vehement gegen die Patentierbarkeit von Software. Im Europäischen Parlament erzielten sie im Jahr 2005 einen vorläufigen Erfolg, als eine entsprechende Novellierung des Patentrechts überraschend mit großer Mehrheit abgelehnt wurde. Das Europäische Parlament und die Europäische Kommission sind sich in dieser Frage jedoch nicht einig. Aktivisten befürchten, dass die Kommission versuchen könnte, die Interessen der Patent-Lobby auch am Widerstand des Parlaments vorbei durchzusetzen.[24]

Die Zukunft der GPL

Im Januar 2006 wurde eine öffentliche Debatte zur Ausarbeitung der nächsten Version der General Public License (GPL) begonnen. Da sie eines der Gründungsdokumente der Bewegung ist, und bis heute die meistverbreitete Li-

zenz für Freie Software überhaupt, kommt dieser Debatte einige Bedeutung zu. Obwohl sich am Prinzip der GPL nichts ändern wird, wirft die Debatte doch auch ein bezeichnendes Licht darauf, welches die Fragen sind, die in der Szene und darüber hinaus heute diskutiert werden.

Bereits die Form, in der diese Debatte stattfindet, ist vielsagend: Die erste Version der GPL hatte Stallman im Jahr 1989 noch völlig im Alleingang geschrieben, für die zweite Version von 1991 hatte er sich lediglich mit einigen Rechtsanwälten beraten. Für die neue Version 3 wurde hingegen ein öffentliches Verfahren vereinbart, bei dem hunderte von Vertretern der Szene wie auch der Industrie sich beteiligen. Außerdem soll sichergestellt sein, dass auch einzelne, nicht weiter organisierte Programmierer und Interessierte sich mit ihren Vorschlägen und Einwänden Gehör verschaffen können. Der Zeitraum für die Revision der Lizenz wurde auf ein Jahr festgesetzt; spätestens im Frühjahr 2007 soll die neue Version erscheinen. Mehrere internationale Konferenzen werden bis zu diesem Termin durchgeführt werden, um strittige Fragen zu diskutieren. Die letztgültige Entscheidung, welche Änderungen an der Lizenz gemacht werden, behält Stallman sich allerdings vor.

So sind beispielsweise *Patent-Vergeltungs-Klauseln* im Gespräch, um freie Programmierer vor Patentklagen zu schützen. Die Idee besteht darin, dass jemand, der einen Programmierer wegen einer Patentverletzung belangt, dadurch automatisch das Recht verlöre, die unter GPL lizenzierte Software selber einzusetzen.

Ein anderer Aspekt hat mit der Tatsache zu tun, dass Software immer häufiger über das World Wide Web eingesetzt wird. Das eigentliche Programm läuft in diesem Fall

auf dem entfernten Rechner, der den Dienst zur Verfügung stellt, und wird im Browser des Benutzers nur optisch dargestellt. Der ausführbare Code wird also gar nicht »vertrieben« oder auf den Rechner des Benutzers heruntergeladen. Die neue GPL soll nun optionale Klauseln erlauben, die auch bei solcher Software verlangen, dass der Quelltext allgemein zugänglich zur Verfügung gestellt wird.

Um der zunehmenden Vielfalt der Szene Rechnung zu tragen, sind außerdem Änderungen geplant, die eine bessere Verträglichkeit der GPL mit anderen Lizenzen bewirken. Es ist heute aus geringfügigen, formulierungstechnischen Gründen oft nicht möglich, Code unter der GPL mit Code zu kombinieren, der unter einer anderen Lizenz veröffentlicht wurde. Diese Hürden sollen so weit minimiert werden, dass nur noch solche Fälle ausgeschlossen sind, die dem Geist der GPL widersprechen würden.

Der wichtigste, aber auch der am heftigsten diskutierte Änderungspunkt weist jedoch über die Software-Szene im engeren Sinne hinaus, mitten in das Zentrum der Auseinandersetzung, die heute zwischen den Giganten der Medienindustrie, der Politik und der Bevölkerung geführt wird. Es geht um das *Digital Rights Management (DRM)*, worunter man alle Verfahren zusammenfasst, die dafür sorgen, dass ein Anbieter die unerwünschte Verbreitung der von ihm bereitgestellten Daten verhindern kann, sei das ein Musikstück, ein Film, der Text eines Buches oder auch ein Stück Software, etwa ein Computerspiel. DRM ist gewissermaßen die Weiterentwicklung des Kopierschutzes: Während ein klassischer Kopierschutz versucht, Daten an ein physisches Medium zu fesseln (etwa eine Audio-CD), geht DRM davon aus, dass die Daten zwar beweglich sind

(etwa weil sie von einem Online-Musikdienst heruntergeladen werden), aber dass nur bestimmte Personen das Recht haben sollen, etwas mit den Daten anzufangen – die nämlich, die dafür bezahlt haben. Um das zu erreichen, werden kryptographische Verfahren, besondere Software und besondere Hardware eingesetzt, die dafür sorgen, dass beispielsweise nur bestimmte, einzelne Geräte ein Musikstück abspielen können oder dass die Daten nur für einen bestimmten Zeitraum »gültig« sind und sich danach selber »vernichten«, oder dass nur eine bestimmte Anzahl von Sicherungskopien der Daten angefertigt werden dürfen.

DRM ist also die Antwort der Industrie auf die Möglichkeit des freien Austauschs von Information. Es ist der Versuch, die technische Entwicklung wieder einzufangen, zu regulieren und zu kontrollieren. Offensichtlich sind die Ziele der Freien Software Bewegung, die auf der Idee des freien Austauschs von Informationen beruht, einem solchen »Digital Rights Management« fundamental entgegengesetzt. Stallman, in seinem typischen Eifer für begriffliche Präzision, weigert sich beispielsweise, den Begriff »Digital Rights Management« auch nur zu benutzen und verlangt, dass man von »Digital Restrictions Management« reden müsse, also von digitalen »Einschränkungen« und nicht etwa digitalen »Rechten«, um so die Propaganda der Industrie zu entlarven.

Auch die neue GPL soll – und das ist der kontroverse Punkt – einige Klauseln enthalten, die dieser Bedrohung des freien Informationsaustauschs entgegenwirken. Die Klauseln, die diskutiert werden, sind dabei vergleichsweise harmlos: Es soll nicht etwa verboten werden, GPL-lizenzierten Code zur Realisierung von DRM-Verfahren zu ver-

wenden. Wenn aber der Code für ein DRM-Verfahren unter der GPL lizenziert ist, dann soll es gleichzeitig erlaubt sein, Programme zu schreiben, die einen freien Zugriff auf die entsprechenden Daten erlauben, das Verfahren soll also legal umgangen werden können. Genau das steht im Widerspruch zum berüchtigten *Digital Millenium Copyright Act (DMCA)*, einem im Jahr 1998 verabschiedeten US-amerikanischen Gesetz, das bereits das Umgehen eines Kopierschutzes unter Strafe stellt.

Obwohl sich die Szene in ihrer Ablehnung von DRM weitgehend einig ist, befürchten doch manche, dass die GPL durch die Einführung solcher Klauseln politisch überladen wird und dass ihr Status als universell in der Szene akzeptiertes Dokument in Gefahr geraten könnte.

Das Thema DRM wird in den folgenden Kapiteln dieses Buchs stets präsent sein. Zunächst geht es um die Entwicklungen im Bereich Kryptographie, die eine Idee wie das »Digital Restrictions Management« überhaupt möglich machen. Anschließend wird von den Anwendungen dieser Verfahren in der Musik- und Filmindustrie die Rede sein, sowie von ihren Auswirkungen auf die Zukunft des Buchs.

Kryptographie

Wie die meisten anderen Bereiche unserer Kultur hat auch die Kryptographie, also die Wissenschaft des Verschlüsselns und Entschlüsselns von Nachrichten, im 20. Jahrhundert mehrere Revolutionen erfahren.

Die wichtigste dieser Revolutionen war ein mathematischer Durchbruch: Es gelang, kryptographische Verfahren zu entwickeln, bei denen Sender und Empfänger einer Nachricht sich nicht vorher auf einen geheimen Schlüssel einigen müssen. Dieser Durchbruch ist der Grund, warum heute zum Beispiel Bankgeschäfte im Internet überhaupt möglich sind.

Die zweite Revolution besteht darin, dass Programme, die diese neuen Verfahren realisieren, heute für jeden frei erhältlich sind, und nicht etwa nur Geheimdiensten oder der Industrie zur Verfügung stehen. Diese Tatsache führte zu einer schweren »Identitätskrise« der Geheimdienste und hat weitreichende Auswirkungen auf Fragen der Privatsphäre, staatlicher Souveränität und gesellschaftlicher Sicherheit. Soll jeder Bürger das Recht haben, Informationen so zu verschlüsseln, dass nicht einmal die mächtigsten Geheimdienste der Welt diese Verschlüsselung aufbrechen können?

Geheime und öffentliche Schlüssel

Fast alle Verfahren, die in der Geschichte der Kryptographie bis zum späten 20. Jahrhundert eingesetzt wurden, hatten eines gemein: Sender und Empfänger einer Nachricht mussten vorher auf einem sicheren Weg einen geheimen Schlüssel miteinander austauschen. Das galt zum Beispiel auch für die von den Deutschen im Zweiten Weltkrieg eingesetzte, berühmte Maschine *Enigma*. Sie war zu ihrer Zeit die fortschrittlichste Methode der Kryptographie auf der Welt, und die Geschichte ihrer Entwicklung und letztendlichen Überwindung durch die Gegner ist bis heute lehrreich.[25]

Die Enigma war ein tragbares Gerät ähnlich einer Schreibmaschine, in deren Innerem sich mehrere rotierende Walzen und eine Matrix von Steckkontakten befanden. Jede Einheit wie zum Beispiel eine U-Boot-Besatzung musste, um die verschlüsselten Befehle des Oberkommandos entgegennehmen zu können, über eine solche Enigma verfügen. Man ging darum beim Entwurf der Maschine davon aus, dass Exemplare der Enigma früher oder später auch in die Hände der Gegner fallen würden. Die Sicherheit des Verfahrens durfte davon nicht abhängen.

Um die Enigma benutzen zu können, benötigte man darum außerdem einen *Schlüssel*, das heißt eine Buchstabenkombination, die angab, wie die verschiedenen Codierwalzen und Steckkontakte der Enigma eingestellt werden mussten, um eine bestimmte Nachricht zu senden oder zu empfangen. Nur wer über eine Enigma-Maschine *und* über den passenden Schlüssel verfügte, konnte verschlüsselte Nachrichten austauschen. Die Schlüssel wiederum wurden

täglich gewechselt. Eine U-Boot-Besatzung erhielt beispielsweise beim Auslaufen aus dem Heimathafen eine Liste von Schlüsseln für die nächsten Monate in Form eines Code-Buches. Fiel ein solches Code-Buch in gegnerische Hände, dann waren zwar die darin enthaltenen Schlüssel unbrauchbar, aber der Schaden blieb insgesamt beschränkt.

Dass die Enigma schließlich doch überwunden werden konnte, war eine Meisterleistung alliierter Mathematiker. Natürlich kann man jedes kryptographische Verfahren, das auf einem Schlüssel beruht, *im Prinzip* dadurch überwinden, dass man alle möglichen Schlüssel ausprobiert – derjenige Schlüssel, bei dem sich eine sinnvolle Nachricht ergibt, ist der richtige. Man spricht dann von der »brute force« Methode, also dem Knacken des Codes mithilfe »roher Gewalt«. Dieser Ansatz funktioniert natürlich dann nicht, wenn die Menge der möglichen Schlüssel so groß ist, dass das Ausprobieren unvertretbar lange dauern würde. Bei der Enigma war das der Fall: Je nach Modell gab es bis zu 200 Trilliarden mögliche Schlüssel. Als Zweierpotenz geschrieben, entspricht dieser Wert etwa der Zahl 2^{77}, man würde also 77 Nullen und Einsen benötigen, um einen Enigma-Schlüssel digital darzustellen. Man sagt darum auch, die Enigma habe eine Schlüssellänge von 77 bit gehabt. Würde jemand in jeder Minute einen dieser Schlüssel ausprobieren (im Vor-Computer-Zeitalter ein durchaus realistischer Wert), dann bräuchte er bis zum Erfolg viele Billiarden Jahre – das bekannte Alter des Universums ist dagegen nur ein Wimpernschlag.

Durch geniale Einsichten in die mathematische Struktur der Enigma gelang es dem polnischen Mathematiker Marian Rejewski jedoch bereits im Jahr 1932, die Menge

der für eine bestimmte Nachricht in Frage kommenden Schlüssel drastisch zu verkleinern. Mithilfe eigens konstruierter elektromechanischer Rechenapparate ließen sich die verbliebenen Schlüssel in vertretbarer Zeit durchprobieren, so dass der polnische Geheimdienst manche der verschlüsselten Nachrichten abhören konnte. Im Juli 1939, als sich die politische Lage kurz vor dem Ausbruch des Krieges zu verschärfen begann, gab Rejewski seine Erkenntnisse an den britischen Geheimdienst weiter, der ein Team um den späteren Computerpionier Alan Turing auf die Enigma angesetzt hatte. Aufbauend auf Rejewskis Vorarbeit gelang es diesem Team, die Entschlüsselung weiter zu verbessern und mit den Veränderungen, welche die Deutschen im Laufe des Krieges an der Enigma vornahmen, Schritt zu halten. Bis zum Ende des Krieges waren die Alliierten damit in der Lage, einen großen Teil der geheimen deutschen Kommunikation abzuhören. Viele Historiker sind heute der Ansicht, dass sich ohne diesen Erfolg das Kriegsende um ein bis zwei weitere Jahre verzögert hätte.

Als nach dem Zweiten Weltkrieg die Ära der elektronischen Rechenmaschinen begann, wurden Verschlüsselungsverfahren zunehmend in Software realisiert. An die Stelle der Enigma traten mathematische Algorithmen und Programme, die wesentlich mächtiger waren als mechanische Walzen und Steckkontakte, die schnell an die Grenzen der Elektrik und Mechanik stießen. Ein Beispiel für ein solches weit verbreitetes Kryptographie-Verfahren ist der *Data Encryption Standard (DES)*, der in den siebziger Jahren von IBM im Auftrag der amerikanischen Regierung entwickelt wurde. Viele Eigenschaften eines Verfahrens wie DES sind sehr vergleichbar mit denen der Enigma. Wie bei

der Enigma gehört zu einer erfolgreichen Verschlüsselung einerseits der Algorithmus, also die »Rechenvorschrift«, nach der aus dem Klartext der verschlüsselte Text generiert wird. Bei der Enigma wurde diese Rechenvorschrift durch die physische Maschine realisiert, bei DES geschieht das durch ein Programm, das die Rechenvorschrift ausführt. Andererseits ist außerdem ein geheimer Schlüssel erforderlich, also eine Zahlenkombination oder ein Bitmuster, mit dem die Rechenvorschrift »eingestellt« wird. Beim ursprünglichen DES-Verfahren beträgt die Schlüssellänge 56 bit, ist also deutlich geringer als bei der Enigma. Allerdings ist für das DES-Verfahren keine analytische Abkürzung des Bruteforce-Ansatzes bekannt, wie sie Rejewski und Turing für die Enigma gefunden hatten. Wie bei der Enigma besteht die Sicherheit auch nicht darin, dass das Verfahren selbst, also die Rechenvorschrift, geheim wäre: Die Beschreibung ist öffentlich verfügbar und jeder könnte ein entsprechendes Programm schreiben. Die Sicherheit liegt vielmehr darin, dass ohne den geheimen 56-bit-Schlüssel keine Verschlüsselung und Entschlüsselung möglich ist.

Genau darin liegt andererseits auch das größte Problem von Verfahren wie DES, nämlich dass Sender und Empfänger von Nachrichten zuvor über einen getrennten, sicheren Kanal einen Schlüssel austauschen müssen, am besten bei einem persönlichen Treffen. Das Internet, wie wir es heute kennen, wäre damit nicht möglich. Bankgeschäfte und Einkäufe lassen sich nur dann über das Netz abwickeln, wenn Passwörter und Kreditkarten-Nummern verschlüsselt übertragen werden. Wenn man allerdings zum Schlüsselaustausch bei jedem Anbieter persönlich vorstellig werden müsste, würde das System zusammenbrechen.

Die Lösung dieses Problems war eine der größten mathematischen Entdeckungen im 20. Jahrhundert. Sie ist auch darum bemerkenswert, weil sie zweimal erfolgte. Der für den britischen Geheimdienst arbeitende Mathematiker Clifford Cocks entwickelte im Jahr 1973 ein Verfahren, das ohne einen geheimen Schlüsselaustausch funktionierte. Da er jedoch für den Geheimdienst arbeitete, durfte er sein Ergebnis nicht veröffentlichen und war zum Stillschweigen verpflichtet. Etwa drei Jahre später stießen die Amerikaner Ronald Rivest, Adi Shamir und Leonard Adleman auf dasselbe Verfahren und gaben ihm, die Anfangsbuchstaben ihrer Nachnamen benutzend, den Namen *RSA-Algorithmus*. Dass Clifford Cocks ihnen bereits zuvor gekommen war, erfuhren sie und die Öffentlichkeit erst im Jahr 1997, als die Geheimhaltung aufgehoben wurde.

Das RSA-Verfahren revolutionierte die Kryptographie. Seine mathematischen Details zu erklären, würde an dieser Stelle zu weit führen, aber das Prinzip ist einfach. Die Grundidee von RSA ist, dass der Algorithmus nicht mit einem, sondern mit *zwei* Schlüsseln arbeitet. Eine Nachricht wird durch eine Rechenvorschrift verschlüsselt, die mit dem ersten dieser beiden Schlüssel »eingestellt« wird, ganz wie in der klassischen Kryptographie. Anders als bei den klassischen Verfahren ist die Rechenvorschrift jedoch so gewählt, dass sie nicht *umgekehrt* werden kann, auch dann nicht, wenn man den ersten Schlüssel kennt. Man benötigt, um die Nachricht wieder entschlüsseln zu können, eine andere Rechenvorschrift und den zweiten Schlüssel. Die beiden Rechenvorschriften bilden das RSA-Verfahren, sie sind öffentlich bekannt und dokumentiert. Die beiden Schlüssel hingegen sind – etwas vereinfacht

gesagt – zufällig gewählte Zahlenkombinationen, also Bitmuster.

Mithilfe dieses Prinzips können zwei Personen eine verschlüsselte Nachricht austauschen, ohne dass vorher ein geheimer Schlüssel übermittelt werden müsste. Wie das funktioniert, erklären die Kryptographen oft anhand zweier fiktiver Personen namens *Alice* und *Bob*. Alice möchte Bob eine verschlüsselte Nachricht schicken. Damit Alice das tun kann, erzeugt Bob zunächst die beiden Schlüssel für das RSA-Verfahren. Einen der beiden Schlüssel behält Bob für sich, den anderen schickt er an Alice. Er muss dazu keinen sicheren Kanal verwenden, tatsächlich könnte Bob den Schlüssel im Telefonbuch oder auf seiner Homepage veröffentlichen. Alice besorgt sich diesen Schlüssel und benutzt ihn, um die Nachricht an Bob zu verschlüsseln. Nicht einmal sie selbst könnte die Nachricht jetzt wieder entschlüsseln, nur Bob kann das, weil er über den zugehörigen, zweiten Schlüssel verfügt. Man nennt den ersten der beiden Schlüssel auch den *öffentlichen Schlüssel* (engl. *public key*), der zweite ist der zugehörige *geheime Schlüssel* (engl. *private key*).

Man kann sich das Verfahren auch so vorstellen, dass der öffentliche Schlüssel, den Bob an Alice schickt, ein offenes Vorhängeschloss ist, zu dem nur er selber den Schlüssel besitzt. Alice legt ihre Nachricht in eine Kiste, verschließt sie und lässt das Vorhängeschloss einschnappen. Dann verschickt sie die Kiste an Bob, der mit seinem Schlüssel das Vorhängeschloss öffnet und die Nachricht entnimmt.

Das Verfahren funktioniert aber auch andersherum (und hier versagt das Bild mit dem Vorhängeschloss): Bob kann eine Nachricht mit seinem *geheimen* Schlüssel ver-

schlüsseln; sie lässt sich dann nur mit seinem *öffentlichen* Schlüssel wieder entschlüsseln. Da Bobs öffentlicher Schlüssel allseits bekannt ist, bietet diese »Verschlüsselung« natürlich keine Abhörsicherheit. Wenn es Alice gelingt, die Nachricht mit Bobs öffentlichem Schlüssel zu entschlüsseln, dann kann sie jedoch sicher sein, dass die Nachricht wirklich von Bob stammt, denn nur er verfügt über den geheimen Schlüssel, mit dem sich eine solche Nachricht herstellen lässt. Nach diesem Prinzip funktionieren *digitale Unterschriften*. Sie erlauben es, festzustellen, ob eine digitale Information wirklich von dem stammt, der als Autor angegeben ist, und ob die Information auf dem Weg vom Sender zum Empfänger auch nicht verändert wurde.

Fast die gesamte verschlüsselte Kommunikation im Internet beruht heute auf den beiden Spielarten dieses Verfahrens. Man spricht wegen der beiden unterschiedlichen Schlüssel auch von *asymmetrischer Kryptographie* bzw. einem *Public-Key-Verfahren*. Ein Beispiel dafür ist das HTTPS-Protokoll, auf das die heute gängigen Web-Browser umschalten, wenn ein »sicherer« Website aufgerufen wird. Der Benutzer merkt das daran, dass der Anfang der Adresszeile des Browsers von »http://« auf »https://« wechselt, außerdem wird in der Statuszeile das Symbol eines Vorhängeschlosses eingeblendet. Der Benutzer kann dann sicher sein, dass die Verbindung zum Zielrechner erstens abhörsicher verschlüsselt ist und dass er zweitens auch wirklich mit dem Rechner verbunden ist, der in der Adresszeile angegeben ist. Letzteres wird durch die digitale Unterschrift des Zielrechners garantiert. Achtet der Benutzer darauf, dann ist er zum Beispiel sicher vor den in letzter

Zeit häufigen Phishing-Angriffen, bei denen ein Website, der dem einer Bank täuschend ähnlich sieht, dazu benutzt wird, Kontodaten und Geheimzahlen zu erbeuten.

Die Identitätskrise der Geheimdienste

Es gehört zum Selbstverständnis eines jeden Geheimdienstes, dass er einerseits der eigenen Regierung und den eigenen Streitkräften möglichst sichere kryptographische Verfahren zur Verfügung stellt und andererseits alles daransetzt, die Kommunikation möglicher Gegner zu entschlüsseln. Viele Dienste beschäftigen dazu hochkarätige Mathematiker, und sie verfügen außerdem über gigantische Arsenale schierer Computerleistung. Das alles steht unter strenger Geheimhaltung, und so entsteht eine Art Schattenwelt, abgeschlossen vom öffentlichen mathematischen Forschungsbetrieb mit seinen Konferenzen und Publikationen. Nur selten und wenn, dann mit jahrzehntelanger Verzögerung, erfährt die Öffentlichkeit etwas von dem, was in dieser Schattenwelt vorgeht.

Als in den USA in den siebziger Jahren das DES-Verfahren eingeführt wurde, gab es vielfache Spekulationen, dass die US-amerikanische Kryptobehörde, die berüchtigte *National Security Agency (NSA)*, bereits über eine Hintertür zu diesem Algorithmus verfügte, ja, dass der Algorithmus nur deshalb die staatliche Absegnung erhalten hatte, weil die NSA ihn durchbrechen konnte. Eine geheime mathematische Entdeckung war dazu wohl nicht einmal nötig. Wenn, wie vermutet wurde, die NSA über ein Vielfaches der damals kommerziell vorhandenen Compu-

terleistung verfügte, dann ließ sich DES schon durch den reinen Brute-force-Ansatz aushebeln. Heute, wo sich ein Vielfaches der Leistung damaliger Großrechner auf jedem Schreibtisch befindet, geschieht das auch ganz routinemäßig: Bereits mehrfach wurden Wettbewerbe zum Auffinden eines DES-Schlüssels innerhalb weniger Tage gelöst, meist durch das Zusammenschalten großer Mengen von ansonsten unbeschäftigten Computern über das Internet. Die Electronic Frontier Foundation demonstrierte im Jahr 1998 den Bau eines Spezialcomputers, der einen DES-Schlüssel innerhalb weniger Tage finden konnte und lediglich 250.000 US-Dollar gekostet hatte.

Ein sicheres kryptographisches Verfahren, also eines, dem weder ein krimineller Lauscher noch ein Geheimdienst etwas anhaben kann, muss also unter anderem über eine gewisse Mindestlänge des Schlüssels verfügen. Fachleute gehen heute davon aus, dass Schlüssel mit bis zu 80 bit Länge theoretisch in der Reichweite sehr mächtiger Geheimdienste liegen könnten oder bald liegen werden. Moderne Verfahren wie zum Beispiel der *Advanced Encryption Standard (AES)* verfügen dagegen über eine Schlüssellänge von 128 bit oder mehr. Dabei muss man bedenken, dass sich die Zahl der möglichen Schlüssel mit jedem zusätzlichen Bit verdoppelt, was wahrhaft astronomische Verhältnisse ergibt. Es wurde zum Beispiel ausgerechnet, dass man allein zum Aufzählen aller möglichen 128-bit-Schlüssel die Leistung von zehn Großkraftwerken 100 Jahre lang konzentrieren müsste – aus rein thermodynamischen Erwägungen. Damit freilich hätte man die Schlüssel noch keineswegs wirklich ausprobiert, was einen noch viel höheren Aufwand bedeuten würde.

Beim RSA-Verfahren liegen die Verhältnisse etwas anders, denn hier sind die Schlüssel nicht einfach zufällige Bitmuster. Der öffentliche RSA-Schlüssel ist eine Zahl, die durch die Multiplikation zweier sehr großer Primzahlen entsteht (groß heißt hier: Zahlen mit mehreren hundert Stellen); der geheime Schlüssel besteht aus diesen beiden Primzahlen. Um zu einem öffentlichen Schlüssel den zugehörigen geheimen Schlüssel zu finden, muss man also die große Zahl in ihre beiden Faktoren zerlegen – eine Aufgabe, die jeder irgendwann in der Schule gelöst hat, für die es aber bis heute kein effizientes Verfahren gibt: Es hilft nichts, als alle möglichen Faktoren stur durchzuprobieren. Weil es also letztlich um Primzahlen geht und nicht um rein zufällige Bitmuster, sind sichere RSA-Schlüssel sehr viel länger als symmetrische Schlüssel wie beim DES- oder AES-Verfahren. Man geht heute davon aus, dass ein RSA-Schlüssel der Länge 1024 etwa einem symmetrischen Schlüssel von 80 bit entspricht, ein RSA-Schlüssel der Länge 2048 einem von 112 bit und ein Schlüssel der Länge 3072 einem von 128 bit.[26]

Solche Schlüssellängen sind nach allem, was man heute weiß, für einen Brute-force-Ansatz völlig unerreichbar. Zu glauben, dass ein Geheimdienst wie die NSA über eine »Wunderwaffe« verfügt, die diese kombinatorischen Klippen überwinden kann, grenzt an eine Verschwörungstheorie. Natürlich *könnten* Geheimdienste über unvorstellbar viel schnellere Rechner verfügen, als sie der Öffentlichkeit bekannt sind – aber die Idee eines derart radikalen technischen Gefälles gehört wohl doch eher in das Reich der James-Bond-Filme. Natürlich *könnten* die Mathematiker der Geheimdienste zu Durchbrüchen gelangt sein, die alle be-

kannten Krypto-Algorithmen hinfällig werden lassen – aber es ist ebenso evident, dass dieselben Durchbrüche früher oder später von der zahlenmäßig weit größeren Community der öffentlichen mathematischen Forschung wiederholt werden würden. Das tatsächliche Verhalten der Geheimdienste in den vergangenen Jahrzehnten scheint denn auch eher gegen die Existenz solcher »Wunderwaffen« zu sprechen.

Als Rivest, Shamir und Adleman ihre Entdeckung der asymmetrischen Kryptographie gemacht hatten, versuchte die NSA zunächst, die Veröffentlichung zu verhindern.[27] Der Versuch scheiterte an der Zivilcourage der Autoren. Zu Beginn der achtziger Jahre versuchte die NSA, mit den Universitäten eine Vereinbarung zu erreichen, die besagte, dass kryptographisch relevante Ergebnisse vor der Publikation der NSA vorzulegen waren. Für die Nutzung insbesondere durch die Industrie versuchte man außerdem kryptographische Verfahren zu etablieren, bei denen der Schlüssel, oder ein Teil des Schlüssels, bei einer Treuhandstelle hinterlegt werden musste (»key escrow«). Bei einem Straftatverdacht oder einer Gefährdung der nationalen Sicherheit würde der Geheimdienst auf richterliche Anordnung den Schlüssel erhalten und so die Kommunikation abhören können. Das Verfahren setzte sich nicht durch – und das vor allem deshalb, weil die Öffentlichkeit inzwischen über sehr viel mächtigere Werkzeuge verfügte.

Pretty Good Privacy

Im Jahr 1991 – im selben Jahr, als Linus Torvalds den Linux-Kern schrieb – veröffentlichte der amerikanische

Bürgerrechtler Phil Zimmermann das Programm *Pretty Good Privacy*, abgekürzt *PGP*. Es war die erste frei verfügbare Implementierung des RSA-Verfahrens, und die Software war ausdrücklich so verpackt und mit verständlicher Dokumentation versehen, dass auch Laien damit umgehen konnten. Denn genau das war Zimmermanns Ziel: Er wollte »Kryptographie für die Massen« ermöglichen, ausgehend von der Überzeugung, dass Privatsphäre durch starke Kryptographie – also Kryptographie, die auch von Geheimdiensten nicht durchbrochen werden kann – ein Bürgerrecht sein müsse.

»Es ist vertraulich. Es ist privat. Und es geht niemanden etwas an außer Ihnen. Sie planen vielleicht eine politische Kampagne, diskutieren über Ihre Steuern oder haben eine geheime Liebschaft. Oder Sie kommunizieren mit einem Dissidenten in einem repressiven Land. Was immer es ist, Sie wollen nicht, dass Ihre persönliche E-Mail oder Ihre vertraulichen Dokumente von irgendwem sonst gelesen werden. Es ist nichts Falsches daran, seine Privatsphäre einzufordern. Sie ist so selbstverständlich wie die amerikanische Verfassung.«[28]

Mit diesen Sätzen beginnt die Bedienungsanleitung von PGP, und was ihnen folgt, ist ein leidenschaftliches Plädoyer für die Privatsphäre im Zeitalter elektronischer Kommunikation. Zimmermann argumentiert, dass der übliche Einwand: »Wer nichts zu verbergen hat, braucht auch nichts zu verschlüsseln«, nicht verfängt. Beim papierenen Nachrichtenaustausch schreiben wir unsere Mitteilungen schließlich auch nicht offen auf Postkarten, sondern benutzen ganz selbstverständlich Umschläge, auch dann, wenn wir »nichts zu verbergen« haben. Der E-Mail-

Verkehr (er ist in dieser Hinsicht heute nicht anders als 1991) gleicht hingegen dem Verschicken offener Postkarten – jeder, dessen Systeme die Nachricht auf dem Weg vom Sender zum Empfänger durchläuft, kann mitlesen. Wer in dieser Situation, wo die meiste Kommunikation vollkommen offen und im Klartext über das Netz geht, hingegen seine persönliche Post verschlüsselt, erregt damit schon fast automatisch Argwohn – selbst wenn er »nichts zu verbergen« hat. Die Lösung, so Zimmermann, wäre, dass jeder seine elektronische Post einfach grundsätzlich verschlüsselt, ohne dass er sich jedes Mal neu und bewusst dafür entscheiden müsste. PGP sollte genau das ermöglichen.

Die »Briefumschläge«, die Zimmermann den Leuten in die Hand gegeben hatte, waren freilich weitaus effektiver als ihre papierenen Vorbilder – so effektiv, dass kein Staat und kein Geheimdienst sie, im Interesse welcher Verbrechensbekämpfung oder nationalen Sicherheit auch immer, durchdringen konnte.

Natürlich bekam Zimmermann Ärger.

Zur damaligen Zeit galten die Krypto-Algorithmen, die PGP verwendete, als Rüstungstechnologie und durften darum nicht aus den USA exportiert werden. Dies aber war über das Internet längst geschehen, und so nahm die US-Zollbehörde im Jahr 1993 Ermittlungen gegen Zimmermann auf. Die Strafandrohungen waren horrend, und allein die Prozesskosten hätten ihn in den Ruin treiben können. PGP hatte jedoch inzwischen eine breite Anwenderschaft insbesondere bei Menschenrechtsorganisationen und Bürgerrechtlern gewonnen. In kürzester Zeit organisierten sich Unterstützer und begannen damit, Geld für einen eventuellen Prozess zu sammeln. Um die Ex-

portbestimmungen zu unterlaufen, publizierte Zimmermann den Quelltext von PGP zusätzlich in Buchform.[29] Im Vorwort wies er ausdrücklich darauf hin, dass die gewählte Schriftart das Wiedereinscannen mit einem optischen Lesegerät ermöglichen würde – technisch gesehen ein großer Unsinn, denn der Code konnte jederzeit über das Internet heruntergeladen werden. Zimmermann spekulierte jedoch, dass die Behörden die Publikation eines Buches nicht würden verbieten können. Die Strategie ging auf: Im Jahr 1996 wurde das Verfahren eingestellt, ohne dass Anklage erhoben worden war.

PGP wird GPG

Nachdem die exportrechtlichen Probleme durchgestanden waren, versuchte sich Phil Zimmermann in der kommerziellen Vermarktung seiner überaus erfolgreichen Software. Er bemühte sich dabei, drei teilweise widerstrebende Aspekte zu vereinen: Einerseits sollte PGP privaten Benutzern, politischen Dissidenten und Bürgerrechtsorganisationen frei zur Verfügung stehen, andererseits wollte Zimmermann mit dem kommerziellen Vertrieb der Software Geld verdienen. Schließlich und drittens war es für die Vertrauenswürdigkeit des Produktes wesentlich, dass der Quelltext offen im Internet zur Verfügung stand und von Experten auf Schwachstellen kontrolliert werden konnte. Die widerstrebenden Interessen führten zu Unklarheiten und Reibungsverlusten; in kurzer Zeit entstanden mehrere, teilweise miteinander inkompatible Versionen von PGP. Die Probleme wurden noch dadurch verstärkt, dass Zim-

mermann Ärger aufgrund von Software-Patenten bekam: Sowohl der RSA-Algorithmus als auch ein anderer, symmetrischer Algorithmus namens IDEA, den PGP intern verwendete, waren patentiert. Was RSA betraf, glaubte Zimmermann, sich auf eine mündliche Vereinbarung berufen zu können, die ihm erlaubte, den Algorithmus für nicht-kommerzielle Zwecke einzusetzen. Der Halter des Patents, die ursprünglich von Rivest, Shamir und Adleman gegründete Firma RSA Data Security, behauptete hingegen, von so einer Vereinbarung nichts zu wissen.

Die Free Software Foundation hatte sich Zimmermanns Idee einer »Kryptographie für die Massen« schnell zu Eigen gemacht. Der rechtliche Status von PGP war jedoch so unbefriedigend, dass bald klar wurde, dass eine Neuimplementierung als Freie Software erforderlich war: Im September 1999 erschien die erste Version von *GPG*, dem *GNU Privacy Guard* (auch *GnuPG* genannt). GPG wurde von Grund auf neu geschrieben, verwendet also keinerlei Code von PGP, und ist unter der General Public License (GPL) lizenziert. GPG verwendet außerdem keine patentierten Algorithmen: Statt RSA kommt für die asymmetrische Verschlüsselung der *ElGamal-Algorithmus* zum Einsatz, und statt IDEA können mehrere verschiedene, allesamt patentfreie symmetrische Algorithmen verwendet werden. (Das Patent auf RSA ist im Jahr 2001 ausgelaufen; seither unterstützt GPG optional auch diesen Algorithmus. Das IDEA-Patent wird in einzelnen Ländern allerdings noch bis etwa 2010 gültig sein.) Signalkräftige Unterstützung bekam das GPG-Projekt bereits im Jahr 1999, als das deutsche Bundeswirtschaftsministerium die Entwicklung mit 250.000 DM förderte.

Auch 15 Jahre nach Zimmermanns Aufruf »Kryptographie für die Massen!« hat sich an der allgemeinen Praxis des E-Mail-Schreibens kaum etwas geändert. Nach wie vor gehen fast alle Nachrichten im Klartext über die Leitung, nach wie vor kommunizieren die Internet-Nutzer, als schrieben sie ihre persönlichen Mitteilungen offen auf Postkarten. In der Industrie ist die Situation kaum anders; auch innerhalb oder zwischen Unternehmen wird nur höchst selten Verschlüsselung eingesetzt, selbst dann, wenn Daten übermittelt werden, die unter anderen Voraussetzungen strengsten Sicherheitsbestimmungen unterlägen.

Ändern würde sich das vermutlich erst dann, wenn Krypto-Software derart fest in die gängigen E-Mail-Programme integriert wäre, dass es keinerlei Anstrengung mehr bedürfte, ausgehende E-Mail zu verschlüsseln und eingehende zu entschlüsseln. Kryptographie müsste, mit anderen Worten, so transparent sein, wie sie es heute zum Beispiel beim Online-Banking ist, wo sie dem Benutzer, wenn er nicht weiß, worauf er achten muss, gar nicht weiter auffällt (vgl. S. 74).

Dort, wo verschlüsselte E-Mail mitunter wirklich lebenswichtig ist, wird sie freilich sehr wohl eingesetzt, so etwa bei Bürgerrechtsorganisationen wie *amnesty international*, die zu den prominentesten Benutzern von PGP gehören. Dass die Technik allerdings auch für Gruppen wie Al-Qaida »interessant« ist, versteht sich von selbst. Nach den Anschlägen vom 11. September 2001 dauerte es nur wenige Tage, bis Phil Zimmermann sich dem Vorwurf ausgesetzt sah, er habe den Terroristen mittels PGP die Arbeit erleichtert. Am 21. September erschien ein Artikel in der *Washington Post*, in dem Zimmermann mit der Aus-

sage zitiert wurde, er sei »überwältigt von Schuldgefühlen«[30]. Zimmermann widersprach umgehend: Er habe, wie viele andere Amerikaner, über die Tragödie der Anschläge geweint, und er sei nicht glücklich darüber, dass Terroristen seine Software möglicherweise zur Planung benutzt hatten. Nach wie vor gelte jedoch seine Überzeugung, dass die Verfügbarkeit starker Kryptographie für jedermann der Gesellschaft mehr nütze als schade.[31]

Musik

Danke, dass Sie diese CD gekauft haben und die Künstler, Songschreiber, Musiker und alle anderen, die sie hergestellt und ermöglicht haben, unterstützen.

Bitte denken Sie daran, dass diese Aufnahme und ihre Verpackung durch das Urheberrecht geschützt sind. Da Sie nicht der Rechteinhaber sind, sind Sie auch nicht zum Vertrieb berechtigt. Bitte benutzen Sie keine Internet-Dienste, die den illegalen Vertrieb urheberrechtsgeschützter Musik fördern, geben Sie keine illegalen Kopien von CDs weiter und verleihen Sie keine CDs zum Kopieren an andere. Es schädigt die Künstler, die die Musik gemacht haben. Es hat denselben Effekt, als ob Sie eine CD aus einem Laden stehlen würden, ohne dafür zu bezahlen. Einschlägige Gesetze verlangen strenge zivil- und kriminalrechtliche Strafen für die unerlaubte Reproduktion, den Vertrieb und die digitale Übertragung urheberrechtsgeschützter Tonaufnahmen. Legale Downloads finden Sie unter der Adresse www...com.

Man braucht wenig Einfühlungsvermögen, um zu spüren, dass hier eine Industrie in großen Schwierigkeiten steckt. Seit dem Aufkommen des Internet werden die etablierten Strukturen zur Herstellung und Verbreitung von Musik (die ihrerseits erst im 20. Jahrhundert entstanden sind) zunehmend infrage gestellt.

Was bisher geschah

Mit Einführung der *Compact Disc* in den achtziger Jahren war Musik zu einem digitalen Medium geworden, aber die Übertragung solcher Musik-Daten über Computer-Netzwerke war noch bis in die späten neunziger Jahre hinein praktisch unmöglich: Die Datenmenge einer typischen Audio-CD, etwa 600 Megabyte, ließ sich nicht sinnvoll über eine Modem-Verbindung übertragen und fand auch keinen Platz auf einer gängigen Computer-Festplatte. Das änderte sich rapide durch die Einführung und Verbreitung des MP3-Verfahrens, einer Technik, die Audio-Daten durch einen psychoakustischen Trick extrem komprimieren kann – etwa ein Zehntel der ursprünglichen Größe lässt sich erreichen, ohne dass das menschliche Gehirn eine deutliche Qualitätsminderung wahrnimmt. Das MP3-Verfahren, das heute weitgehend synonym für alle Arten digital komprimierter Musik genannt wird, ist dabei keineswegs die einzige Möglichkeit. Manche Programmierer meiden dieses Format auch, da es nicht frei ist, sondern von der deutschen Fraunhofer-Gesellschaft patentiert. Ein ähnliches, aber freies Verfahren ist beispielsweise das Format *Ogg Vorbis*, das in Hörvergleichstests gegenüber MP3 in der Regel besser abschneidet, aber sich im Markt kaum durchsetzen konnte.

Mit diesen Verfahren wurde es möglich, Musik in akzeptabler Qualität über die neuen Computernetze zu übertragen. Was zunächst nur wie eine technische Neuerung unter vielen aussah, wurde über Nacht zu einer Bedrohung für die Medienkonzerne, als der 17-jährige Programmierer Shawn Fanning im Sommer 1999 die Tauschbörse *Napster*

einrichtete. Angemeldete Benutzer konnten durch Napster »sehen«, welche MP3-Dateien die anderen Benutzer auf ihren Festplatten hatten, und auf Wunsch jede beliebige davon direkt vom Rechner des anderen Benutzers herunterladen. Napster war damit das erste *Peer-to-Peer File-sharing* System (auch als *P2P* bezeichnet), also ein System, das gleichberechtigte, private Benutzer zusammenbringt, aber selber keinerlei Daten kopiert oder anbietet.

Die Mächtigkeit dieser Technik zeigte sich fast augenblicklich, als ein unveröffentlichtes Demo der Band Metallica in das Napster-Netzwerk gelangte, sofort weltweit verbreitet wurde und in der Folge von einigen amerikanischen Radiosendern gespielt wurde. Ähnliches geschah kurze Zeit später mit einer noch unveröffentlichten Single von Madonna.

Im Zeitalter analoger Tonaufnahmen und physischer Tonträger hatten sich die Medienkonzerne nach kurzem Kampf damit abgefunden, dass ihre Kunden sich zu Hause Kopien von Schallplatten machten oder sie an Freunde weitergaben. Das Kopieren zum nicht-kommerziellen, persönlichen Gebrauch wurde entweder stillschweigend geduldet oder sogar ausdrücklich von der Strafverfolgung ausgenommen (z.B. im amerikanischen *Audio Home Recording Act*, 1992). Über entsprechende Umlagen, zum Beispiel im Rahmen der deutschen GEMA, wurden die durch das private Kopieren entgangenen Einnahmen summarisch kompensiert. Die eigentlichen Gegner der Medienkonzerne waren die professionellen Raubkopierer, die Kopien in großer Auflage herstellten und in direkte, kommerzielle Konkurrenz zum eigentlichen Produkt traten. Eine Tauschbörse wie Napster stellte diese Rechnung jedoch auf den

Kopf: Plötzlich war das private, nicht-kommerzielle Kopieren den Vertriebswegen der Konzerne mehr als ebenbürtig. Eben Moglen, Rechtsprofessor an der Columbia Law School in New York und Rechtsberater der Free Software Foundation, bringt es auf den Punkt: »Es ist der Industrie nicht möglich, Musik besser zu vertreiben, als Zwölfjährige das können.«[32]

Die zweite Generation

Die Medienkonzerne gingen mit aller Härte gegen Napster vor und erreichten, dass der Dienst eingestellt werden musste. Der Geist allerdings war aus der Flasche. In rascher Folge entstand eine neue Generation von Peer-to-Peer-Systemen, bei denen es keinen zentralen Server mehr gab wie noch bei Napster, sondern ausschließlich gleichberechtigte *peers* – mit der Folge, dass es auch keine einzelne Instanz oder Firma mehr gab, die von der Musikindustrie hätte verklagt werden können.

Zu diesen Systemen der zweiten Generation gehört das *Gnutella*-Protokoll, das von Programmierern der Firma Nullsoft bereits im Jahr 2000 entwickelt worden war. (Das »GNU« im Namen war eine Referenz an das GNU-Projekt, aber es bestand keine offizielle Verbindung.) Nullsoft war zu dieser Zeit gerade vom Online-Giganten AOL gekauft worden, und die neue Geschäftsleitung stoppte das Projekt wegen rechtlicher Bedenken noch vor der Veröffentlichung. Lediglich eine Testversion der Software, die keinen Quelltext enthielt, hatte man einige Tage lang herunterladen können – das allerdings war genug gewesen,

um die Idee hinreichend in der Szene zu verbreiten. Programmierer fanden durch reverse-engineering des Nullsoft-Programms heraus, wie das Protokoll arbeitete, bauten die Software nach und entwickelten sie unabhängig weiter. Zahlreiche Client-Programme (also Software, die der Benutzer auf seinem eigenen Rechner braucht, um auf das *Gnutella*-Netzwerk zuzugreifen) entstanden unter den Namen *LimeWire*, *BearShare*, *Gnucleus* usw. Zum Teil handelt es sich dabei um Freie Software, zum Teil auch um proprietäre Produkte einzelner Firmen. Andere, vergleichbare Peer-to-Peer-Netze sind *FastTrack* (mit den Clients *KaZaA* und *Grokster*) sowie *eDonkey2000*.

Ihre Popularität ist groß. Mehrere Millionen Benutzer sind rund um die Uhr in den verschiedenen Netzen aktiv (bei eDonkey2000, dem derzeitigen Spitzenreiter, sind es nach einer Statistik vom Sommer 2004 drei bis vier Millionen Benutzer, Gnutella kommt auf etwa eine halbe Million gleichzeitiger Nutzer). Sie haben Zugriff auf tausende von Titeln. Besonders die Massenware der Top 40 Charts ist fast augenblicklich verfügbar: Kein Stück, das man nicht innerhalb weniger Minuten gefunden und auf den eigenen Rechner heruntergeladen hätte. Aber auch Anspruchsvolleres muss man nicht lange suchen.

Piraten in Nadelstreifen?

Die Musikindustrie argumentiert, dass die freien Tauschbörsen vor allem die Künstler schädigen – siehe den einleitend zitierten Hinweis auf der CD-Verpackung. Kritiker halten dagegen, dass das so nicht stimmt: Nach den gängi-

gen Verträgen bekommt ein Künstler vom Ladenpreis einer CD nämlich nur um die fünf Prozent. Vom Rest wird, so erklären die Konzerne, die Produktion, die Vervielfältigung, der Vertrieb und das Marketing finanziert. Doch manche sehen das anders. In einem geharnischten Essay rechnet die Rocksängerin Courtney Love vor, wie sich der Deal aus der Sicht einer hypothetischen Band darstellt, die ihr erstes Album veröffentlicht, einen Chart-Hit landet und anschließend auf Welttournee geht. Am Ende der Tour ist die Band bei einer schwarzen Null angekommen – der Konzern aber hat sieben Millionen Dollar an den Musikern verdient. »Was ist Piraterie?«, fragt Courtney Love. »Piraterie bedeutet, das Werk eines Künstlers zu stehlen, ohne auf die Idee zu kommen, dafür zu bezahlen. Ich rede hier nicht von Napster-Software. Ich rede von den Verträgen der großen Labels.«[33]

Ähnlich sieht es John Buckman, Gründer des Internet-Labels *Magnatune*: »Die Plattenfirmen sperren ihre Künstler in Vereinbarungen, die sie für ein Jahrzehnt oder mehr binden. Läuft das Geschäft nicht wie gewünscht, dann stellen die Firmen die Alben nicht mehr her, halten die Künstler aber weiterhin unter Vertrag und zwingen sie damit, Jahr für Jahr neue Alben zu produzieren. Am Ende schulden oft sogar sehr erfolgreiche Künstler ihrer Plattenfirma Geld.«[34]

Zwar ist es richtig, dass manche Künstler in diesem System zu Millionären werden. Verglichen damit, wie viele talentierte Musiker es insgesamt gibt, sind das jedoch nur sehr wenige. Die allermeisten Musiker können in diesem System nicht einmal hoffen, mit der Musik auch nur ihren Lebensunterhalt zu verdienen. Die Krise der Musikindus-

trie besteht jedoch nicht darin, dass infrage gestellt würde, ob sie ihre kulturelle Aufgabe angemessen erfüllt – die Krise besteht darin, dass diese Industrie *ihrer technischen Funktion nach* im Begriff ist, überflüssig zu werden. Die Peer-to-Peer-Netzwerke zeigen, dass Musik ganz ohne physische Tonträger fast augenblicklich rund um den Globus verbreitet werden kann, und das zu vernachlässigbaren Kosten. Gleichzeitig wird die Produktion der Musik durch Fortschritte in der Aufnahmetechnik immer erschwinglicher – man braucht nicht mehr zwingend ein großes Label und hunderttausende von Dollars, um ein radiotaugliches Musikstück aufzunehmen. Auch die Gesetze und Mechanismen des Marketing werden durch das Internet neu definiert.

Und so machen sich manche bereits auf die Suche nach neuen Wegen. Das Label Magnatune bietet beispielsweise alle verlegten Künstler grundsätzlich als kostenlosen MP3-Stream an, ausgehend von der Überlegung, dass die Musik die beste Werbung für sich selbst ist.[35] Gefällt dem Kunden ein Album, kann er es für 5 bis 18 Dollar komplett herunterladen – den genauen Preis bestimmt der Kunde selbst. Ob die Kunden auch tatsächlich kaufen (und nicht etwa den kostenlosen MP3-Stream auf die eigene Festplatte umlenken), bleibt Vertrauenssache. Die Rechnung scheint aufzugehen, wohl nicht zuletzt deshalb, weil Magnatune damit wirbt, dass der Künstler von jedem Verkauf garantiert fünfzig Prozent bekommt.

Manche Bands versuchen es auch ganz auf eigene Faust. Im Herbst 2005 veröffentlichten Harvey Danger aus Seattle ihr neues Album vollständig zum kostenlosen Download auf dem eigenen Website. »Bitte bedient euch«, schrieb die Band dazu. »Wenn's euch gefällt, gebt's an eure

Freunde weiter.« Parallel konnte man eine konventionelle CD bestellen, aufwändig gestaltet und mit einer besonderen Bonus-Disc versehen, die es nicht im Internet gab. Die Band hoffte, dass die zusätzliche Publicity über das Internet letzten Endes auch die CD-Verkäufe, besonders an enthusiastische Fans, ankurbeln würde. Nach den ersten Monaten zog die Band im Frühjahr 2006 eine positive Bilanz. Das Album war inzwischen 125.000-mal heruntergeladen worden und von der CD-Ausgabe waren 3.400 Exemplare verkauft.[36] Dass insgesamt also doch nur recht wenig CDs verkauft wurden, erklärte sich die Band unter anderem damit, dass sie über fast keine Infrastruktur verfügte, um die CD auch konventionell in den Geschäften anzubieten, sondern alles auf eigene Faust über das Internet abwickelte. Tatsächlich aber sei die Auflage der CD nur einer von vielen Faktoren in der Popularität der Band.

Gerade für unbekannte Musiker bietet das Internet völlig neue Möglichkeiten, ihre Hörer zu finden. Dienste wie *last.fm* und *Pandora* analysieren auf Wunsch das Hörverhalten ihrer Benutzer und schlagen ihnen per Ähnlichkeitsanalyse Musik vor, die sie erstens noch nicht gehört haben und die ihnen zweitens vermutlich gefallen dürfte. Sogar der Bekanntheitsgrad der neu vorgeschlagenen Musiker läßt sich per Schieberegler einstellen, von Mainstream bis zum Independent-Geheimtipp.

Auf die Frage angesprochen, wie Musiker gerechter entlohnt werden könnten, skizzierte Richard Stallman ein System, das realisiert werden könnte, wenn es ein wirklich einfaches, weit verbreitetes Bezahlsystem im Internet gäbe, bei dem man nicht erst umständlich seine Kreditkarten-Details eingeben müsste. Man könnte dann, so Stallman,

in die Abspielsoftware einen Knopf integrieren: »Gib dem Musiker, den ich gerade höre, einen Dollar.« Stallman ist davon überzeugt, dass viele einen solchen Knopf auch benutzen würden und die Musiker wahrscheinlich besser leben würden als heute.

Gesetze der großen Zahlen

Die Musikindustrie ist von solchen Ideen naturgemäß wenig angetan. Sie betrachtet den freien Austausch der Daten über das Netz als durchweg illegal und geht mit großer Wucht dagegen vor. Da die Peer-to-Peer-Netzwerke der zweiten Generation keine zentrale Instanz mehr kennen, die rechtlich belangt werden könnte, und man schlechterdings auch nicht gegen Millionen von Benutzern Prozesse führen kann, verlegt sich die Industrie auf das Gießkannenprinzip. Zufällig ausgewählten Endbenutzern flattern Anzeigen ins Haus – manchmal solchen, die besonders große Mengen von Musik transferieren, manchmal allerdings auch einer schon verstorbenen Rentnerin oder einem strafunmündigen Teenager. Das sorgt bisweilen für Heiterkeit, aber die Industrie zeigt wenig Sinn für Humor. Anfang des Jahres 2006 lasen Besucher, die sich auf den Website des inzwischen geschlossenen Filesharing-Providers *Grokster* verirrten:

IHRE IP-ADRESSE IST 80.185.167.21 UND WURDE GESPEICHERT.
Glauben Sie nicht, man könnte Sie nicht erwischen. Sie sind nicht anonym.[37]

Die Rechnung, durch einige wenige Prozesse ein Exempel zu statuieren, ging freilich nicht auf. Weder schien das Unrechtsbewusstsein der Filesharing-Nutzer zu steigen, noch konnte die außerordentlich geringe Wahrscheinlichkeit, wirklich zu denen zu gehören, die »erwischt« wurden, die Tauschbörsen signifikant entvölkern. Und so scheint die Industrie inzwischen gewillt, den Kampf in ganz andere Dimensionen auszuweiten: Statt einiger Dutzend soll sich die Zahl der eingeleiteten Verfahren inzwischen auf etwa 18.000 in den USA und etwa 5.500 in Europa belaufen.[38] Geht man grob davon aus, dass die Zahl der Filesharing-Nutzer um den Faktor tausend größer ist, dann rückt das die Wahrscheinlichkeit, »erwischt« zu werden, immerhin in den Promillebereich. Zum Vergleich: Das ist etwa zehnmal wahrscheinlicher, als in Deutschland im Verlauf eines Jahres durch einen Verkehrsunfall ums Leben zu kommen.[39]

Drei Szenarien, wie dieser Kampf zwischen den Benutzern und der Musikindustrie ausgeht, sind vorstellbar. Das erste wäre, dass sich der freie Austausch von Musikstücken mithilfe der Strafandrohungen so weit zurückdrängen lässt, dass neue, kostenpflichtige Vertriebsmechanismen sich behaupten können. Zu den bekanntesten dieser neuen Plattformen gehört der Apple *iTunes Music Store*, viele andere Angebote sind im Aufbau. Die Preise für ein Musikstück auf diesen Sites sind freilich an den existierenden Marktstrukturen orientiert, zum Beispiel 99 Cent pro Titel. Die Künstler werden jedoch weiter nach den existierenden Verträgen bezahlt, bekommen von den 99 Cent also höchstens fünf. Von Sony BMG heißt es zudem, der Konzern würde den Künstlern auch für die Online-Ver-

käufe eine Pauschale für Lagerhaltung und Transport in Rechnung stellen. Die Band Cheap Trick und die Allman Brothers haben im April 2006 dagegen Klage eingereicht.[40] Das System der »legalen« Downloads scheint die bestehenden Strukturen der Musikindustrie bislang mehr oder weniger zu perpetuieren.

Die zweite Möglichkeit wäre, dass sich die Ressourcen der Industrie im Kampf gegen das Filesharing erschöpfen. Eine Weile lang mag es möglich sein, zehntausende von Prozessen zu führen, aber wenn sich die Benutzer dadurch nicht in großer Zahl vom Filesharing abhalten lassen, dann könnte irgendwann die Macht des Faktischen siegen. Die Situation erinnert an Zeiten der Prohibition: Kein Staat kann es sich langfristig leisten, große Teile seiner Bevölkerung zu kriminalisieren.

Am wahrscheinlichsten ist wohl eine dritte Möglichkeit – dass die Benutzer nämlich, wenn ihnen der Boden unter den Füßen zu heiß wird, zu neuen Techniken greifen, um ihre Aktivitäten zu verbergen. Eine neue, dritte Generation von Filesharing-Systemen befindet sich denn auch bereits in der Entwicklung.

Filesharing, dritte Generation

Bislang ist das Filesharing von einer Kultur der Offenheit geprägt: Die Benutzer lassen sich gewissermaßen bereitwillig auf die Festplatten schauen. Sie bleiben in der Regel anonym, denn sie sehen voneinander nur ihre IP-Adressen. Diese Anonymität kann aber durch die Behörden leicht durchbrochen werden, denn Internet-Service-Anbieter sind

verpflichtet, auf Verlangen mitzuteilen, wem wann welche IP-Adresse zugeteilt wurde.

Weil sie Anonymität als ein Bürgerrecht betrachten, entwickeln Gruppen wie die Electronic Frontier Foundation darum eigene Anonymisierungsdienste.[41] Ihr Prinzip ist, dass zwei Internet-Rechner nicht mehr direkt miteinander kommunizieren, sondern über eine Reihe von zufällig ausgewählten Zwischenstationen. Beim Weiterleiten von einer Zwischenstation zur nächsten wird die Nachricht jedes Mal neu verschlüsselt, so dass ein einzelner Rechner, der in der Kette durchlaufen wird, weder weiß, was der Inhalt der Nachricht ist, noch, woher sie stammt oder welches ihr Ziel ist. Erst wenn die Nachricht den eigentlichen Empfänger erreicht, wird sie – wiederum automatisch – in Klartext umgewandelt. Niemand, auch keine Ermittlungsbehörde, kann jetzt noch sagen, woher die Nachricht tatsächlich kam, denn sie lässt sich bestenfalls eine Zwischenstation weit zurückverfolgen. Die einzelnen Zwischenstationen sind dabei gewöhnliche Rechner mit Internet-Verbindung, auf denen die Benutzer eine spezielle Software installieren und sie damit dem Anonymisierungsnetz zur Verfügung stellen.

Das Projekt *Freenet* (nicht zu verwechseln mit dem gleichnamigen deutschen Internet-Provider) geht noch einen Schritt weiter.[42] Hier öffnen die Teilnehmer einander nicht mehr den Blick auf die Daten, die sie ohnehin auf ihren Festplatten gespeichert haben, sondern sie stellen dem Freenet-System ein gewisses Speicherkontingent zur Verfügung, das anfangs leer ist. Klinkt sich ein Rechner ins Freenet ein, dann beginnt das System, in diesem Speicherbereich verschlüsselte Dateien oder Teile von Dateien ab-

zulegen. Kein Benutzer weiß, was für Daten sich im Freenet-Bereich seines eigenen Rechners befinden, erst beim Abruf einer Datei wird sie beim Empfänger wieder zusammengesetzt und entschlüsselt. Das System ist so entworfen, dass sich weder nachverfolgen lässt, woher ein Empfänger eine bestimmte Datei bekommen hat, noch, ob ein bestimmter Rechner überhaupt an Freenet teilnimmt.

Bislang eignen sich diese Systeme noch nicht für den breiten Einsatz, und die Behörden sehen sie eher als ein Spielzeug für Experten denn als zu befürchtendes, neues Massenphänomen. Sollte der Strafverfolgungsdruck auf das Filesharing jedoch gleich bleiben oder sogar noch zunehmen, dann könnten sie sich vielleicht schneller entwickeln und verbreiten als erwartet.

Kampf gegen die eigenen Kunden

Die vielleicht langfristigste Strategie der Musikindustrie besteht darin, die unerwünschte Verbreitung der Inhalte auf technische Weise zu verhindern. Dazu gehören einerseits alle Formen des Kopierschutzes, also Vorkehrungen, die verhindern, dass die Bits von einem Datenträger auf einen anderen übertragen werden können, andererseits das bereits erwähnte Digital Rights Management (DRM), worunter man Verfahren versteht, bei denen ein Inhalt, zum Beispiel ein Musikstück, so markiert wird, dass es nur auf den Geräten desjenigen Benutzers abgespielt werden kann, der dafür bezahlt hat.

Kopiergeschützte CDs lassen sich zum Beispiel herstellen, indem der Hersteller das Datenformat gerade so weit

abändert, dass ein gewöhnlicher CD-Spieler die Daten nach wie vor lesen kann, ein Laufwerk in einem Computer (der die Daten in MP3 konvertieren und ins Netz einspeisen könnte) jedoch in Schwierigkeiten kommt. Angesichts der unüberschaubaren Fülle von Laufwerkstypen und Abspielgeräten sind solche methodischen Verletzungen des CD-Standards allerdings eine Gratwanderung für den Hersteller. Häufig beschweren sich die Kunden zu Recht, wenn die rechtmäßig erworbene CD zwar auf der Heimstereoanlage funktioniert, nicht aber im Autoradio oder im PC, der immer mehr zur Multimedia-Abspielstation wird.

Es war der Sony BMG Konzern, der darum einen Schritt tat, der sich zu einem Public-Relations-Desaster auswachsen sollte. Auf äußerlich unscheinbaren Audio-CDs wurde ein Kopierschutz-Programm namens XCP hinzugefügt, das sich automatisch auf dem PC des Kunden installierte, wenn die CD darauf abgespielt wurde (nur Windows-Rechner waren betroffen, GNU/Linux und MacOS waren immun). XCP war das, was man in Fachkreisen einen *Rootkit* nennt: Ein Programm, das tief in das Betriebssystem des befallenen Rechners eindringt und sich dabei unter anderem selbst unsichtbar macht. Rootkits sind üblicherweise ein Werkzeug, das kriminellen Crackern dazu dient, einen Rechner zu kapern und der eigenen Kontrolle zu unterwerfen, ohne dass der Eigentümer etwas davon bemerkt. Einmal installiert, manipulierte der Sony BMG-Rootkit den Treiber für das CD-Laufwerk so, dass nur noch die Abspielsoftware von Sony BMG auf die Musikdaten zugreifen konnte. Diese Abspielsoftware wiederum erlaubte nur eine begrenzte Anzahl von Kopien der Daten auf andere CDs, und sie unterstützte zum Beispiel auch nicht die

Konvertierung in das Format des iPod von Sony-Konkurrent Apple.

Die betreffenden CDs waren bereits mehrere Monate auf dem Markt, als der amerikanische Programmierer Mark Russinovich im Oktober 2005 zufällig den Rootkit entdeckte. Die Nachricht verbreitete sich in den einschlägigen Internet-Foren innerhalb weniger Stunden und löste einen Sturm der Entrüstung aus. Der Konzern wiegelte zunächst ab. Thomas Hesse, der Präsident des entsprechenden Geschäftsbereichs bei Sony BMG, erklärte in einem Radio-Interview: »Ich glaube, die meisten Benutzer wissen gar nicht, was ein Rootkit ist, also warum sollte sie das kümmern?«[43]

Doch die Empörung wurde immer größer, als weitere Einzelheiten bekannt wurden: Der Rootkit hinterließ auf den betroffenen Rechnern eine Sicherheitslücke und konnte überdies, einmal installiert, gar nicht mehr ohne weiteres entfernt werden. Außerdem wurde die Abspielsoftware dabei erwischt, wie sie »nach Hause telefonierte«: Über die Internet-Verbindung des Benutzers nahm sie Kontakt mit einem Website von Sony BMG auf und meldete dorthin, welche CD der Benutzer gerade anhörte. Der Sinn dieses Mechanismus' war offenbar, automatisch Songtexte und Albumcover für die CD herunterzuladen, so dass die Abspielsoftware diese anzeigen konnte. Der Benutzer wurde jedoch nie über die Existenz dieses Mechanismus' informiert, und es gab auch keine Möglichkeit, ihn abzuschalten. Obwohl Sony BMG es hartnäckig bestritt, hätte das Unternehmen damit auch protokollieren können, welcher Benutzer welche CDs auf seinem Rechner abspielte.

Die Welle der Empörung schwappte bis in die Mainstream-Medien, und Sony BMG sah sich schließlich zum

Einlenken gezwungen. Ein Programm zum Entfernen des Rootkits wurde angeboten und der Konzern entschuldigte sich öffentlich. Alle betroffenen CDs wurden aus den Regalen zurückgerufen, und Sony BMG verpflichtete sich zum Umtausch bereits verkaufter Titel. Nichtsdestoweniger wurden mehrere Klagen auf Schadensersatz gegen Sony BMG eingereicht.

Verräterische Computer und Freie Hardware

Eines haben alle beschriebenen Kopierschutz-Verfahren gemein: Sie funktionieren nicht. Das liegt zum einen daran, dass jedes neue Verfahren sogleich der konzentrierten Aufmerksamkeit und bisweilen dem sportlichen Ehrgeiz vieler tausend Hacker ausgesetzt wird und dass es nur einem davon gelingen muss, den Kopierschutz zu umgehen, um die geschützten Inhalte sofort weltweit verfügbar zu machen. Schließlich aber haben alle Verfahren eine prinzipielle und unüberwindliche Schwachstelle, nämlich die, dass der Kunde seine Musik ja auch irgendwann *anhören* will. Spätestens also wenn sich die Daten, in analoge Schwingungen verwandelt, durch die Kabel auf den Weg zum Lautsprecher machen, kann man sie abfangen und in jedes beliebige freie Format umwandeln.

Den Konzernen ist das nicht verborgen geblieben, und so wird der Kopierschutz zunehmend eher als Indikator für einen Rechtsbruch denn als wirkliche technische Barriere verstanden. Der US-amerikanische Digital Millenium Copyright Act (DMCA) aus dem Jahr 1998 verschärft das Urheberrecht dahingehend, dass nicht mehr nur der illega-

le Vertrieb geschützter Inhalte strafbar ist, sondern bereits das Umgehen des Kopierschutzes.

Unumgehbar wäre ein Verfahren nur dann, wenn es bereits in der Hardware ansetzen würde. Nur wenn die Öffentlichkeit nicht mehr über universell einsetzbare Computer verfügen würde, also über Geräte, die beliebige Bits von A nach B – vom CD-Laufwerk auf die Festplatte, vom Internet in den Arbeitsspeicher oder vom Prozessor in die Soundkarte – befördern können, ließen sich die Daten wieder einfangen und der Kontrolle durch die Konzerne unterwerfen. Und tatsächlich gibt es entsprechende Ansätze. Der bekannteste von ihnen ist die *Trusted Computing Platform Alliance (TCPA)*, ein Konsortium, in dem sich Microsoft, Sony, IBM, Intel und andere Hersteller zusammengeschlossen haben, um die Architektur eines »vertrauenswürdigen Computers« zu definieren. Vertreter der Freien Software Szene sprechen dagegen verächtlich von *treacherous computing*, also einem »verräterischen Computer«, dem zwar die Konzerne »vertrauen« können, nicht aber der Benutzer, der ihn erworben hat. Das Herz dieser Architektur ist ein versiegelter Krypto-Chip in jedem Computer, der einen für das jeweilige Gerät eindeutigen, geheimen Schlüssel enthält. Der Hersteller eines Betriebssystems oder der Verkäufer eines Musikstücks kann seine Daten nun so verschlüsseln, dass sie nur von einem ganz bestimmten Computer gelesen werden können – eben dem Computer, der über den entsprechenden, geheimen Schlüssel verfügt. Da der Schlüssel auf einem Mikrochip sitzt, eingegossen in Kunststoff, ist es für den Endbenutzer unmöglich, den eigenen Schlüssel zu erfahren oder an andere weiterzugeben.

Mit einem solchen archimedischen Punkt im System lässt sich der Kopierschutz effektiv und unumgehbar in Hardware realisieren – immer vorausgesetzt, dass alle Anwender einen solchen TCPA-Computer besitzen und dass sie außerdem nicht mehr über Geräte aus der Zeit vor der TCPA-Ära verfügen. Bedenkt man, wie es der Industrie in den vergangenen Jahrzehnten immer wieder möglich war, auch weit verbreitete Technologien vollständig durch andere abzulösen (die Vinylplatte durch die CD, das analoge Fernsehen durch digitales etc.), dann erscheint ein solches Ansinnen keineswegs mehr unrealistisch. Zumal erste Vorboten davon längst in den Regalen sind: So sind die meisten tragbaren MP3-Player mitnichten universelle Abspielgeräte für MP3-Dateien, sondern so konstruiert, dass die Musikstücke nur mithilfe eines herstellereigenen Programms vom PC auf den Player transferiert werden können. Dieses Programm wiederum bestimmt, was der Benutzer darf und was nicht – zum Beispiel können MP3-Dateien zwar auf den Player übertragen, aber nicht wieder vom Player heruntergelesen werden. Beim Apple iPod und der zugehörigen Software iTunes ist das Verfahren noch so einfach, dass es ohne größeren Aufwand in alternativen Programmen nachgebaut werden konnte. Der Sony-Konzern ging mit seinen jüngsten Geräten einen Schritt weiter: Hier muss die Musik vor dem Hochladen auf den Player erst durch Sonys eigene Software (»SonicStage«) verschlüsselt werden, nur dann kann der Player sie abspielen (und die Daten sind umgekehrt, selbst wenn man es schafft, sie vom Player wieder herunterzuladen, für alle anderen Anwendungen wertlos).

Die Freie Software Szene betrachtet diese Entwicklungen mit einiger Besorgnis, aber auch angriffslustig. Man

spricht inzwischen, analog zur Freien Software, von »Freier Hardware« und meint damit Geräte, die das tun, was ihr Eigentümer von ihnen verlangt, und nicht etwa derjenige, dessen Daten kurzzeitig durch das Gerät wandern. Es geht, mit anderen Worten, um die Verteidigung des universell einsetzbaren Computers, der beliebige Bits von A nach B bewegen kann.

Eben Moglen, der Rechtsberater der Free Software Foundation, erklärt: »Wir werden in den nächsten Jahren einen kreativen, bisweilen ironischen Dialog mit der unfreien Hardware führen. Wir werden zeigen, dass unfreie Hardware befreit werden kann.«

Auf die Erfolgsaussichten dieses Kampfes angesprochen, gibt sich Moglen gelassen: »Mein Eindruck ist, dass der Enthusiasmus unter den Herstellern, unfreie Hardware zu konstruieren, nicht besonders dauerhaft ist. Er hängt vom Druck der Konsumenten ab: Wir sind die Konsumenten. [...] Es geht nicht so sehr darum, die Hersteller dahin zu bringen, das zu tun, was wir wollen, sondern die Konsumenten dahin zu bringen, das zu verlangen, was sie brauchen.«[44]

Film

Grundsätzlich steckt die Filmindustrie in sehr ähnlichen Problemen wie die Musikindustrie. Die Hoheit über das Medium der bewegten Bilder begann zu bröckeln, als in den siebziger Jahren der Videorekorder eingeführt wurde; sie geriet ins Wanken mit Einführung der DVD in den neunziger Jahren, als der Film, wie vorher die Musik, zu einem digitalen Medium wurde und in die Computer und Datenleitungen der Bevölkerung gelangte. Und so werden Spielfilme heute oft in denselben Tauschbörsen gehandelt wie Musikdateien. Aus der Natur der Sache ergeben sich jedoch einige ganz wesentliche Unterschiede zur Musik, die dafür sorgen, dass der Kampf um die Befreiung der Information hier manche andere Wendungen nimmt als dort.

Zunächst einmal ist die Produktion eines typischen Spielfilms um ein Vielfaches teurer als die Produktion einer Musik-CD. Auch abgesehen von gigantischen Hollywood-Epen, die dreistellige Millionenbeträge verschlingen, ist die Produktion eines Spielfilms ein Unterfangen, das tiefe Taschen voraussetzt – und damit eine Industrie, die großes Interesse daran hat, das eingesetzte Kapital auch zurückzubekommen.

Ihr Einkommen erhielt die Filmbranche bis vor sehr kurzer Zeit fast ausschließlich aus der öffentlichen Aufführung in Kinos und (in viel geringerem Maße) dem Fernsehen. Das Geld, das für die Produktion eines Films

ausgegeben worden war, musste typischerweise innerhalb weniger Wochen an den Kinokassen wieder eingespielt werden. Das jedoch ändert sich nun rapide durch das Aufkommen der DVD-basierten Heimkino-Technologie mit ihren Großbildschirmen und Dolby-Surround-Sound. Die Filmindustrie ist in dieser Hinsicht im Begriff, der Musikindustrie sehr viel ähnlicher zu werden: Sie stellt ein Produkt her, dass für die Verwendung zu Hause oder unterwegs gedacht ist, nicht mehr für die ortsgebundene Aufführung an einem öffentlichen Ort.

Wenn man jedoch das Produkt den Kunden selbst in die Hand gibt, wächst die Wahrscheinlichkeit, dass diese damit auch tun, was sie wollen. Und so kursieren denn Filme auch bereits in den einschlägigen Tauschbörsen. Was für Musik ausgezeichnet funktioniert, steckt im Fall des Films allerdings noch in den Anfängen, und das liegt vor allem an den benötigten Datenmengen: Ein Spielfilm im DVD-Format ist etwa um den Faktor tausend größer als ein Musikstück – und dabei ist die Kompression der Videodaten schon mit eingerechnet. In den heutigen Datennetzen ist es noch kaum praktikabel, solche Informationsmengen zu übertragen, jedenfalls nicht in großem Stil. Die Filmindustrie hat somit noch eine gewisse »Gnadenfrist«, bevor sie sich in exakt derselben Situation wiederfinden wird, in der heute die Musikindustrie ist.

Kryptographie für Anfänger

Da die DVD-Technologie sehr viel jünger ist als die Audio-CD, wurde sie bereits mit der Maßgabe entwickelt,

den freien Austausch der gespeicherten Filmdaten möglichst zu verhindern. Bei der Audio-CD stecken die Bits gewissermaßen völlig ungeschützt in der Plastikscheibe, weil es zum Zeitpunkt ihrer Erfindung undenkbar war, dass Privatpersonen sie dort hinein- oder wieder herausbekämen, geschweige denn, sie innerhalb von Sekunden weltweit zu verbreiten. DVDs enthielten dagegen von Anbeginn an einen Kopierschutzmechanismus, der auf Kryptographie beruht. Die Daten sind üblicherweise durch ein Verfahren namens *CSS (Content Scrambling System)* verschlüsselt. Durch einen symmetrischen Verschlüsselungsalgorithmus sorgt dieses Verfahren dafür, dass die Daten nur von solchen Geräten gelesen werden können, die über einen geheimen Player-Schlüssel verfügen. Es gibt etwa 400 solcher Schlüssel, und sie wurden bei Einführung der DVD-Technologie den einzelnen Herstellern zugeteilt. Auf diese Weise sollte sichergestellt werden, dass nur akkreditierte Hersteller Abspielgeräte für DVDs bauen konnten.

So jedenfalls in der Theorie. In der Praxis zeigte sich, dass das Verfahren fast schon verblüffend dilettantisch konzipiert war – so dilettantisch, dass manche Experten es heute als abschreckendes Beispiel für den inkompetenten Einsatz von Kryptographie anführen.

Es beginnt damit, dass der verwendete Algorithmus ein ad hoc entworfenes Verfahren ist, das einer ernsthaften Sicherheitsüberprüfung nicht standhält. Die nominelle Schlüssellänge beträgt nur 40 bit, tatsächlich aber enthält der Algorithmus Schwachstellen, die die effektive Schlüssellänge auf kaum die Hälfte davon reduzieren. Ein gewöhnlicher Heim-PC kann einen solchen Schlüssel in weniger als einem Tag finden.

Womöglich hatte die Industrie aber gar keine andere Wahl, als einen so schlechten Algorithmus einzusetzen, denn jede »ernsthafte« Form von Kryptographie wäre damals, Mitte der neunziger Jahre, unter das amerikanische Exportverbot für Rüstungstechnologie gefallen. Vielleicht ging man auch davon aus, dass die Schwäche des Algorithmus' dadurch kompensiert würde, dass das Verfahren geheim gehalten wurde – ein typisches Beispiel für das, was Experten *security by obscurity*, also etwa »Sicherheit durch Geheimhaltung« nennen. Es besteht in der Fachwelt Konsens, dass *security by obscurity* grundsätzlich nicht funktioniert, weil die Details eines Algorithmus früher oder später doch an die Öffentlichkeit gelangen.

So auch im Fall von CSS. Zwar dauerte es drei Jahre, bis das Verfahren öffentlich bekannt und damit ausgehebelt war, aber das dürfte auch dadurch zu erklären sein, dass die DVD-Technologie einige Jahre brauchte, um überhaupt nennenswerte Verbreitung zu finden. Es war der 15-jährige Norweger Jon Lech Johansen, der im Oktober 1999 ein Programm namens DeCSS veröffentlichte, das die CSS-Verschlüsselung rückgängig machen konnte.

Kurze Zeit später drang die norwegische Polizei in Johansens elterliche Wohnung ein und beschlagnahmte seine Gerätschaften. In dem Verfahren, das daraufhin eröffnet wurde, gab Johansen an, den DeCSS-Code zusammen mit zwei anderen Programmierern geschrieben zu haben, deren Namen er jedoch nicht preisgab. Johansens Verteidigung wurde von der Electronic Frontier Foundation übernommen, die unter anderem argumentierte, dass Johansen nur solche DVDs entschlüsselt hatte, die sich legal in seinem Besitz befanden, und dass es nach norwegischem

Recht erlaubt sei, Kopien zum persönlichen Gebrauch zu machen. Im Januar 2003 wurde Johansen in erster Instanz freigesprochen. Die Anklage ging daraufhin in Revision; im Oktober 2003 erfolgte ein weiterer Freispruch vor der nächsthöheren Instanz. Die Anklage erklärte daraufhin, auf eine weitere Revision verzichten zu wollen.

Das geschah möglicherweise auch deshalb, weil der Geist wieder einmal aus der Flasche war. Aus Solidarität und Protest hatten Aktivisten während des Prozesses begonnen, den CSS-Algorithmus in allen nur erdenklichen Varianten zu publizieren – etwa als T-Shirt oder als Folge von Haiku-Gedichten –, so dass es nicht mehr möglich war, Johansen als alleinigen »Übeltäter« hinzustellen. Heute, im Jahr 2006, sind mehrere praktisch einsetzbare Versionen des CSS-Algorithmus frei erhältlich.

Das ist allerdings auch dringend nötig, denn ohne diesen Code wäre es unter freien Betriebssystemen wie GNU/Linux nicht möglich, DVDs abzuspielen. Weil der rechtliche Status solcher CSS-Module jedoch unklar ist, geht derzeit keine offizielle GNU/Linux-Distribution das Risiko ein, den Code als Bestandteil des Betriebssystems mitzuvertreiben. Der Anwender muss sich diesen Code also nach der Installation noch selber beschaffen. Zwar wird das in der Regel so weit vorbereitet, dass ein einzelnes Kommando oder ein einzelner Mausclick dazu genügt, aber die Tatsache, dass DVDs unter Microsoft Windows oder Apples MacOS »einfach so« abgespielt werden können, während unter GNU/Linux ein winziger, aber entscheidender Schritt des Benutzers erforderlich ist, ist für den Massenmarkt ein nicht zu unterschätzender Faktor.

Erste, zweite bis sechste Welt

Abgesehen vom CSS-Verfahren verfügen DVDs noch über einen weiteren Schutzmechanismus, der davon unabhängig ist. Es handelt sich dabei um das sogenannte *Region Coding*, bei dem DVDs mit einer Zahl markiert werden, die den vorgesehenen Absatzmarkt angibt. Folgende Regionen sind definiert:

Region 1 Bermuda, Kanada, USA und zugehörige Territorien
Region 2 Mittlerer Osten, Europa, Ägypten, Grönland, Japan, Lesotho, Südafrika, Swasiland
Region 3 Südostasien, Hongkong, Macau, Südkorea und Taiwan
Region 4 Mittelamerika, Karibik, Mexiko, Ozeanien (einschließlich Australien und Neuseeland), Südamerika
Region 5 Restliches Afrika, ehemalige Sowjetunion, indischer Subkontinent, Mongolei, Nordkorea
Region 6 China
Region 7 reserviert
Region 8 »internationale Orte« wie Flugzeuge und Kreuzfahrtschiffe

Jedes Abspielgerät ist mit einem entsprechenden Code markiert, der angibt, zu welcher Region das jeweilige Gerät gehört. DVD-Player, die man in deutschen Geschäften kauft, haben zum Beispiel in der Regel den Region Code »2«, erkennbar an einem Logo auf der Rückseite des Geräts. Wird beim Einlegen einer DVD erkannt, dass der

Region Code nicht übereinstimmt, weigert sich das Gerät, die DVD abzuspielen.

Der Sinn dieses Mechanismus' ist natürlich, die weltweite Vermarktung von Filmtiteln stärker kontrollieren zu können. Die Industrie möchte in der Lage sein, denselben Titel in den unterschiedlichen weltweiten Märkten zu unterschiedlichen Preisen und mit unterschiedlichen Zeitfenstern anzubieten. So könnte ein Film zum Beispiel in Nordamerika schon auf DVD erhältlich sein, während in Europa der Kinostart noch gar nicht erfolgt ist. Darüber hinaus geht es natürlich darum, die lukrativen Märkte der westlichen Welt von den professionellen Raubkopierern vor allem in Asien, die dort weitgehend vom Staat toleriert werden, abzuschotten.

In der Praxis scheitert das Verfahren jedoch an der zunehmenden Mobilität der Weltgesellschaft. Gerade die eher zahlungskräftigen Kunden ärgern sich, wenn sie die beim Weihnachtsshopping in New York oder Bangkok gekauften DVDs nicht auf dem heimischen Fernseher anschauen können. Und so kursieren heute im Internet für fast jedes Gerät auf dem Markt Anleitungen, wie man den Region Code umstellen oder ganz abschalten kann (denn natürlich werden die Geräte alle in denselben Fabriken hergestellt und erst nachträglich per Software für den entsprechenden Absatzmarkt »eingestellt« – und was per Software aktiviert wurde, lässt sich auch per Software wieder deaktivieren). Inzwischen bieten selbst manche Hersteller ihre Geräte, zum Teil gegen geringen Aufpreis, in einer Version ohne Region Code an, ohne dass sie deshalb fürchten müssten, aus der Riege der akkreditierten Hersteller ausgeschlossen zu werden.

Das Verfahren ist damit weitgehend ad absurdum geführt beziehungsweise zu einer bloßen Lästigkeit geworden – und in der Regel sind es die ehrlichen, aber technisch unbedarften unter den Kunden, die es mit ganzer Härte trifft.

Das Imperium schlägt zurück

Inzwischen befindet sich eine neue, zweite Generation der DVD-Technologie vor der Markteinführung. Es handelt sich dabei um die konkurrierenden Standards *Blu-Ray* und *HD-DVD*, mit denen hochauflösende Videodaten gespeichert werden können – Daten also in besserer Qualität, als sie der mittlerweile recht betagte Heimfernseher hergibt. Die Speicherkapazität einer Disc wurde dazu mithilfe neuer Verfahren noch einmal um das bis zu Fünffache erhöht.

Gleichzeitig bietet das der Industrie die Gelegenheit, neue Kopierschutzmechanismen einzuführen – und sie scheint fest entschlossen, die Fehler, die zum Fiasko des CSS-Verfahrens geführt haben, nicht zu wiederholen. Das neue Verfahren zur Verschlüsselung der Videodaten heißt *AACS (Advanced Access Content System)* und wird sowohl bei Blu-Ray als auch HD-DVD zum Einsatz kommen. Der Einsicht folgend, dass *security by obscurity* nicht funktioniert, ist AACS in allen Einzelheiten öffentlich dokumentiert.[45]

Das Prinzip ist dasselbe wie bei CSS: Die Videodaten auf einer Disc sind symmetrisch verschlüsselt, und zwar mit einem Schlüssel, der sich auf der Disc selber befindet. Der Schlüssel selber ist wiederum verschlüsselt, und zwar mit einem geheimen Player-Schlüssel, über den nur akkre-

ditierte Geräte verfügen. Als Verschlüsselungs-Algorithmus kommt der Advanced Encryption Standard (AES) zum Einsatz, der offizielle Nachfolger des DES-Algorithmus (vgl. S. 70). Da die Schlüssellänge von AES 128 bit beträgt, ist der Algorithmus selbst für die größten Supercomputer (oder die Gemeinschaft aller PCs im Internet) unverwundbar. Es existieren keine bekannten Schwachstellen.

Das wirklich Neue an AACS ist jedoch das System, mit dem die geheimen Player-Schlüssel vergeben werden. Anders als bei CSS, wo es nur eine Frage der Zeit war, bis die 400 »geheimen« Schlüssel allesamt geknackt oder sonst wie in die Öffentlichkeit gelangt waren, können bei AACS bis zu zwei Milliarden unterschiedliche Schlüssel vergeben werden. Das bedeutet, dass nun im Prinzip jedes einzelne Gerät einen eigenen, unverwechselbaren Schlüssel bekommen kann.

Möglich wird das durch eine neues mathematisches Konzept, die sogenannten Naor-Naor-Lotspiech-Teilmengen-Differenz-Bäume. Sie erlauben nicht nur die große Menge möglicher Schlüssel, sondern auch, einzelne dieser Schlüssel gezielt zu *deaktivieren*. Das bedeutet: Sollte es dazu kommen, dass ein einzelner Player-Schlüssel in die Öffentlichkeit gelangt – zum Beispiel durch eine Indiskretion eines Angestellten in der Fabrik, wo die Player hergestellt werden –, dann kann auf allen zukünftig hergestellten DVDs dieser eine Schlüssel deaktiviert werden, ohne dass davon andere Schlüssel betroffen wären. Alte DVDs könnten also mit diesem Schlüssel nach wie vor abgespielt werden, neue dagegen nicht mehr. Der entwendete Schlüssel wäre damit über kurz oder lang nutzlos.

Obwohl AACS damit eine nicht unbeeindruckende kryptographische Leistung darstellt, reagiert die Fachwelt bislang eher skeptisch. Die renommierte Fachzeitschrift *IEEE Spectrum* rechnete es beispielsweise unter die Technologien, die in den nächsten Jahren am allerwahrscheinlichsten »floppen« werden.[46] Zur Begründung wurden gleich mehrere Szenarien skizziert, wie man das Verfahren aushebeln könnte: Käme zum Beispiel irgendjemand auf der Welt in den Besitz eines geheimen Player-Schlüssels und würde ihn verwenden, um unverschlüsselte Versionen bestimmter Filme ins Netz zu stellen, dann hätten die Unternehmen keine Möglichkeit, herauszufinden, *welcher* Schlüssel es ist, den sie deaktivieren müssten.

Jon Johansen, der Entwickler des DeCSS-Programms, hat sich vorsorglich schon einmal die Internet-Adresse DeAACS.com registrieren lassen.

King Kong geht in die Knie

Es war der Star-Wars-Epiker George Lucas, der am Tag nach der Oscar-Verleihung 2006 mutmaßte, die Zeit der gigantischen Produktionen mit einem Budget von über 100 Millionen Dollar sei wohl vorbei. Als Menetekel sah er das jüngst enttäuschende Abschneiden von Peter Jacksons King Kong an den Kinokassen. In der Zukunft, so Lucas, seien wahrscheinlich nicht mehr als 15 Millionen Dollar pro Film finanzierbar.[47]

Tatsächlich sind die Zahlen durchaus nicht eindeutig. In Deutschland ging im Jahr 2005 die Zahl der Kinobesucher zwar deutlich zurück, aber gleichzeitig nahmen die

Einnahmen durch den Verkauf und Verleih von DVDs zu.

Diese neue Einnahmequelle steht jedoch auf wackligen Füßen. Da, wie die Vergangenheit gezeigt hat, letzten Endes keine Form des Kopierschutzes funktionieren wird, ist es nur eine Frage der Zeit, bis die Kapazität der Datenleitungen so weit angewachsen sein wird, dass den Benutzern jeder beliebige, jemals produzierte Film genauso frei zur Verfügung stehen wird, wie das heute bei Musikstücken der Fall ist: innerhalb von Sekunden, mit einem einzigen Mausclick und kostenlos.

Der entscheidende Unterschied zur Musikindustrie ist jedoch, dass die Filmbranche nicht aus reinen Mittelsmännern besteht, auf die man, wenn ihre Vertriebsinfrastruktur technisch obsolet wird, verzichten könnte. Die Filmindustrie stellt das von ihr vertriebene Produkt selbst her, und das in einem um ein Vielfaches aufwendigeren Prozess, als es bei Musikstücken der Fall ist.

Auf die Musikindustrie könnte die Öffentlichkeit mit anderen Worten verzichten und hätte trotzdem Musik. Auf die Filmindustrie kann sie hingegen nicht verzichten, ohne zugleich das Produkt zu verlieren. Wenn aber auch die von George Lucas geschätzten 15 Millionen Dollar pro Film nicht mehr direkt durch das Zielpublikum zusammenkommen, dann könnte eine Zeit bevorstehen, in der auch Hollywood auf die staatliche Filmförderung angewiesen ist.

Wort

Verglichen mit Musik und Film ist das geschriebene Wort und seine physische Erscheinungsform, das Buch, noch am wenigsten von der digitalen Revolution betroffen. Es gibt keine Tauschbörsen für digitalisierte Bücher, und die Versuche verschiedener Hersteller, »E-Books« auf den Markt zu bringen, stießen bei den Lesern bislang noch auf wenig Gegenliebe.

Dabei sollte das geschriebene Wort für die digitale Welt eigentlich leichte Beute sein. Auf einer einzigen CD-ROM findet der Text von 500 Büchern Platz, und im Arbeitsspeicher eines typischen MP3-Players ließen sich mehr unterbringen, als auch der fleißigste Leser in seinem ganzen Leben bewältigen könnte. Warum sind wir nicht längst auf E-Books umgestiegen und tragen die Bibliothek unseres Lebens immer bei uns, beim Joggen, beim Einkaufen, in der U-Bahn?

Den Grund dafür dürfte Umberto Eco benannt haben, als er im November 2003 einen Vortrag zur Wiedereröffnung der Bibliothek von Alexandria hielt.[48] Das Buch, so stellt Eco darin gelassen fest, ist als Technologie nicht verbesserbar. Es ist nach wie vor das einzige Medium, das es dem Leser erlaubt, längere Texte praktisch ermüdungsfrei aufzunehmen. Keine der bisher vorgeschlagenen Display-Technologien kann damit auch nur annähernd konkurrieren. Auch als Langzeitspeicher ist das Buch bislang unerreicht: Es vermag Worte über hunderte von Jahren aufzu-

bewahren, ohne dass man befürchten muss, das Speicherformat könnte in Vergessenheit geraten oder das Medium physisch unbrauchbar werden; alles Eigenschaften, denen die digitale Welt noch nichts Vergleichbares entgegenzusetzen hat. Die zusätzlichen, sekundären Vorteile des Buchs – dass es keinen Strom benötigt, stoßfest ist, unempfindlich gegen extreme Temperaturen usw. – sind so offensichtlich, dass sie kaum einer Erwähnung bedürfen. Das Buch, so ist Eco sich sicher, wird darum auch weiterhin ein wichtiges, wenn nicht das wichtigste Medium des geschriebenen Wortes bleiben.

Es kommt hinzu, dass sich die E-Books bisher als nicht eben leserfreundlich erwiesen haben. Sobald das geschriebene Wort in digitalisierter Form vorliegt, kann man es natürlich ebenso einfach kopieren und manipulieren wie Audio- oder Videodaten. Um sich nicht über Nacht in derselben Situation wie die Musik- und die Filmindustrie wiederzufinden, haben die E-Book-Hersteller ihre Geräte darum von vornherein mit Lizenzmodellen und Kopierschutzmechanismen ausgestattet, die angesichts der Einfachheit des Mediums Buch sehr bizarr wirken. Ein Buch, einmal erworben, kann man verleihen, verschenken, weiterverkaufen, und man kann es, wenn es ein Kinderbuch ist, zum Beispiel auch denen, die des Selberlesens noch nicht kundig sind, *vorlesen*.

Nicht so bei manchen E-Books. Lawrence Lessig, Autor des Standardwerks *Freie Kultur*,[49] war denn auch ziemlich verblüfft, als ihm seine E-Book-Software weiszumachen versuchte, er sei nicht berechtigt, Ausdrucke oder Kopien von Aristoteles' *Politik* zu machen – ein Text, dessen Autor seit über zwei Jahrtausenden keine Urheber-

rechtsansprüche mehr geltend machen kann. Bei *Alice im Wunderland* behauptete die Software sogar, es sei nicht erlaubt, den Text laut vorzulesen – wobei man zur Ehrenrettung des Herstellers hinzufügen muss, dass es sich um eine missverständliche Formulierung handelte, die sich nur auf die Rechte an der Hörfassung des Buches bezog.

Unübertroffen dürfte jedoch der verbürgte Fall eines Kunden sein, der beschlossen hatte, sein E-Book auf Papier auszudrucken. Als er nach ein paar Seiten feststellte, dass der Drucker falsch eingestellt war, brach er den Vorgang ab, änderte die Einstellung und versuchte es erneut. Aber diesmal verweigerte die Software den Dienst – mit der Begründung, dass man das Buch laut Lizenz nur einmal im Jahr ausdrucken dürfe.[50]

Diese Beispiele mögen amüsant klingen. Sie zeigen aber auch, dass gerade der unbestreitbare Vorteil der digitalen Technik, die einfache, widerstandslose Verfügbarkeit von Information, durch den verzweifelten Versuch seiner Regulierung ad absurdum geführt wird. Im günstigsten, »harmlosesten« Fall bleiben damit Chancen der Digitalisierung ungenutzt, im ungünstigsten Fall – wenn sich solche Technologien in großem Stil durchsetzen sollten – könnten sie zu einer extrem verschärften Regulierung von Urheberrechtsfragen führen. Das wiederum könnte uns in eine Lage versetzen, in der wir schlechter dran sind als vorher: wenn Dinge, die wir heute für selbstverständlich nehmen, plötzlich nicht mehr ohne weiteres erlaubt sind – sei es das Weiterverkaufen eines ausgelesenen Buchs oder die Möglichkeit, einem Freund auf die Schnelle ein paar Seiten kopieren zu können.

Freilich gibt es Bereiche, in denen das Buch auf dem Rückzug ist. Für Lexika, Wörterbücher und sonstige Nachschlagewerke ist die Papierform keineswegs das beste denkbare Medium. Das liegt einerseits daran, dass solche Werke unweigerlich veralten und früher oder später komplett ersetzt werden müssen. Zum anderen lässt Information sich in elektronischen Medien sehr viel leichter auffinden als durch das alphabetische Nachschlagen in einem Buch. Gedruckte Lexika – so vermutet auch Umberto Eco – werden darum sehr bald verschwinden.

Hoch im Kurs stehen gedruckte Bücher wiederum – vielleicht überraschenderweise – in der Welt der Programmierer und sonstigen Computerbenutzer. Fast jeder kann bestätigen, dass man sich an ein unbekanntes Programm leichter herantraut, wenn man ein Referenzhandbuch neben sich auf dem Tisch liegen hat. Programmierer sind darin nicht anders.

Die Frage nach der Zukunft des Buches in der digitalen Welt ist also keine *Überlebensfrage*, wohl aber eine nach der Stellung und Aufgabe des Buches in dieser Welt. Zwei Aspekte, die dabei zunehmend eine Rolle spielen werden, sind die der *Änderbarkeit* und der *Verfügbarkeit* von Informationen in Buchform.

Freie Bücher

In der Frühzeit des Internet war häufig die Überzeugung zu hören, dass die Zukunft der Literatur in Hypertextualität und Interaktivität bestehen würde. An den Texten der Zukunft würde jeder mitschreiben können, sie würden

sich fortwährend verändern und keinen Autor im eigentlichen Sinn mehr kennen. In seinem Alexandria-Vortrag weist Umberto Eco darauf hin, dass solche Ansätze zwar interessant sein können (»ich hoffe, dass die Schulen der Zukunft das lehren werden»), aber dass die Unveränderlichkeit des geschriebenen Wortes auch eine konstitutive Eigenschaft von Literatur ist: »Die tragische Schönheit von Hugos Waterloo besteht darin, dass die Leser fühlen, dass die Dinge unabhängig von ihren Wünschen geschehen. Der Charme tragischer Literatur ist, dass wir spüren, dass ihre Helden ihrem Schicksal hätten entgehen können, aber dass es ihnen nicht gelingt aufgrund ihrer Schwäche, ihres Stolzes oder ihrer Blindheit.«

Eco folgert: »Es gibt Bücher, die wir nicht umschreiben können, weil ihre Funktion darin besteht, uns zu erklären, was Notwendigkeit ist, und nur wenn sie als solche respektiert werden, können sie uns diese Weisheit vermitteln. Ihre repressive Lektion ist unverzichtbar, um eine höhere Form geistiger und moralischer Freiheit zu erreichen.«[51]

Die widerstandslose Änderbarkeit gerade literarischer Texte ist also nicht unbedingt wünschenswert und wohl auch für den Informationsraum der Zukunft wenig wahrscheinlich – dazu ist es zu offensichtlich, dass ein wesentliches Merkmal des geschriebenen Wortes dadurch verloren ginge.

Für andere Arten von Text ist die Änderbarkeit hingegen sehr wesentlich, und dazu gehören zum Beispiel alle Formen von technischer Dokumentation. Die Freie Software Szene sah sich in dieser Hinsicht mit dem Problem des Auseinanderklaffens von Software und den zugehöri-

gen Handbüchern konfrontiert. Da guter Schreibstil auch unter Programmierern nicht häufig zu finden ist, werden die Handbücher oft von anderen Autoren geschrieben und in Verlagen veröffentlicht, die mit der Szene nicht näher zu tun haben. Das führt zu der paradoxen Situation, dass zwar jeder das Recht hat, die Software zu verändern und weiterzugeben, die zugehörigen Handbücher aber durch traditionelle Autorenverträge »blockiert« sind. Die Programmierer können sie nicht selbstständig anpassen, sondern sind in dieser Hinsicht den Autoren und Verlagen ausgeliefert.

Um das zu ändern, brachte die Free Software Foundation im Jahr 2000 die *GNU Free Documentation License (GFDL)* heraus. Sie wendet dieselben Prinzipien, die mit der GPL für Computer-Programme eingeführt wurden (vgl. S. 28), auf schriftliche Dokumente an: Jeder soll die Möglichkeit haben, ein solches Dokument zu kopieren und weiterzuvertreiben, mit oder ohne eigene Änderungen, kommerziell oder nicht-kommerziell. Gleichzeitig stellt die Lizenz sicher, dass der ursprüngliche Autor des Dokuments, wenn er das wünscht, in allen weiteren Versionen genannt werden muss und dass er auch gewisse Teile, zum Beispiel eine philosophische oder politische Präambel, als unveränderlich markieren kann – diese Teile müssen dann in allen folgenden Versionen erhalten bleiben.

Um die Kopier- und Änderbarkeit eines Dokuments zu gewährleisten, verlangt die GFDL außerdem, dass der Text in einem *transparenten Format* zur Verfügung gestellt wird. Um als transparent anerkannt zu werden, muss ein Format maschinenlesbar sein, seine Spezifikation muss öffentlich zugänglich sein, und es muss mit allgemein verfügbaren, nicht-proprietären Werkzeugen bearbeitet werden können.

Es ist bezeichnend für die informationstechnischen Machtstrukturen der Gegenwart, dass kaum eines der heute weit verbreiteten Textformate diese Anforderungen erfüllt. Das *Portable Document Format (PDF)* scheidet beispielsweise aus, weil es nicht modifizierbar ist. Auch das allgegenwärtige *Microsoft Word Format* ist nicht wirklich akzeptabel, weil es zwar inzwischen von nicht-proprietären Programmen wie *OpenOffice.org* bearbeitet werden kann, aber seine Spezifikation dennoch nicht öffentlich verfügbar ist. Zulässig ist hingegen zum Beispiel die Sprache des Web, *HTML* (weil durch internationale Standards definiert), oder das *Rich Text Format (RTF)*. Ein neuer, offener Dokumentenstandard ist beispielsweise das *Open Document Format (ODF)*, das im Umfeld von *OpenOffice.org* entwickelt wurde. Auch Microsoft hat der Kritik an der proprietären Natur des *Word*-Formats inzwischen nachgegeben, sich allerdings nicht an den ODF-Standard angeschlossen, sondern ein eigenes, offenes Format definiert, das von der nächsten Generation der Microsoft Office Programme, *Office 2007*, verwendet werden soll.

Trotz ihrer strengen Forderungen wird die GFDL heute für zahlreiche Software-Handbücher benutzt. Die Texte der Internet-Enzyklopädie Wikipedia (von der im nächsten Kapitel ausführlich die Rede sein wird) sind ebenfalls unter der GFDL veröffentlicht, um sicherzustellen, dass sie nicht eines Tages von cleveren Geschäftemachern in ein proprietäres Produkt umgewandelt werden können.

Ein anderes Lizenzmodell, das inzwischen weite Verbreitung gefunden hat, stammt von der Initiative *Creative Commons*. Unter dem Slogan »Some Rights Reserved« bietet Creative Commons für Künstler und Kreativschaffende

eine »einstellbare« Lizenz an. Der Lizenzgeber kann dabei aus verschiedenen vorgefertigten Klauseln auswählen und auf diese Weise bestimmen, ob sein Werk weitervertrieben werden darf oder nicht (und wenn ja, ob zu kommerziellen oder nur zu nicht-kommerziellen Zwecken), ob es verändert werden darf und ob bei veränderten Versionen der Name des ursprünglichen Autors genannt werden muss oder nicht. Die Creative Commons Initiative unterhält Ableger in zahlreichen Ländern, die dafür sorgen, dass die Lizenzen jeweils an das Rechtssystem des betreffenden Landes angepasst sind. Sie werden nicht nur für Bücher verwendet, sondern ebenso für Musik, Filme oder andere Arten von Werken.

Das Buch im globalen Informationsraum

Die zweite große Frage, die an das Buch im digitalen Zeitalter gestellt wird, ist die Frage nach der *Verfügbarkeit* und der *Auffindbarkeit* der Information. Es ist offensichtlich, dass die Technik hier eine völlig neue Qualität bringt, denn moderne Suchalgorithmen können auch in gigantischen Informationsmengen innerhalb von Sekundenbruchteilen ein bestimmtes Wort oder eine Redewendung finden. Zwar hat das gedankenschnelle Durchsuchen eines Textes nicht das mindeste mit seiner Aneignung zu tun – eine Bibliothek zu durchsuchen heißt noch nicht, sie gelesen zu haben –, aber dass die Möglichkeit, eine bestimmte Stelle im Werk eines Autors verlässlich finden zu können, einen kulturellen Gewinn darstellt, dürften nur wenige bezweifeln.

Dass die Idee, mit Computerhilfe in Büchern zu suchen, Sinn macht, zeigt sich auch daran, dass entsprechende Dienste zunehmend kommerziell angeboten werden, etwa durch Amazon oder Google. Ein Problem dieser Dienste ist jedoch, dass sie den Text zwar durchsuchbar machen, aber nicht »hergeben« wollen. Ein einzelnes Wort in einem tausendseitigen Werk kann man so sehr schnell finden; es wird dem Benutzer mit ein paar Zeilen Kontext angezeigt, ergänzt durch die Seitenzahl in der gedruckten Fassung, wo es gegebenenfalls nachgelesen werden kann. Will aber ein Wissenschaftler eine Worthäufigkeitsanalyse in Musils *Mann ohne Eigenschaften* machen oder statistisch die Eigenheiten von Kafkas Kommasetzung untersuchen, dann werden ihm diese Suchdienste nicht weiterhelfen, er bräuchte dazu *den Text selbst*.

Es kommt hinzu, dass die spezialisierten und restriktiven Buch-Suchmaschinen nicht in den weltweiten Informationsraum eingebunden sind, zu dem das Internet zunehmend wird. Wer heute ein paar Stichwörter in eine der etablierten, globalen Suchmaschinen eingibt, bekommt meist eine Menge höchst relevanter Informationen angezeigt – nicht aber den größten, wahrscheinlich den wichtigsten und besten Teil dessen, was die Menschheit bisher zu einem gegebenen Thema zusammengetragen hat. Dieser ruht, wie der größte Teil eines Eisbergs unter der Wasseroberfläche, in gedruckten Werken in Bibliotheken und ist damit sehr viel schwerer zugänglich.

Schon seit der Frühzeit des Internet gab es darum Projekte, die sich zum Ziel setzten, Bücher in großem Stil zu digitalisieren und ihren Inhalt so besser verfügbar zu machen. Eines der ersten davon, fast älter als das Internet

selbst, war das *Projekt Gutenberg*, das bereits 1971 von einem Studenten an der Universität Illinois gegründet wurde.[52] Sein Ziel ist, eine möglichst umfangreiche digitale Bibliothek aufzubauen, wobei man darauf achtet, nur solche Werke zu übernehmen, deren Urheberrecht bereits ausgelaufen ist. Die Digitalisierungsarbeit wird grundsätzlich von Freiwilligen übernommen und die resultierenden Texte werden in möglichst einfachen, frei zugänglichen Formaten (also z.B. als reiner Text) für jeden zur freien Verfügung gestellt. Die Bibliothek des Projekts Gutenberg umfasst inzwischen etwa 18.000 Titel, pro Woche kommen etwa 50 neue hinzu.

In die Schlagzeilen geraten ist in jüngerer Zeit das Digitalisierungsprojekt des Google-Konzerns, das im Jahr 2004 unter dem Namen *Google Print* gestartet und inzwischen in *Google Book Search* umbenannt wurde.[53] Das Projekt hat eine heftige Debatte ausgelöst, unter anderem deshalb, weil Google auch und gerade Bücher aufnimmt, deren Urheberrecht nicht abgelaufen ist und die im Buchhandel erhältlich sind. Dem Konzern wurden daraufhin Urheberrechtsverletzungen vorgeworfen, geschäftsschädigendes Verhalten den Verlagen gegenüber sowie das generelle Ansinnen, das gedruckte Buch durch das Internet Googlescher Prägung ablösen zu wollen.

Gerade das Gegenteil sei der Fall, konterte das Unternehmen. Viele Verlage und Autoren, so hieß es, würden Google Book Search als willkommene Verkaufsplattform betrachten, und tatsächlich werden in der Regel nur kurze Textpassagen als Suchergebnisse angezeigt, zusammen mit mehreren Links, über die man das Buch sogleich online bestellen kann. Nur wenn der Verlag oder der Autor es aus-

drücklich wünscht, werden ganze Seiten des Buches oder auch der vollständige Text freigegeben. Google sieht das Projekt als Schritt zum Aufbau eines weltweiten Wissensnetzes der Menschheit, das keineswegs das Buch ersetzen, sondern vielmehr die darin aufbewahrten Informationen leichter zugänglich machen solle.

Von einem wirklich freien Austausch von Informationen kann bei Google Book Search allerdings keine Rede sein. Ganz anders als beim Projekt Gutenberg wird der Text der digitalisierten Bücher zentralistisch auf den Google Servern verwaltet und keineswegs der Öffentlichkeit zur freien Verfügung gestellt. Angezeigt werden Suchergebnisse nur in einem Format, das keinerlei Weiterverarbeitung oder auch nur das Ausdrucken eines Textschnipsels zulässt, auch nicht bei Werken, deren Urheberrecht bereits ausgelaufen ist.

Die Freie Software Bewegung ist aus der Grundidee entstanden, dass die freie Verfügbarkeit von Information ein Wert an sich ist und dass es unethisch ist, diesen Wert irgendwelchen »ökonomischen Realitäten« unterzuordnen. Stattdessen, so argumentierte Stallman von Anfang an, sollte man besser nach ökonomischen Alternativen suchen. Er weigerte sich also, die Ökonomie an die erste Stelle zu setzen, und zeigte gleichzeitig, dass die »ökonomischen Realitäten« erstaunlich biegsam sind, wenn man es nur darauf anlegt.

Überträgt man diese Ideen auf das Buch, dann ist klar, wie seine wünschenswerte Zukunft aussehen müsste: Es wäre eine Welt, in der jeder neu erscheinende Titel vom ersten Tag an vollständig und ohne Einschränkungen onli-

ne verfügbar wäre. Eine Welt, in der jedes jemals geschriebene Wort sich augenblicklich auffinden ließe. Wenn das, was weiter oben über die absoluten, uneinholbaren Vorteile des Buches gesagt wurde, stimmt, dann würden die Bücher dadurch keineswegs verschwinden. Mit Sicherheit würde der Buchmarkt ein mittleres Erdbeben aushalten müssen, aber es würde nicht bedeuten, dass keine Bücher mehr verkauft würden. Möglicherweise würden sogar nicht einmal signifikant *weniger* Bücher verkauft werden als heute – ausgenommen vielleicht solche, die das Papier nicht wert sind, auf dem sie gedruckt sind.

Kooperation

Das Internet nur als ein Medium zu betrachten, das Information von wenigen an viele übermitteln kann, also als eine neue Spielart der klassischen Massenmedien, griffe zu kurz. Das wahre Potential dieses Netzwerks besteht darin, dass es jeden mit jedem verbindet und dabei jeden zu einem gleichberechtigten Partner macht. Es führt darum wie von selbst zur Kommunikation und Kooperation zwischen seinen Benutzern.

Während das 20. Jahrhundert für die private Kommunikation über weite Distanzen nur den Brief und das Telefon kannte, brachte das Internet in den wenigen Jahren seines Bestehens eine ganze Palette weiterer, bislang unbekannter Kommunikationsmechanismen hervor. Zu ihnen gehört die *E-Mail*, die viele Elemente des klassischen Briefs besitzt, aber spontaner ist, weil sie jeden Punkt auf dem Globus innerhalb weniger Sekunden erreichen kann. Noch unmittelbarer ist der *Chat*, also das Übermitteln einzelner Textzeilen oder sogar einzelner Tastendrücke an ein oder mehrere Gegenüber. Für die Kommunikation innerhalb einer Gruppe etablierten sich zunächst *Mail-Verteiler* und *Mailing-Listen*, also die elektronische Form des Rundschreibens an mehrere Empfänger. Sie unterscheiden sich darin, ob die Teilnehmer die Liste der Adressaten in ihren Briefen ausdrücklich angeben oder ob ein zentraler Server irgendwo im Netz verwendet wird, bei dem sich die Teilnehmer an- und abmelden können und wo auch die an die

Liste geschickten Nachrichten zusätzlich archiviert werden. Einen Schritt weiter ging das bereits in den achtziger Jahren entstandene *USENET*, ein System thematisch geordneter Diskussionsgruppen, *Newsgroups* genannt, die dezentral realisiert waren: Die von den Benutzern geschriebenen Beiträge, auch *Artikel* oder *Postings* genannt, wurden automatisch zwischen allen am USENET beteiligten Rechnern repliziert und konnten so mit vergleichsweise geringem Aufwand überall abgerufen werden. Das USENET, obwohl es nach wie vor existiert, wurde inzwischen weitgehend von den web-basierten *Diskussionsforen* abgelöst, bei denen die Benutzer ihren gewöhnlichen Web-Browser verwenden, um ihre Beiträge auf einem zentralen Server abzulegen. Eine Verallgemeinerung dieses Prinzips sind die *Weblogs* bzw. das *Blogging*, wo jeder Teilnehmer sozusagen über sein eigenes, web-basiertes Diskussionsforum verfügt, das er als öffentliches Tagebuch oder Notizbuch verwenden kann. Über ein Kommentar- und Referenzierungssystem sind die *Blogs* wiederum zu einer übergreifenden Struktur verbunden. Eine andere, sich inzwischen etablierende Form der Kommunikation sind die *Wiki-Wiki-Webs*, von denen noch ausführlich die Rede sein wird.

Diese Mechanismen dienen keineswegs nur der Befriedigung diffuser kommunikativer Grundbedürfnisse, also einer Art globalem Geplapper. Sie erweisen sich vielmehr als höchst effizientes Mittel, um Gleichgesinnte zusammenzubringen, Wissen anzusammeln und zu archivieren oder konkrete soziale, politische und kognitive Aufgaben zu bewältigen.

Kasparov gegen den Rest der Welt

Ein eindrückliches Beispiel für die kognitiven Fähigkeiten, die aus der Vernetzung entstehen können, ließ sich im Jahr 1999 beobachten, als der damalige Schachweltmeister Garri Kasparov gegen »den Rest der Welt« antrat. Die Partie wurde von Microsoft ausgerichtet. Auf einem öffentlichen Brett machte Kasparov alle 48 Stunden einen Zug, und in der dazwischenliegenden Zeit hatte jeder Internet-Benutzer die Möglichkeit, an einer Abstimmung über den Gegenzug teilzunehmen.[54]

Nun bringt Demokratie im Schach nicht viel, und üblicherweise sind solche Veranstaltungen eine sichere Bank für den Alleinspieler. Die Mehrheitsentscheidung der Gegenspieler führt bestenfalls zu durchschnittlichen Zügen, zumal wenn, wie in diesem Fall, Spieler jeder Spielstärke teilnehmen können. Microsoft bemühte sich daher, diesen Effekt etwas abzumildern, indem man vier junge Schachmeister einlud, die auf der Webseite die Züge Kasparovs analysieren und dem abstimmenden »Rest der Welt« Empfehlungen geben sollten. Auch damit freilich versprach das Spiel kaum mehr als ein PR-Gag zu werden.

Es kam anders. Die damals 15-jährige Irina Krush, die unter den vier eingeladenen Juniormeistern war, erkannte und nutzte als einzige der vier die Möglichkeiten des Internet. Es gelang ihr, die Keimzelle eines sich in Windeseile organisierenden Netzwerks aus Großmeistern, Schachvereinen und dutzenden von talentierten Einzelspielern zu bilden und ihre Empfehlungen aus der »Rechenleistung« dieses Netzwerks zu gewinnen. Die Mehrzahl der abstimmenden Teilnehmer – es waren im Schnitt

sechs- bis siebentausend pro Zug – begriff schnell, dass Irina Krush das eigentliche Sprachrohr des Weltteams war, und stimmte fast durchgehend für die von ihr empfohlenen Züge. Die Partie erreichte daraufhin ein völlig unerwartetes Niveau. Nach etwa fünfzig Zügen schien das Weltteam ein Remis gegen Kasparov erreicht zu haben, was einer Sensation gleichkam. Kasparov erklärte später, dass er sich noch nie derart intensiv mit einer einzelnen Partie beschäftigt hatte, und viele Kommentatoren waren sich einig, dass die Partie einen Platz in der Schachgeschichte verdient hatte. Das Spiel war auch keineswegs in eine Schlacht der Elektronengehirne ausgeartet, obwohl das von vielen vorausgesagt worden war. Zwar wurden tatsächlich ständig Computerprogramme eingesetzt, um Varianten zu analysieren und auf Schwachstellen zu untersuchen, aber alle wesentlichen Entscheidungen des Spiels beruhten letztlich auf der Kreativität und dem Sachverstand der Beteiligten.

Dass Kasparov die Partie schließlich dann doch gewann, lag vermutlich an technischen Unregelmäßigkeiten und Vandalismus. Ab etwa dem fünfzigsten Zug gelang es einzelnen Teilnehmern, mehrfach für bestimmte Züge zu stimmen. (Microsoft hatte dagegen nie besondere Vorkehrungen getroffen, das Problem war lediglich vorher nicht aufgetreten.) In der Folge wurden Irina Krushs Empfehlungen plötzlich bei einigen kritischen Zügen nicht mehr befolgt, so dass sich die Stellung des Weltteams verschlechterte. Im 58. Zug schließlich verzögerte sich eine entscheidende E-Mail von Irina Krush an Microsoft um zehn Stunden, so dass ihre Zugempfehlung nicht veröffentlicht wurde. Die Mehrheit stimmte daraufhin für

einen Zug, den das Weltteam-Netzwerk bereits als Verlust erkannt hatte. Die anderen Kommentatoren hatten ihn jedoch empfohlen.[55]

Der Fall CDDB

Ein anderes Kooperationsprojekt, das allerdings eine unerfreuliche und darum lehrreiche Wendung nahm, war das Projekt einer Datenbank für CD-Titel, *CDDB*.

Zum Hintergrund: Audio-CDs enthalten normalerweise keine Informationen darüber, wie die auf einer Disc gespeicherten Titel heißen oder wer der Interpret ist. Diese Information ist beim Abspielen in einem einfachen CD-Player entbehrlich, da man meistens ohnehin das CD-Cover zur Hand hat und die Titelinformation nachschauen kann. Diese Meta-Daten werden aber sehr wichtig, wenn man die Musikstücke in Form einzelner Dateien zwischen unterschiedlichen Medien hin und her kopiert, wenn man sie auf einem Computer oder MP3-Player anhören oder über ein Filesharing-Netz austauschen möchte.

Um dieses Problem zu lösen, kamen zwei Programmierer auf eine ebenso einfache wie erstaunliche Idee. Die Anzahl und die Länge der einzelnen Tracks sind durchaus auf einer CD gespeichert, und dieses Inhaltsverzeichnis ergibt einen mit hoher Wahrscheinlichkeit eindeutigen »Fingerabdruck« der CD – man kann also jede jemals erschienene CD sehr leicht und verlässlich identifizieren. Würde man die zu einem Fingerabdruck gehörende Titelliste nun in einer über das Internet erreichbaren Datenbank zur Verfügung stellen, dann könnte die Abspiel-

software in einem Computer sie bei Bedarf einfach abrufen. Wer aber hätte die Ressourcen, um eine Datenbank mit Millionen von CD-Titeln zu erstellen?

Die Lösung lag in dezentraler Kooperation. In das Programm, das die Titelinformation aus der CDDB-Datenbank abrufen konnte, baute man als zusätzliche Funktion ein, dass bei einer unbekannten CD eine Eingabemaske erschien, die den Benutzer aufforderte, die Titelinformation selber einzugeben. Die Daten wurden dann an die zentrale Datenbank übermittelt und standen von diesem Moment an allen Benutzern zur Verfügung. Die Idee war so simpel, wie das Mitmachen offenbar Spaß machte, denn die Datenbank füllte sich in Windeseile. Schon nach kurzer Zeit waren fast alle jemals erschienenen CDs katalogisiert, neue Titel tauchten fast augenblicklich nach ihrem Erscheinen in der Datenbank auf.

Dann begann der unerfreuliche Teil. Die Entwickler des CDDB-Systems gründeten eine Firma unter dem Namen *Gracenote* und schickten sich an, die von einem Heer von Freiwilligen erstellte Datenbank kommerziell zu vermarkten.[56] Entwickler von Abspielsoftware, die CDDB benutzen wollten, mussten von nun an Lizenzgebühren an Gracenote zahlen. Die Lizenz verlangte außerdem von ihnen, dass ihre Programme ein CDDB-Logo anzeigen mussten und dass es mit den Programmen nicht möglich sein durfte, eine andere Datenbank als CDDB für die Titelinformation zu benutzen.

Viele der Benutzer, die mitgeholfen hatten, die Datenbank zusammenzutragen, waren darüber empört. Die ursprüngliche CDDB-Software war unter der GPL lizenziert gewesen, darum war man davon ausgegangen, bei

einem freien Projekt mitzuarbeiten. Zwar gab es keine Klausel in der GPL, die das Verhalten von Gracenote verboten hätte, aber dennoch war man sich einig, dass hier nicht nach den richtigen Regeln gespielt wurde.

Die Affäre hatte zwei Folgen. Erstens taten sich schnell einige andere Programmierer zusammen und begannen unter dem Namen *FreeDB* ein neues Projekt, das die ganze Arbeit noch einmal von vorn aufnahm.[57] Die zusammengetragenen Daten wurden diesmal jedoch unter die GNU Free Documentation License (GFDL) gestellt. Jeder Benutzer, der sich beteiligte, konnte also sicher sein, dass sein Beitrag nicht eines Tages in einer proprietären Datenbank enden würde. FreeDB zog, was die Vollständigkeit der Information betraf, in kürzester Zeit mit CDDB gleich. In der Praxis ist es heute so, dass proprietäre Abspielprogramme wie Apples iTunes oder der Windows Media Player die proprietäre CDDB-Datenbank benutzen oder benutzt haben, einschließlich der Zahlung entsprechender Lizenzgebühren, während Freie Programme FreeDB verwenden (auch dann, wenn diese Funktion manchmal in der Software als »CDDB« bezeichnet wird).

Die andere Folge der Affäre war, dass man in der Szene hochsensibel für das zugrundeliegende Problem geworden war und dass viele Projekte, zum Beispiel auch die im folgenden Abschnitt beschriebene Enzyklopädie Wikipedia, seither sehr genau darauf achten, den rechtlichten Status der kollektiv zusammengetragenen Information über eine freie Lizenz (in der Regel die GFDL) abzusichern.

Neuere Abspielprogramme benutzen statt CDDB oder FreeDB inzwischen oft den Dienst des freien Projekts *MusicBrainz*, wo digitale »Fingerabdrücke« der Musik-

stücke selbst verwendet werden, um sie zu identifizieren und zu katalogisieren, auch dann, wenn sie nicht mehr direkt von einer Audio-CD stammen.[58]

Wikimanie

Die wohl spektakulärste Erfolgsgeschichte der freien Kooperation im Internet ist heute die Online-Enzyklopädie *Wikipedia*.[59] Begründet im Jahr 2001, umfasst sie allein in der englischen Ausgabe inzwischen über eine Million Artikel und zählt zu den dreißig meistgenutzten Adressen im World Wide Web. Dennoch ruft das Prinzip von Wikipedia oft ungläubiges Staunen hervor oder auch scharfe Kritik.

Jimmy Wales, der Gründer von Wikipedia, spielte seit den späten neunziger Jahren mit dem Gedanken einer Online-Enzyklopädie, die ähnlich funktionieren sollte wie die Projekte der Freien Software Szene. Auch Richard Stallman hatte im Jahr 1999 einen entsprechenden Aufruf veröffentlicht, die Idee lag also in der Luft.[60]

Im März 2000 begann Wales, unterstützt durch den Programmierer Larry Sanger, das Projekt *Nupedia.*[61] Es sollte ein Lexikon hervorbringen, das jedem frei zur Verfügung stehen würde und an dem prinzipiell jeder mitarbeiten konnte. Für das Schreiben der Artikel sollten jedoch vor allem Experten der jeweiligen Gebiete gewonnen werden, und »Experte«, das hieß: wenn möglich mit einem entsprechenden Doktortitel. Um die Qualität der Beiträge zu sichern, wurde ein formaler Redaktionsprozess definiert, den jeder Artikel durchlaufen sollte. Der Prozess ähnelte der Arbeitsweise etablierter Lexikaverlage und umfasste sieben

einzelne Schritte, die über mehrere inhaltliche Reviews bis zur orthographischen Durchsicht und Formatierung reichten. Auch an dieser Redaktionsarbeit sollte prinzipiell jeder teilnehmen können, genau wie bei den Programmierern eines Freien Software Projekts. Etablierte Lexika, allen voran die *Encyclopædia Britannica*, empfand man dabei als den großen Goliath: Wenn man die vernetzten Ressourcen des Internet nur geschickt ausnutzte, würde man diesem Goliath vielleicht sogar ebenbürtig sein können.

Die Realität war freilich ernüchternd. Obwohl sich einige begeisterte Mitstreiter einfanden, wurden im Laufe des ersten Jahres kaum mehr als ein Dutzend Artikel fertig gestellt. Zwar waren diese wirklich von außerordentlicher Qualität, aber eine Enzyklopädie war so in absehbarer Zeit nicht in Sicht. Wales und Sanger suchten darum nach einer effektiveren Methode. Sie gerieten dabei an Ward Cunninghams Idee eines *Wiki-Wiki-Webs* (der Ausdruck »Wiki-Wiki« stammt aus dem Hawaiianischen und bedeutet »Schnell-schnell«): Man versteht darunter einen Website, bei dem sich auf jeder Seite, für jeden Benutzer ein Knopf befindet mit der Aufschrift: »Diese Seite jetzt ändern.«

Trotz großer Bedenken und Widerstände im Nupedia-Projekt beschlossen Wales und Sanger, die Idee auf die Online-Enzyklopädie anzuwenden. Am 15. Januar 2001 ging Wikipedia online – und der Rest ist Internet-Geschichte. Innerhalb des ersten Jahres entstanden allein in der englischen Version 18.000 Artikel, nach dem zweiten waren es 100.000, im Frühjahr 2005 wurden 500.000 Artikel erreicht und am 1. März 2006 die Millionenmarke überschritten. (Zum Vergleich: Die Encyclopædia Britannica enthält, je nach Ausgabe, etwa 120.000 Artikel.)

Es ergibt sich eine Fülle von Fragen: Wie hoch ist die Qualität der Wikipedia-Artikel und wer garantiert dafür? Wenn jeder jede Seite sofort ändern kann, gibt es dann keinen Vandalismus oder, schlimmer noch: gezielte und subtile Desinformation? Und nicht zuletzt: Wer bezahlt für das alles?

Die Qualität der Wikipedia-Artikel variiert selbstverständlich, ist aber im Allgemeinen erstaunlich hoch. Die meisten Artikel beginnen als kurze Absätze oder Stoffsammlungen und werden im Laufe der Zeit immer weiter verfeinert; die besten brauchen keinen Vergleich mit einer klassischen Enzyklopädie zu scheuen. Die faktische Korrektheit ist ebenfalls hoch. Im Dezember 2005 verglich die britische Fachzeitschrift *nature*, eines der weltweit angesehensten wissenschaftlichen Journale, zweiundvierzig zufällig ausgewählte, naturwissenschaftliche Artikel sowohl in Wikipedia als auch der Encyclopædia Britannica. Die Qualität erwies sich als sehr vergleichbar: Britannica enthielt im Schnitt drei faktische Fehler pro Artikel, Wikipedia vier.[62]

Wie ist das möglich? Zunächst einmal zeigt die Studie, dass keines der beiden Werke den Nimbus einer unfehlbaren Referenz beanspruchen kann – auch die Encyclopædia Britannica nicht. Die Studie erinnert vielmehr daran, dass ein Lexikon-Artikel für sich allein niemals ausreicht, eine Aussage wissenschaftlich-fundiert abzusichern (was freilich unter Wissenschaftlern auch nie umstritten war). Die Arbeit fest angestellter und bezahlter Lexikon-Redakteure auf der einen Seite und der kollektive Gradient dessen, was entsteht, wenn zahllose weitgehend anonyme Freiwillige jeweils ein paar Sätze über ein Thema schreiben, mit dem

sie sich gut auskennen, führt aber offenbar zu sehr vergleichbaren Ergebnissen.

Bei der Encyclopædia Britannica sah man das allerdings anders. Kurz nach Erscheinen der *nature*-Studie schaltete die Britannica-Redaktion eine halbseitige Anzeige in der Londoner *Times* und veröffentlichte eine Gegendarstellung, in der sie die Ergebnisse in Frage stellte.[63] Es sei schlicht lächerlich, anzunehmen, ein Heer von Freiwilligen könne auch nur in die Nähe der Qualität der ältesten Enzyklopädie des englischen Sprachraums kommen. Die Studie müsse methodische Fehler aufweisen. Die Redaktion von *nature*, selber einer wissenschaftlichen Arbeitsweise und höchsten Standards verpflichtet, wies diese Anschuldigungen zurück.[64]

Tatsächlich aber greift der bloße Vergleich der Artikelqualität eigentlich zu kurz. Wikipedia enthält bereits jetzt zehnmal *mehr* Artikel als die Encyclopædia Britannica, was für die enorme Breite des abgedeckten Wissens spricht. Nicht nur jedes Land der Erde oder jedes bekannte Antibiotikum ist verzeichnet, sondern ebenso hat jede Musikband, jede Comic- oder Fernsehserie ihren eigenen Eintrag, meist akribisch zusammengetragen von Fans mit einschlägigem Expertenwissen. Jedes Spiel der Fussballweltmeisterschaft ist dokumentiert, einschließlich prozentualem Ballbesitz und den Namen der Torschützen. Bei politischen Krisen oder Naturkatastrophen werden in Windeseile entsprechende Seiten angelegt, die fast zeitgleich mit der Entwicklung der Ereignisse aktualisiert werden. Eine klassische Enzyklopädie kann angesichts dieser Breite und Aktualität nicht mithalten. Bezeichnend ist hierzu ein Vergleich, den kürzlich ein Forumsteilnehmer vorgeschla-

gen hat: Man möge sich doch einmal anschauen, was in Wikipedia über die Encyclopædia Britannica steht, und was in der Encyclopædia Britannica über Wikipedia. In einem der beiden Fälle lautet die Antwort: gar nichts.

Dass die grosse Breite und die Aktualität dennoch nicht zu Lasten der Qualität gehen, erklärt sich unter anderem dadurch, dass die Artikel keineswegs einfach drauflos geschrieben werden. Mit der Zeit hat sich – wiederum durch Selbstorganisation und wiederum in Form eigener Wikipedia-Seiten – ein umfangreiches System von Empfehlungen und Richtlinien herausgebildet, die beim Schreiben der Artikel als Messlatte und Korrektiv dienen. Zu den wichtigsten dieser Richtlinien gehört das Prinzip des *Neutral Point of View (NPOV)*, also der wertungsfreien Darstellung. Gerade bei kontroversen Themen gelingt eine solche wertungsfreie Perspektive oft erst nach langem und zähem Ringen, und die entsprechenden Artikel bekommen während dieser Phase – wiederum durch eine gewöhnliche Änderung, die von jedem Benutzer gemacht werden kann – eine spezielle Markierung, die besagt: »Die Neutralität dieses Artikels wird in Frage gestellt«. Besucher werden dann auf die dem Artikel zugeordnete Diskussionsseite verwiesen, wo die verschiedenen Ansichten meist kontrovers diskutiert werden. In der Praxis zeigt sich, dass solche Artikel und ihre Diskussionsseiten oft einen hervorragenden Einblick in das Problemfeld eines bestimmten Themas geben, auch und gerade dann, wenn die Neutralität des Artikels noch nicht erreicht ist.

Mitunter werden aus den Kontroversen aber auch echte Streitigkeiten. Es kann dann vorkommen, dass zwei oder mehr Benutzer wechselseitig die Änderungen des jeweils

anderen umschreiben oder rückgängig machen. Diese sehr unerfreuliche und darum allseits gefürchtete Eskalation wird als »edit war«, also »Änderungskrieg« bezeichnet. Kommt es zu keiner Einigung, dann tritt – wiederum selbstorganisiert – eine Reihe von Mechanismen in Kraft, die den Konflikt beilegen soll. Zunächst versucht ein unbeteiligter Dritter, die Kontroverse auf die Diskussionsseite des Artikels zurückzuführen und zwischen den Parteien zu vermitteln. Sollte das zu keinem Erfolg führen, dann kann eine Art Schiedsstelle, das *Arbitration Committee,* angerufen werden, dessen Mitglieder regelmäßig durch öffentliche Abstimmung gewählt werden. Dieses Komitee kann verbindlich entscheiden, wie mit der Kontroverse verfahren werden soll. Als äußerstes Mittel können Benutzer von Wikipedia ausgeschlossen werden – zur Not auch über ihre IP-Adresse, falls ein Benutzer unter anderem Namen den Streit weiterzutreiben versucht. (In der Praxis kommt das extrem selten vor, es gibt unter vielen tausend Benutzern nur eine Hand voll notorischer Fälle.)

Reiner, ungerichteter Vandalismus ist demgegenüber ein eher gängiges Problem. Bei einem durchschnittlichen Artikel kommt es zur Zeit etwa einmal pro Woche vor, dass Obszönitäten oder sonstiger Unfug eingefügt oder Teile des Artikels unmotiviert gelöscht werden. Wikipedia kann diesem Problem allerdings im Rahmen der eigenen Voraussetzungen sehr gut begegnen: Meistens sind solche destruktiven Änderungen schon nach wenigen Minuten wieder behoben.

Möglich ist das, weil jede Änderung, die ein Benutzer vornimmt, intern als Differenz zum vorherigen Stand des Artikels gespeichert wird. Es ist also sehr leicht möglich,

auf die ursprüngliche Version des Artikels zurückzugehen, sollte sich eine Änderung als Vandalismus herausstellen. Das wiederum ist in der Praxis meist leicht zu erkennen, da Vandalismus sich in der Regel nicht durch hohe Kreativität auszeichnet. Auf einer besonderen Seite wird ein ständig mitlaufendes Protokoll aller Änderungen angezeigt, die sogenannte Liste der *Recent Changes (RC)*. Geübte Wikipedianer erkennen Vandalismus darin meistens auf einen Blick und stellen die ursprüngliche Version des Artikels sofort wieder her. Tatsächlich hat sich von selbst eine Gruppe von Benutzern herausgebildet, die sogenannte »RC Patrouille«, die sich solche Aufräumarbeiten zur besonderen Aufgabe gemacht hat. Jimmy Wales schätzt, dass ein harter Kern von etwa 600-1000 Benutzern auf diese und andere Weise dafür sorgt, dass die einmal erreichte Qualität der Beiträge erhalten bleibt.

Zwar hat sich durch die rapide steigende Popularität von Wikipedia auch das Vandalismus-Problem verschärft, gleichzeitig aber entstehen in der selbstorganisierten Gemeinschaft immer neue Techniken, dagegen vorzugehen. So gehen Änderungen inzwischen schneller als im Sekundentakt ein, und es ist nicht mehr möglich, die RC-Liste manuell zu verfolgen. Zur Abhilfe wurden spezielle *Bots* geschrieben, also Programme, die automatisch nach bestimmten Auffälligkeiten in der Liste suchen und diese in einem eigens dafür eingerichteten *Chatroom* anzeigen. Dort wiederum halten sich zu jeder Tages- und Nachtzeit einige Wiki-Sheriffs auf, um bei einem Alarm sofort aktiv zu werden.

Bei einigen besonders prominenten oder kontroversen Artikeln war wiederum auch das nicht genug: Hier nahm der Vandalismus so sehr überhand, dass an normales Ar-

beiten nicht mehr zu denken war. (Im Wesentlichen waren das die Artikel über *George W. Bush*, *Jesus Christus* und *Adolf Hitler*.) Bei diesen wenigen Artikeln ging man darum noch einen Schritt weiter und ließ nur noch Änderungen durch solche Benutzer zu, die mindestens seit vier Tagen bei Wikipedia registriert waren. (Um sich zu registrieren genügt eine E-Mail-Adresse; bei anderen Artikeln können auch anonyme Benutzer ohne Anmeldung Änderungen vornehmen.) Ähnliches geschieht inzwischen bei Artikeln, die kurzzeitig hohe Sichtbarkeit erreichen, zum Beispiel weil sie von einem anderen prominenten Ort im Internet gelinkt werden. Kurzfristig, meist nur für einige Stunden, werden dann alle Änderungen gesperrt, um den unvermeidlichen Vandalismus-Sturm, der mit solcher Publicity einherginge, abzufangen (besonders die zu einem Artikel gehörenden Bilder waren sonst ein beliebtes Ziel und wurden fast augenblicklich durch Pornographie ersetzt). Auf diese Weise ließ sich das Vandalismus-Problem eindämmen, ohne gleichzeitig das Konstruktionsprinzip von Wikipedia, nämlich die absolute Offenheit und das Fehlen jeder »Schwelle zum Mitmachen« aufzugeben.

Von weit geringerem Ausmaß, aber auch schwieriger zu behandeln ist das Problem gezielter Sabotage und subtiler Desinformation. Aufsehen erregte im Herbst 2005 eine Kontroverse um John Seigenthaler Sr., einen bekannten US-Journalisten und ehemaligen Berater von Robert F. Kennedy. Seigenthaler entdeckte im September 2005, dass der Wikipedia-Eintrag über ihn die hanebüchene Mutmaßung enthielt, er sei in die Ermordung sowohl John F. Kennedys als auch Robert Kennedys verwickelt gewesen. Er beschwerte sich umgehend bei Wikipedia und die üble

Nachrede wurde sofort entfernt. In einem ungewöhnlichen Schritt ging Jimmy Wales sogar so weit, die Änderungshistorie des Artikels zu sperren, so dass die falsche Information öffentlich nicht mehr zugänglich war (nicht in Wikipedia jedenfalls – andere, unabhängige Websites hatten die Aussage bereits aus Wikipedia übernommen und behielten sie noch einige Zeit bei).

Seigenthalers Empörung war damit aber keineswegs besänftigt, und so schrieb er in zwei großen amerikanischen Zeitungen Leitartikel, in denen er vor Wikipedia warnte und die Enzyklopädie als ein »fehlerhaftes und unverantwortliches Forschungsinstrument« bezeichnete.[65] Dies wiederum führte zu einer Welle von negativer Berichterstattung über Wikipedia in anderen Medien. Ein Redakteur der *New York Times* bat seine Mitarbeiter, Wikipedia nicht mehr zur Überprüfung von Fakten heranzuziehen, die in der Zeitung erscheinen sollen.[66]

Befürworter Wikipedias halten die ganze Affäre, sowie auch die Kritik, die sich in ihrem Gefolge ergab, für ein fundamentales Missverständnis der Natur der Sache. Sie weisen darauf hin, dass Wikipedia, wie auch jede andere Enzyklopädie, nur die erste und nicht etwa die letzte Quelle der Information in einer strittigen Frage sein kann – dass eine Enzyklopädie mit anderen Worten ein Mittel zur Exploration von Wissen, nicht aber zu dessen hieb- und stichfester Absicherung ist. (Gleichzeitig legen die Stil-Richtlinien von Wikipedia nahe, die Artikel durch wissenschaftlich korrekt zitierte Referenzen auch in dieser Hinsicht zu verbessern.) Eklatante Rechtsbrüche wie zum Beispiel üble Nachrede könnten darüber hinaus durchaus verfolgt und geahndet werden, wie auch der Fall Seigen-

thaler zeigte: Schon kurze Zeit nach dem Bekanntwerden der Affäre gelang es einem unabhängigen Wikipedia-Kritiker, den Täter durch Rückverfolgung der IP-Adresse ausfindig zu machen. (Eine offizielle Nachforschung durch Seigenthalers Anwälte beim fraglichen Internet-Provider war im Sande verlaufen.) Der Täter, der die Sache eher als Scherz verstanden hatte und sich von den Ausmaßen der Affäre selber erschreckt zeigte, entschuldigte sich sehr kleinlaut bei Seigenthaler, der daraufhin von einem Prozess absah.[67]

Mit ein bisschen mehr Augenmaß hätte Seigenthaler das vielleicht auch einfacher haben können. »Wenn's ihm nicht gefiel, warum hat er's dann nicht einfach geändert?« war ein oft zu hörender Kommentar in den einschlägigen Diskussionsforen, sowie auch die Anmerkung, dass man *jeden* Text, also auch Wikipedia, am besten unter Zuhilfenahme des eigenen kritischen Verstandes lesen sollte. Seigenthaler selbst zeigte sich am Ende der Affäre ebenfalls nachdenklich: Es sei ein sehr merkwürdiges Gefühl für ihn, der immer für das Recht auf freie Meinungsäußerung eingetreten sei, nun jemand wegen eben dieses Rechts zu belangen.

»Ich glaube immer noch an die freie Meinungsäußerung. Aber was ich verlange, ist, dass man herausfinden kann, wer für eine Äußerung verantwortlich ist.«

Jimmy Wales räumte mögliche Schwächen in diesem Bereich ein und stellte in Aussicht, dass in Zukunft mehr Informationen darüber bereitgestellt würden, wer welche Änderung gemacht habe.

Durch die steigende Popularität von Wikipedia wird auch ein anderes Problem allmählich virulent, nämlich das

der Finanzierung. Bisher wurde das Projekt ausschließlich durch Spenden getragen, was angesichts bescheidener Bedürfnisse auch gut funktionierte: Wikipedia verfügt über genau zwei bezahlte Mitarbeiter (Jimmy Wales gehört nicht zu ihnen), und darüber hinaus muss nur die technische Infrastruktur bezahlt werden, also die Serverfarm, die sich in St. Petersburg, Florida befindet. Diese allerdings wird, seit Wikipedia zu den dreißig meistbesuchten Sites im World Wide Web zählt, immer anspruchsvoller, wohingegen das Spendenaufkommen nicht in gleichem Maße zugenommen hat.

Vielleicht zeigt sich hier ein Effekt, der in der Ökonomie und Philosophie unter dem Namen »Tragedy of the Commons« bekannt ist. Man versteht darunter die Beobachtung, dass viele Systeme, bei denen eine Ressource von der Allgemeinheit auf freiwilliger Basis benutzt und unterhalten wird, früher oder später zusammenbrechen, weil es immer zu viele »Trittbrettfahrer« gibt, also Leute, die zwar gerne die Ressource in Anspruch nehmen, aber selber nichts zu ihrem Unterhalt beitragen. Es könnte also sein, dass Wikipedia in dem Moment, in dem es von einem Projekt von Enthusiasten zu einem allgemeinen Massenphänomen wird, sich nicht mehr selber zu tragen imstande ist. Man wird dann möglicherweise auf Werbefinanzierung oder Partnerschaft mit großen Unternehmen zurückgreifen müssen, obwohl Jimmy Wales das bislang kategorisch ausschließt. Der Effekt ist freilich nicht zwangsläufig, sondern nur eine statistische Generalisierung, die auf eine gegebene Situation zutreffen kann oder auch nicht. Dass die Gesetzmäßigkeit nicht unbeschränkt gilt, zeigen andere Projekte der digitalen Welt wie zum Beispiel

GNU/Linux, die keinerlei Anzeichen verraten, unter der »Tragedy of the Commons« zusammenzubrechen.

Wikipedia wächst derweil unaufhaltsam weiter. Inzwischen existieren Teilprojekte in über 200 verschiedenen Sprachen, von denen die kleinsten Wikipedias nur einige dutzend bis hundert Artikel enthalten. Aber auch sie wachsen.

»Stellt euch eine Welt vor, in der jede einzelne Person freien Zugang zu der Summe allen menschlichen Wissens hat. Das ist es, was wir machen.«[68]

So benennt Jimmy Wales die Maxime des Projekts. »Jede einzelne Person«, das bedeute auch, dass jeder dieses Wissen in seiner eigenen Sprache vorfinden könne.

»Obwohl noch viel zu tun ist, kann man denen, die Englisch, Deutsch, Französisch oder Japanisch sprechen und über einen Breitband-Internet-Zugang verfügen, bereits sagen: Ihr habt eure Enzyklopädie.«[69]

Jede dieser Sprachen sei über die kritische Grenze von 100.000 Artikeln hinaus, andere würden in den nächsten Jahren folgen. Das Ziel sei erreicht, wenn es Wikipedias mit mindestens 250.000 Artikeln für jede Sprache mit mindestens einer Million Sprechern gäbe und gleichzeitig ernsthafte Projekte auch für sehr kleine Sprachen vorlägen. Jimmy Wales geht davon aus, dass dieses Ziel in fünfzehn Jahren erreicht werden kann.

Die Befreiung der Information

Alle in diesem Buch beschriebenen Projekte, Subkulturen und Bewegungen haben Gemeinsamkeiten. Sie sind Ausdruck einer übergreifenden technologischen und gesellschaftlichen Entwicklung, in der sich drei Faktoren ausmachen lassen.

Faktor: Kopie

Der erste Faktor besteht darin, dass es extrem einfach geworden ist, Information zu kopieren und zu vertreiben. Daten gleich welcher Art zu kopieren ist eine der Grundoperationen eines Computers, die im normalen Betrieb ständig erfolgt: Um ein Programm auszuführen, muss es von der Festplatte in den Arbeitsspeicher kopiert werden, um ein Musikstück anzuhören, müssen die Bits von der CD ebenfalls in den Arbeitsspeicher und von dort in die Soundkarte kopiert werden. Zu dieser Einfachheit des Kopierens tritt nun die Tatsache, dass es für Daten gleich welcher Art inzwischen ein planetares Informationsnetz gibt, dessen Kosten gesamtgesellschaftlich amortisiert und nicht auf die einzelnen Datentransfers umgelegt werden. In der Folge ist eine Situation entstanden, die es so historisch noch nie gegeben hat: Eine Situation, in der prinzipiell jede Information prinzipiell jedem Erdenbürger augenblicklich zur Verfügung stehen kann, ohne dass direkte Kosten entstehen.

So neu und ohne Beispiel diese Situation auch sein mag, gehen die Menschen doch äußerst gelassen und selbstverständlich mit ihr um. Da praktisch keine Kosten damit verbunden sind, gehört es heute zum guten Ton, einander an den Bits, über die man verfügt, teilhaben zu lassen. Wer die Bitte »Machst du mir mal 'ne Kopie davon?« mit dem Hinweis auf die Rechtslage ablehnen würde, dürfte für ziemlich wunderlich gehalten werden, wenn er nicht sogar riskiert, einen Freund zu verlieren. Den Zugriff auf eine Ressource zu verweigern, mit der keine erkennbaren Kosten verbunden sind, scheint sehr grundlegenden menschlichen Instinkten zu widersprechen.

Die Industrie hat das durchaus erkannt und versucht darum auch nach Kräften, die Menschen *umzuerziehen*. In Kinospots werden unbescholtene Familienväter in Handschellen abgeführt, um die Ansicht »Raubkopierer sind Verbrecher« durchzusetzen. »Gedankendiebe« war das Thema eines bizarren Filmwettbewerbs, zu dem die Firma Microsoft Teenager in Großbritannien einlud.[70] Ob es den Konzernen gelingen wird, sich auf diese Weise neue Menschen, d.h. gefügige Kunden zu schaffen, ist fraglich.

Dabei sind mit den Bits natürlich durchaus Kosten verbunden, nämlich die ihrer Erschaffung durch die Künstler, Programmierer, Journalisten, Wissenschaftler etc. Da die Herstellung und das Aneignen einer Kopie jedoch derart einfach geworden sind, werden diese Vorgänge bald nicht mehr die Basis für die Entlohnung der Bitschaffenden sein können – es sei denn, man würde die Kopiervorgänge und Datenwege derart stark reglementieren und die Reglementierung derart drakonisch überwachen, dass es die technische Entwicklung *ad absurdum* führen würde.

Richard Stallman verglich diese Situation mit einer Raumstation, auf der für die Bewohner Sauerstoff produziert werden muss. Eine Möglichkeit, die Sauerstoffproduktion gerecht zu finanzieren, bestünde darin, den Bewohnern Gasmasken mit Volumenzählern aufzusetzen und jeden nach der Menge Sauerstoff, die er einatmet, bezahlen zu lassen. Das wäre freilich eine bizarre Gesellschaft, und sie würde zudem auch nur funktionieren können, wenn es eine Geheimpolizei gäbe, die überwacht, dass niemand etwa die Gasmaske abnimmt und »einfach so« atmet. Sinnvoller und fast genauso gerecht wäre zum Beispiel eine Sauerstoffsteuer, mit der die Kosten auf die Bewohner umgelegt werden und die ihnen so die Gasmasken erspart.

Faktor: Kommunikation und Kooperation

Der zweite gemeinsame Faktor aller hier beschriebenen Entwicklungen ist, dass sie auf hochgradig dezentraler, selbstorganisierter Kommunikation zwischen Individuen beruhen. Es ist diese stetige und durch die Natur des Internet praktisch kostenlose Kommunikation in Mailinglisten, Chaträumen und Diskussionsforen, die erst die Kooperation ermöglicht, ohne die Projekte wie GNU/Linux oder Wikipedia nicht denkbar wären.

Das Internet hat nicht nur die Kommunikationsinfrastruktur des Planeten auf eine neue Grundlage gestellt, es verändert auch die *Wege* der Kommunikation. Am Ende des 20. Jahrhunderts bestanden diese Wege vor allem aus 1:1-Beziehungen (individuelle Kommunikation per Brief

oder Telefon), oder aus 1:n-Beziehungen (den klassischen Massenmedien, bei denen wenige, staatlich zugelassene »Sender«, d.h. Fernsehstationen, Zeitungen oder Radiosender die »Vielen«, d.h. die allgemeine Bevölkerung mit Informationen versorgen). Das Internet bringt demgegenüber zunehmend n:n-Beziehungen hervor, bei denen prinzipiell jeder zum Informationslieferanten für viele werden kann. Beispiele dafür sind Nachrichtendienste wie *Slashdot*, *Kuro5hin* oder *Digg*, bei denen die Leser selbst die Beiträge einreichen. Sie werden dann von einer Redaktion gesichtet und ausgewählt, oder aber die Leser entscheiden selbst per Abstimmung über die Publikation. Das Projekt *WikiNews*, ein Wikipedia-Schwestersite, versucht, das Wiki-Wiki-Prinzip auf klassische Nachrichten anzuwenden. Rapide an Bedeutung gewinnt auch die Szene der *Weblogs* bzw. der *Blogger*, d.h. der offenen Tagebücher im Netz, die von den Lesern kommentiert und verlinkt werden können. In der Zukunft wird man Informationen über ein Ereignis vielleicht vor allem dadurch einholen können, dass man sich anschaut, was die Augenzeugen vor Ort in ihren Weblogs zu sagen haben – ein Phänomen, für das sich der Ausdruck *Bürgerjournalismus* zu etablieren beginnt.

Im Umfeld der Free Software Foundation denkt man noch weiter. Schon heute ist der Preis, den der Einzelne für die Anbindung an das weltweite Kommunikationsnetz bezahlen muß, zumindest in den westlichen Gesellschaften sehr stark gesunken. Er könnte aber buchstäblich bei *null* liegen, wenn man die Technik der drahtlosen Vernetzung, heute bekannt unter den Namen *WiFi* bzw. *WLAN*, konsequent ausnutzen würde. Das Netz könnte dann weitge-

hend ohne physische Infrastruktur, d.h. ohne Kupfer- und Glasfaserkabel im Boden aufgebaut werden. Es würde sich, entsprechende Software vorausgesetzt, vollkommen selbst regulieren können. Schon heute gibt es in den meisten Metropolen der Welt Bürgerinitiativen, deren Mitglieder ihre ohnehin vorhandenen Internet-Zugänge per Funk frei für die Allgemeinheit zur Verfügung stellen. Die Stadt San Francisco plant, noch im Jahr 2006 alle ihre Bürger mit kostenlosem und drahtlosem Internet zu versorgen. Die Gebühren, die man anderswo für die Erlaubnis, einen *Hotspot* zu benutzen, aufbringen muss, wirken vor diesem Hintergrund anachronistisch – sie werden mittelfristig keiner realen Ressource, die »knapp«, d.h. verhandelbar wäre, mehr entsprechen.

Die Schwierigkeit, die sich dieser weltweiten, kostenlosen Vernetzung entgegenstellt, ist die Regulierung des elektromagnetischen Spektrums. Es sind staatliche Stellen, die entscheiden, wer auf welcher Frequenz senden darf, und mit welcher Leistung. Die heutige WLAN-Technologie hat dabei ein sehr kleines und nicht besonders nützliches Frequenzband zugeteilt bekommen, das sich nur für die Kommunikation innerhalb nächster Nähe eignet – ein einzelnes Haus, bestenfalls ein Straßenzug. Unter der Losung »Freies Spektrum« tritt die Free Software Foundation darum für die Aufhebung solcher Regulierungen ein. Eine weltumspannende, jedermann frei zugängliche Kommunikationsinfrastruktur könnte so möglich werden.

Der dritte gemeinsame Faktor der beschriebenen Entwicklungen ist schließlich, dass Dinge, die bisher Geld gekostet haben – möglicherweise *viel* Geld –, der Allgemeinheit plötzlich umsonst oder für sehr viel weniger Geld zur Verfügung stehen. Wie ist das möglich? Wie kann das wirtschaftlich und gesellschaftlich funktionieren?

Es kann nicht genug betont werden, dass keine der erwähnten Bewegungen ihrem Wesen nach anti-kommerziell ist. Die Free Software Foundation tritt nicht dafür ein, dass Software kostenlos sein sollte, sondern sie weist darauf hin, dass der Mechanismus, mit dem häufig die Kosten für Software gedeckt werden, nämlich die Geheimhaltung des Quelltextes und die Einschränkung der Benutzung, der Gesellschaft mehr schadet als nützt. Die Menschen, die sich per Filesharing mit Musik und Filmen versorgen, dürften zu einem großen Teil zustimmen, dass die Künstler für ihre Arbeit bezahlt werden sollen – aber sie sehen gleichzeitig, dass das Herstellen und Vertreiben von Kopien, weil es zu einem Vorgang geworden ist, der so einfach ist wie das Atmen, kein gutes Kriterium mehr dafür abgibt, wer bezahlen muss, wann und wofür.

Das Ziel muss darum sein, andere Mechanismen zu finden, die mit den technischen Realitäten besser in Einklang stehen. Bei diesem *reality check* wird es Verlierer geben. Reine Vertriebsindustrien erweisen sich beispielsweise als technisch überflüssig. Der Versuch, sie um ihrer selbst willen zu erhalten, würde an die »Heizer« erinnern, die auf Druck der Eisenbahnergewerkschaften auf den ersten Diesellokomotiven mitfahren mussten, obwohl es

für sie dort nichts zu tun gab. Auch eine Redaktion für ein Lexikon zu unterhalten, dürfte in Kürze kein praktikables Geschäftsmodell mehr sein, weil sich zeigt, dass die Menschen bereitwillig ihr Wissen selber zusammentragen und sich das Ganze so organisieren lässt, dass die Qualität dabei nicht auf der Strecke bleibt.

Die Tatsache, dass viele Menschen es sich offenbar leisten *können* und auch leisten *wollen*, bei freien, d.h. gemeinnützigen Projekten wie GNU/Linux oder Wikipedia mitzuarbeiten, ohne dabei unmittelbar an Bezahlung zu denken, ist vielleicht der erstaunlichste Aspekt dieser Bewegungen. Es handelt sich dabei keineswegs nur um reiche Philanthropen oder um Studenten, die noch bei Mama wohnen – die meisten von ihnen sind eher in den mittleren Einkommensklassen zu finden und würden mitnichten behaupten, ihre finanzielle Situation sei in irgendeiner Weise »entspannt«. Ihr Engagement dürfte eher unbewusst motiviert sein und nur indirekt darauf hinweisen, dass sie offenbar über die nötigen Ressourcen vor allem an Lebenszeit verfügen, die solch ein Engagement möglich machen. Was dieses Engagement für sie so attraktiv macht, ist wahrscheinlich, dass sie der kapitalistischen Gesellschaft mit ihrer Erwerbsfixierung, ihrer abhängigen Beschäftigung und ihrer entfremdeten Arbeit zumindest punktuell entgehen können: Es macht großen, sehr großen Spaß, ein Problem zu *sehen* und es zu *lösen*, ohne es zu *müssen*.

Folgt man der Argumentation von Eric Raymond (vgl. S. 41), dann ist dies wahrscheinlich mit dem allgemeinen Zuwachs an Wohlstand zu erklären. Es ist gewissermaßen die Dividende der technischen Weiterentwicklung, die sich im 20. Jahrhundert in bislang beispielloser Weise

beschleunigt hat. Kulturkritiker fragen, wo eigentlich die ganze Zeit geblieben ist, die wir durch Erfindungen wie die Waschmaschine und das Interkontinentalflugzeug eingespart haben müssten – und eine Antwort ist möglicherweise, dass diese Zeit heute in Projekte wie GNU/Linux oder Wikipedia wandert, die nicht mehr der direkten wirtschaftlichen Existenzsicherung dienen.

Die bisher erwähnten Projekte sind keine Einzelfälle, keine zufälligen Kuriositäten des Internet-Zeitalters. Man kann heute in den unterschiedlichsten Bereichen ähnliche Vorhaben entstehen sehen; sie alle aufzulisten wäre eine unabschließbare Aufgabe. Stellvertretend und völlig subjektiv ausgewählt seien erwähnt:

– das Projekt *arXiv.org*, in dem Wissenschaftler sich eine Plattform für die Veröffentlichung und den Austausch von Fachartikeln geschaffen haben, um sie schneller und kostengünstiger verbreiten zu können, als es die etablierten akademischen Journale zulassen,

– ein Forum wie *fotocommunity.de*, in dem Amateurfotografen ihre Bilder unter einer Creative Commons Lizenz zusammentragen, um sie zu archivieren, zu diskutieren und zu bewerten,

– die offenen Datenbanken für *Ahnenforschung*, in denen Benutzer die Informationen sammeln und verbinden, die sie aus Kirchenbüchern, Urkunden und sonstigen Quellen gewonnen haben,[71]

– die Emanzipation der Bürger gegenüber der Industrie durch Foren, in denen sie eigene *Testberichte* über Produkte und Dienstleistungen zusammentragen,[72]

– die Idee, Medikamente für *Tropenkrankheiten* auf der Basis von »Open Source« zu entwickeln, um auf diese Weise Krankheiten behandeln zu können, unter denen vor allem arme Teile der Menschheit leiden, an denen die Pharmakonzerne kein Interesse zeigen,[73]

– das *Human Genome Project*, das sich bei der erstmaligen Sequenzierung der vollständigen menschlichen Erbinformation einen Wettlauf mit dem kommerziellen Unternehmen von Craig Venter lieferte, um die gewonnene Information der Allgemeinheit frei zur Verfügung zu stellen und zu verhindern, dass sie von Venter patentiert werden konnte.[74]

Die Idee, die allen diesen Projekten zugrunde liegt, nämlich, Information als ein freies Gut zu betrachten, bedeutet

erstens, dass darüber, wer eine Information herstellen, in die Welt bringen und damit Anerkennung finden kann, nur die Kompetenz entscheiden sollte, nicht aber die Zugehörigkeit zu einem bestimmten Unternehmen, einem bestimmten Land oder einer bestimmten Schicht,

zweitens, dass Information, einmal in die Welt gebracht, jedem frei zur Verfügung stehen kann und soll.

Wie in jeder gesellschaftlichen, politischen Bewegung gibt es auch unter denen, die sich für die Befreiung der Informtion einsetzen, unterschiedliche Ansichten und Konflikte. Die meisten davon lassen sich auf eine einzelne, zentrale Streitfrage zurückführen. Sie besteht darin, ob es sich bei der Befreiung der Information um einen natürlichen Vorgang handelt, der sich aufgrund seiner ihm selbst innewohnenden Vorteile von selbst fortsetzen und durchsetzen wird, oder ob es ein Kampf ist, der gegen feindliche Interessen ausgefochten werden muss.

Diejenigen, die einen natürlichen Vorgang am Werk sehen, orientieren sich in der Regel pragmatisch. Sie sehen kein Problem, mit den heutigen Herren der Information zu kooperieren, und wollen überhaupt alles möglichst entspannt sehen. Ärgerlich werden sie mitunter dann, wenn man, wie sie es nennen, versucht, ihnen vorzuschreiben, was sie tun sollen – dass etwa alle und jede Information, die sie in die Welt bringen, frei sein müsse.

Diejenigen, die in der Entwicklung einen Kampf sehen, tun das unter anderem deshalb, weil sie das Zurückhalten von Information in einer Welt, in der alle Information frei verfügbar sein könnte, für moralisch falsch halten. Sie achten darauf, die Information, die sie selbst in die Welt bringen, vor der Vereinnahmung durch die etablierten Strukturen zu schützen; sie versuchen auch, solche Strukturen zu untergraben, indem sie diese dazu ermuntern oder auch zwingen, ihre bislang proprietäre Information ebenfalls frei verfügbar zu machen.

Welche von diesen beiden Ansichten richtig ist, soll hier nicht entschieden werden. Glücklicherweise spielt das in der täglichen Arbeit auch nur eine untergeordnete Rolle.

Denn eine der bemerkenswerten und neuen Eigenschaften der Idee einer freien Information ist, dass sie trotz ihrer politischen Bedeutung nur sehr wenig mit einem klassischen, politischen Programm zu tun hat, für das geworben und argumentiert werden müsste, bevor es, vielleicht, irgendwann einmal Wirklichkeit werden kann. Befreite Information setzt immer damit ein, dass sie *da* ist, in die Welt gebracht wird, dass jemand den Anfang macht mit einem neuen Projekt, dessen Ergebnisse sofort sichtbar werden und dessen Vorteile gegenüber dem, was es bisher gab, offensichtlich sind.

Kurzfristig wird das bedeuten, dass manche, die bisher mit der Herstellung von Information, bisweilen auch mit deren künstlicher Verknappung, ihr Geld verdient haben, sich nach neuen Erwerbsmöglichkeiten umsehen müssen. Mittelfristig wird sich die Ökonomie um die veränderten Bedingungen des Informationszeitalters herum neu organisieren, so wie sie sich vorher um das Industriezeitalter mit seiner »Rationalisierung« der manuellen Arbeit herum neu organisiert hat. Die Folge war, dass heute die meisten Menschen auf der Welt – ungerechterweise nicht alle – weniger Erwerbsarbeit leisten müssen als noch vor hundert oder zweihundert Jahren, und das bei einem enorm gestiegenen Lebensstandard. Die Befreiung der Information ist die Fortsetzung dieses Prozesses im 21. Jahrhundert. Sie wird die Reibungsverluste aufheben, die heute durch proprietäre, zurückgehaltene Information entstehen, und sie wird dafür sorgen, dass die von der Menschheit als ganzer hervorgebrachte Information schneller, genauer und unterschiedsloser der gesamten

Menschheit zugute kommt. Es wird in der Zukunft immer weniger Gelegenheiten für reine Erwerbsarbeit, und immer mehr für wirkliche Arbeit geben. Die Menschen werden Probleme nicht mehr darum lösen, weil sie damit ihren Lebensunterhalt verdienen müssten, sondern weil diese Probleme wichtig sind, drängend, oder auch faszinierend.

Statistiken

Ein freies Betriebssystem wie GNU/Linux kennt keine Verkaufszahlen, darum ist es nicht so leicht möglich, seinen »Marktanteil« zu bestimmen. Ein möglicher Indikator sind die Besucherzahlen bei bestimmten Websites. Ihre Aussagekraft wird aber durch zwei Faktoren relativiert: Erstens ist die Besuchermenge oft nicht repräsentativ für das Internet schlechthin, zweitens geben nicht alle Web-Browser korrekt an, auf welchem Betriebssystem sie laufen. Abb. 1 zeigt eine solche Statistik, wie sie von w3schools.com erhoben wird.

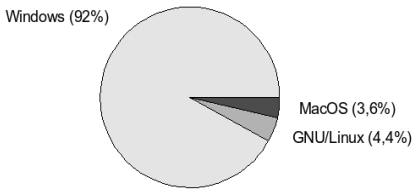


Abb. 1: Relativer Anteil der Betriebssysteme nach w3schools.com, Juni 2006 (Quelle: www.w3schools.com/browsers/browsers_stats.asp)

Das BOINC-Projekt (Berkeley Open Infrastructure for Network Computing) ist der Nachfolger des bekannten SETI@home-Projekts (Search for Extra-Terrestrial Intelligence). Benutzer können ihre Rechner dabei über das Internet an der Lösung wissenschaftlicher Probleme mitarbeiten lassen, zum Beispiel an der Suche nach außerirdischen Funksignalen. Die gezeigte Statistik (Abb. 2) ist der relative Anteil der verschiedenen Betriebssysteme unter den Benutzern von BOINC.

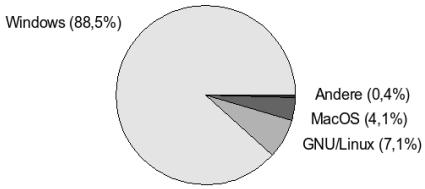


Abb. 2: Relativer Anteil der Betriebssysteme bei BOINC, Juni 2006 (Quelle: www.boincstats.com/stats/project_graph.php?pr=bo&view=hosts)

Land	reg. Anw.	Dichte	Bevölkerung (in Mio.)
Antarctica	5	1215.07	0.004
Finnland	2624	506.76	5.2
Estland	583	429.94	1.4
Dänemark	2202	410.13	5.4
Norwegen	1775	389.43	4.6
Schweden	2505	281.14	8.9
Polen	8176	211.64	38.6
Luxemburg	92	209.20	0.4
Niederlande	3368	207.20	16.3
Schweiz	1339	186.75	7.2
Neuseeland	87	177.06	3.9
Österreich	1428	176.84	8.1
Belgien	1780	173.84	10.2
Spanien	6810	159.41	42.7
Ungarn	1557	154.22	10.1
Bulgarien	1199	152.43	7.9
Kanada	4218	136.00	31.0
Irland	502	130.70	3.8
Deutschland	10718	129.86	82.5
Australien	2316	119.76	19.3
Chile	1719	111.62	15.4
Italien	6089	105.89	57.5
Frankreich	6346	105.44	60.2
Großbritannien	5454	92.66	58.9
Tschechien	895	87.23	10.3
USA	24246	84.80	285.9
Portugal	817	81.42	10.0
Lettland	187	77.72	2.4
Rumänien	1553	71.60	21.7
Litauen	241	65.33	3.7
Slowakei	338	62.55	5.4
Brasilien	9107	52.78	172.6
Griechenland	516	48.57	10.6
Argentinien	1540	41.08	37.5
Russland	2640	18.14	145.5
Ukraine	851	17.90	47.5
Weißrussland	158	15.57	10.1
Türkei	769	11.37	67.6
Iran	242	3.39	71.4
Indien	2918	2.85	1025.1
Japan	244	1.92	127.3
Irak	18	0.76	23.6
China	635	0.49	1285.0
Niger	1	0.09	11.2
Nordkorea	1	0.04	22.4
Kongo (Dem. Rep.)	1	0.02	52.5

Land	reg. Anw.	Dichte	Bevölkerung (in Mio.)
Hamburg	354	204.14	1.7
Bremen	127	191.81	0.67
Berlin	578	170.58	3.4
Hessen	836	137.23	6.1
Baden-Württemberg	1418	133.01	10.7
Bayern	1564	126.26	12.4
Schleswig-Holstein	337	119.63	2.8
Nordrhein-Westfalen	2083	115.24	18.1
Saarland	115	107.98	1.1
Rheinland-Pfalz	438	107.93	4.1
Niedersachsen	784	98.25	8.0
Sachsen	424	97.49	4.3
Thüringen	153	63.96	2.4
Brandenburg	149	57.71	2.6
Sachsen-Anhalt	127	49.82	2.5
Meckl.-Vorpommern	83	47.56	1.7

Tab. 2: Dichte der registrierten GNU/Linux-Anwender in der Bevölkerung, Deutschland, Juli 2006 (Quelle: Linux Counter)

Der *Linux Counter* (counter.li.org) ist ein Projekt, bei dem GNU/Linux-Anwender sich und ihre Rechner registrieren können, um Datenmaterial für Statistiken zu liefern. Nur ein sehr kleiner Teil aller GNU/Linux-Benutzer ist registriert (etwa 140.000). Die Betreiber schließen aber mit Hilfe verschiedener, nicht ganz unplausibler Überlegungen auf eine tatsächliche Benutzerzahl von etwa 30 Millionen, was sich mit den obigen Statistiken decken würde (ca. 700 Millionen Internet-Benutzer gibt es heute weltweit, 30 Millionen davon wären 4,2 %).

Aussagekräftig ist jedoch die *Dichte* von GNU/Linux-Anwendern in bestimmten Ländern, also das Verhältnis zwischen der Zahl registrierter Anwender und der Gesamtbevölkerung. Eine Auswahl findet sich in Tabelle 1 (links). Die Tabelle 2 (oben) zeigt die entsprechenden Daten für die deutschen Bundesländer.

Tab. 1: Dichte der registrierten GNU/Linux-Anwender in der Bevölkerung, Auswahl, Juli 2006 (Quelle: Linux Counter)

Der freie Web-Browser Firefox ist weiter verbreitet als das Betriebssystem GNU/Linux. Das dürfte daran liegen, dass Firefox auch in einer Windows-Version verfügbar ist, und Benutzer darum nicht gleich ihr ganzes Betriebssystem austauschen müssen, um die Vorteile des freien Browsers nutzen zu können. Abb. 3 zeigt die relativen Anteile der Browser nach w3schools.com.

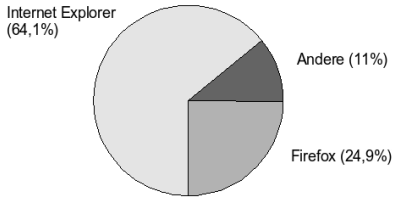
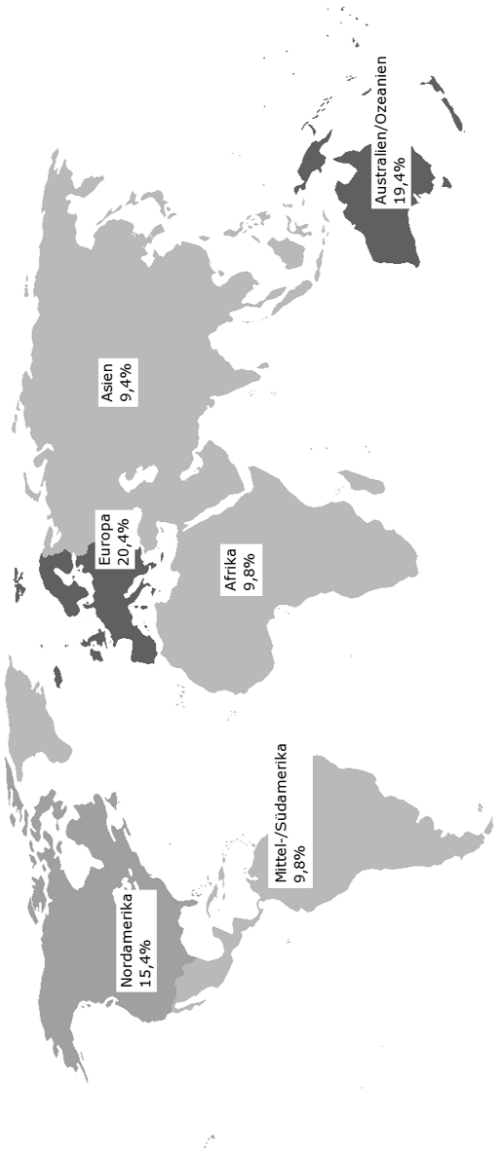


Abb. 3: Relativer Anteil der Web-Browser nach w3schools.com, Juni 2006 (Quelle: www.w3schools.com/browsers/browsers_stats.asp)

Die französische Firma XiTi erhebt regelmäßig Statistiken zu Firefox, die geographisch aufgeschlüsselt sind (ebenfalls durch das Zählen von Besuchern bei bestimmten, als repräsentativ geltenden Websites). Abb. 4 (rechts) zeigt die aktuellen Werte dieser Statistik für die einzelnen Kontinente.

Abb. 4: Geographische Verbreitung von Firefox nach Kontinenten, Juni 2006. (Quelle: www.xitimonitor.com/etudes/equipement16.asp)



Die unten stehende Abbildung vergleicht die Artikelzahl der 20 derzeit größten Wikipedias. Die gezeigten Zahlen sind die offiziellen Artikelzahlen der jeweiligen Wikipedias, bereinigt um administrative Seiten, wie z.B. Querverweise.

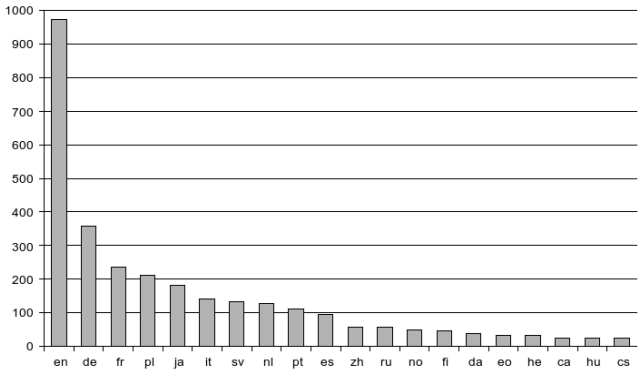


Abb. 5: Artikelzahlen der 20 größten Wikipedias (in 1000), Januar 2006 (Quelle: stats.wikimedia.org/EN/TablesArticlesTotal.htm)

Anmerkungen

1 Barry M. Leiner, Vinton G. Cerf et al.: *A Brief History of the Internet*. The Internet Society, o.J.;
Online-Version: www.isoc.org/internet/history/brief.shtml.
Siehe auch: www.catb.org/jargon/html/I/Internet.html

2 Sam Williams: *Free as in Freedom. Richard Stallman's Crusade for Free Software.* O'Reilly, 2002, S. 8f.;
Online: www.faifzilla.org

3 www.gnu.org

4 Richard Stallman: *Free Software, Free Society.* GNU Press, 2002, S. 31ff.;
Online: www.gnu.org/gnu/manifesto.html

5 Stallman, a.a.O., S. 41ff. Online: www.gnu.org/philosophy/free-sw.html

6 www.voresoel.dk

7 Stallman, a.a.O., S. 195ff. Online: www.gnu.org/licenses/gpl.html

8 Linus Torvalds, David Diamond: *Just for fun.* Harper Business, 2001

9 www.w3schools.com/browsers/browsers_stats.asp

10 www.opensource.org

11 Stallman, a.a.O., S. 55.
Online: www.gnu.org/philosophy/free-software-for-freedom.html

12 Eric S. Raymond: *The Cathedral & the Bazaar.* O'Reilly, 2001. Das Buch enthält den gleichnamigen Essay und weitere, einschließlich *Homesteading the Noosphere*. Die Online-Version des Essays *The Cathedral & the Bazaar* findet sich unter:
www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/

13 Raymond, a.a.O. Online:
www.catb.org/~esr/writings/homesteading/homesteading/

14 os.newsforge.com/os/05/04/11/118211.shtml

15 software.newsforge.com/article.pl?sid=05/04/25/130207

16 en.wikipedia.org/wiki/Open_Letter_to_Hobbyists

17 www.catb.org/~esr/halloween/

18 www.oreillynet.com/pub/wlg/104

19 Chicago Sun-Times, 1. Juni 2001
www.theregister.co.uk/2001/06/02/ballmer_linux_is_a_cancer/

20 perens.com/Articles/StandTogether.html

21 www.ibm.com/ibm/sjp/01-31-2001.html
www.ibm.com/press/us/en/pressrelease/643.wss

22 www.smartisans.com/burn_all_gifs.htm

23 www.eff.org/patent

24 www.heise.de/newsticker/meldung/68620

25 Exzellente Einführung in die Kryptographie und ihre Geschichte: Simon Singh: *Geheime Botschaften*. Carl Hanser, München, 2000

26 en.wikipedia.org/wiki/Key_length

27 Susan Landau: *Primes, Codes, and the National Security Agency*. Notices of the American Mathematical Society, [Special Article Series], Vol. 30, No. 1 (1983), pp. 7-10.; Online: www.totse.com/en/privacy/encryption/primes.html

28 Phil Zimmermann: *Why I Wrote PGP*. Teil der Bedienungsanleitung von PGP, 1991, aktualisiert 1999. Online: www.philzimmermann.com/EN/essays/WhyIWrotePGP.html

29 Phil Zimmermann: *PGP Source Code and Internals*. The MIT Press, 1995. ISBN 0-262-24039-4 (nicht mehr lieferbar) Online-Version des Vorworts: www.philzimmermann.com/EN/essays/BookPreface.html

30 Ariana Eunjung Cha: *To Attacks' Toll Add a Programmer's Grief*. Washington Post, 21. September 2001. Online: www.washingtonpost.com/ac2/wp-dyn/A1234-2001Sep20

31 Phil Zimmermann: *No Regrets About Developing PGP*. 24. September 2001.; Online: www.philzimmermann.com/EN/essays/PRZ_Response_WashPost.html

32 Eben Moglen: »*Die Gedanken Sind Frei«: Free Software and the Struggle for Free Thought*. Eröffnungsvortrag »Wizards of OS 3«, Berlin, 21. Juni 2004; Online: emoglen.law.columbia.edu/publications/berlin-keynote. html

33 www.salon.com/tech/feature/2000/06/14/love/

34 www.magnatune.com/info/why

35 www.magnatune.com

36 Jeff Lin (Harvey Danger), pers. Mitteilung, 30. April 2006

37 www.grokster.com. Um die Strafverfolgung zu erleichtern, weist der Autor darauf hin, dass die gezeigte IP-Adresse seine eigene war, und zwar am 13.1.2006 abends.

38 www.heise.de/newsticker/meldung/71649

39 Quelle: Statistisches Bundesamt. www.destatis.de/basis/d/verk/verktab6. php

40 www.mp3.com/news/stories/4310.html

41 tor.eff.org

42 freenetproject.org

43 Neda Ulaby: *Sony Music CDs Under Fire from Privacy Advocates*. National Public Radio, Morning Edition, 4. November 2005. Online: www.npr.org/templates/story/story.php?storyId=4989260

44 Moglen, a.a.O.

45 www.aacsla.com/specifications/

46 Tekla S. Perry: *Loser: DVD Copy Protection, Take 2*. In: IEEE Spectrum, Januar 2005; Online: www.spectrum.ieee.org/jan05/2703

47 Lloyd Grove: Lucas: *Big pics are doomed*. New York Daily News, 6. März 2006; Siehe auch: Anders Bylund: *Lucas: the blockbuster is doomed*. Ars Technica, 6. März 2006; Online: arstechnica.com/news.ars/post/20060306-6322.html

48 Umberto Eco: *Vegetal and mineral memory: The future of books*. Al-Ahram Weekly, Issue No. 665; 20-26 November 2003. Online: weekly.ahram.org.eg/2003/665/bo3.htm

49 Lawrence Lessig: *Freie Kultur. Wesen und Zukunft der Kreativität*. Open Source Press, München, 2006.

50 www.wynia.org/wordpress/2005/10/29

51 Eco, a.a.O.

52 www.gutenberg.org

53 books.google.com

54 en.wikipedia.org/wiki/Kasparov_versus_the_world

55 David R. Sands: *Kasparov vs. World ending on sour note*. The Washington Times, 21. Oktober 1999; Online (Kopie): members.allstream.net/~pmarko/washtimes.htm

56 www.gracenote.com

57 www.freedb.org

58 www.musicbrainz.org

59 www.wikipedia.org

60 www.gnu.org/encyclopedia/free-encyclopedia.html

61 nupedia.8media.org

62 *nature* 438, 900-901 (2005). Online: www.nature.com/nature/journal/v438/n7070/full/438900a.html

63 corporate.britannica.com/britannica_nature_response.pdf

64 *nature* 440, 582 (30 March 2006). Online: www.nature.com/nature/journal/v440/n7084/full/440582b.html

65 John Seigenthaler Sr., *A false Wikipedia ›biography‹*, USAToday, 29. November 2005; Online: www.usatoday.com/news/opinion/editorials/2005-11-29-wikipedia-edit_x.htm; ders., *Truth can be at risk in the world of the web*, The Tennessean, 4. Dezember 2005; Online: www.tennessean.com/apps/pbcs.dll/article?AID=/20051204/NEWS01/512040352/1006/NEWS

66 poynter.org/forum/view_post.asp?id=10748

67 K. Seelye: *A Little Sleuthing Unmasks Writer of Wikipedia Prank*. New York Times, 11. Dezember 2005; Online: www.nytimes.com

68 www.lessig.org/blog/archives/003068.shtml

69 ebd.

70 www.msn.co.uk/thoughtthieves

71 www.genealogienetz.de

72 www.dooyoo.de, www.kelkoo.de, etc.

73 Stephen M. Maurer, Arti Rai, Andrej Sali: *Finding Cures for Tropical Diseases: Is Open Source an Answer?* PLoS Med 1(3): e56, 2004. Online: medicine.plosjournals.org/perlserv/?request=get-document &doi=10.1371/journal.pmed.0010056

74 www.ornl.gov/sci/techresources/Human_Genome/home.shtml

Alle Links wurden am 25. Juli 2006 überprüft.
Ein Online-Verzeichnis mit evtl. aktualisierten Links befindet sich unter:
www.die-befreiung-der-information.de/links

Index