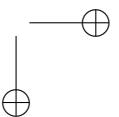
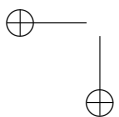
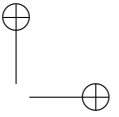
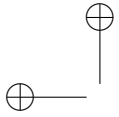




*Copying is an Act of Love. Open Source in den Digitalen Künsten* von Jonas Probst steht unter der Creative Commons Attribution-ShareAlike 4.0 International Lizenz. Die Weiterverwendung ist nur unter gleichen Bedingungen und mit Namensnennung möglich. Eine Kopie der Lizenz befindet sich unter <http://creativecommons.org/licenses/by-sa/4.0/>.

Diese Arbeit wurde in LaTeX verfasst, einer auf dem Open Source Textsatzsystem TeX beruhenden Auszeichnungssprache. Der Quellcode befindet sich unter <http://github.com/jonasprobst/copying-is-an-act-of-love>.



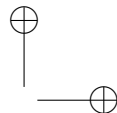
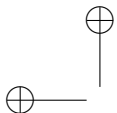
# Copying is an Act of Love

Open Source in den Digitalen Künsten

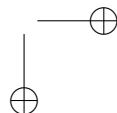
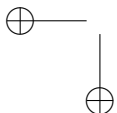
Jonas Probst

27. Februar 2015

Hochschule der Künste Bern  
MA Contemporary Arts Practice

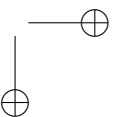
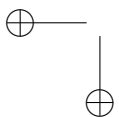
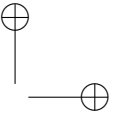
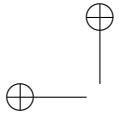


Herzlichen Dank an  
Andi Schoon, Rikka Feuz, Roger Villars, Micha Haarenberg und Hansjürg Wenger



# Inhaltsverzeichnis

<b>Einleitung</b>	<b>5</b>
<b>Was ist Open Source?</b>	<b>9</b>
Ursprung des offenen Quellcodes . . . . .	11
Ein Rückblick: Hackerethik und der Geist der Quelloffenheit . . . . .	15
Mit Offenheit Geschäfte machen . . . . .	18
<b>Open Source in der Digitalen Kunst</b>	<b>19</b>
!Mediengruppe Bitnik - Ein Paket für Herrn Assange . . . . .	23
Christian Waldvogel - Random Planet Production Machine (RPPM) . .	26
<b>Copying is an Act of Love</b>	<b>28</b>
<b>Literaturverzeichnis</b>	<b>32</b>
<b>Abbildungen</b>	<b>35</b>
<b>Anhang: Liste der erwähnten Open Source Projekte</b>	<b>47</b>



## Einleitung

In meiner künstlerischen Praxis bin ich immer wieder mit dem Thema Open Source konfrontiert. Das Werk *Color Permutations* (2012) nutzt ein Open Source Framework<sup>1</sup> und läuft in einem Open Source Webbrowser auf einem Open Source Betriebssystem. Die *Ewige Pfütze* (2013) wird von einem Open Source Mikrocontrollersystem gesteuert, für welches ich auf einer Open Source Programmierumgebung in einer dem Open Standard unterliegenden Programmiersprache<sup>2</sup> Algorithmen entwickelt habe.

Auch für mein nächstes Projekt werde ich von Open Source Hard- und Software Gebrauch machen, denn damit stehen mir eine Vielzahl von ausgeklügelten Bausteinen zur Verfügung, die sich in anderen Projekten bereits mehrfach bewährt haben. Ähnlich einem Lego-Baukastensystem kann ich einzelne Teile zu einem System zusammenfügen, welches dann die technische Grundlage für die Umsetzung meiner Arbeit bildet. Die einzelnen Bausteine selbst zu entwickeln übersteigt meine Ressourcen, auch steht solch ein massgeschneidertes System nicht bei irgend einem Supermarkt im Regal. Dank den Open Source Technologien kann ich meine Vision kostengünstig und effizient umsetzen, ohne das Rad neu erfinden zu müssen. Doch was ist Open Source (auf Deutsch offener Quellcode) eigentlich? Der Begriff stammt ursprünglich aus der Software Entwicklung und bedeutet dort in seiner Essenz, dass ein Autor freiwillig auf einen Teil seiner Urheberrechte verzichtet und den Quellcode seiner Software der Öffentlichkeit zugänglich macht. Diese Software kann sodann frei weiterverwendet, weiterentwickelt und den eigenen Bedürfnissen angepasst werden, ohne sich um Patent- oder Lizenzrechte sorgen zu müssen.

Was seit den neunziger Jahren die Computerwelt revolutioniert, zieht heute Kreise weit über Software hinaus. Da gibt es nebst den bereits erwähnten Projekten auch Strickautomaten, 3D-Drucker, Lexika, Handprothesen, humanoide Roboter, Video

---

<sup>1</sup>Eine Liste der in der Einleitung erwähnten Open Source Projekte mit Weblinks findet sich im Anhang.

<sup>2</sup>JavaScript ist ein ECMA-Standard (Siehe <http://www.ecma-international.org/ecma-262/5.1>, 25.02.2015). ECMA ist eine internationale Normungsorganisation zur Normung von Informations- und Kommunikationssystemen mit Sitz in Genf.

Games und Gamekonsolen. Dank Open Source CNC-Fräsmaschinen lassen sich Open Source Häuser mit Open Source Möbelstücken einrichten. Zur in Open Source Zeichensätzen dargestellten Open Source Sciencefiction-Novelle genießt der Kenner ein Open Source Bier. Und wo es Freibier gibt, ist auch Kultur zu finden: Pop-Musik, Comic Strips, Animationsfilme, Spielfilme, sogar ein Filmfestival und eine Open Source Galerie.

Der Hype hat auch die bildende Kunst erreicht. Besonders bei den *Digitalen Künsten*<sup>3</sup>, wenn Software oder Elektronik ins Spiel kommt, ist Open Source nicht weit, sei es im Theater, Tanz oder in der Performance, der elektronischen Musik, Internetkunst oder wie in meinem Beispiel der Installation. Doch jedes Mal, wenn ich zur Umsetzung einer Arbeit auf Open Source setze, sehe ich mich mit den gleichen Fragen konfrontiert. Was bedeutet Open Source für die Kunst? Für meine Kunst? Für meine Autorschaft? Wie stehe ich dazu, wie gehe ich damit um? Eine ganze Armee von Individuen kollaboriert – meist unbezahlt und in der Freizeit – übers Internet an Projekten, ohne die ich meine Kunstwerke nicht realisieren könnte. Nutze ich das schamlos aus oder gebe ich etwas zurück? In welcher Form könnte ich als Künstler in diesem Umfeld überhaupt mitwirken?

Zudem kommt noch die sogenannte *Share Alike* Klausel, auch *Copyleft* genannt, hinzu. Weil ein Autor seine Urheberrechte explizit abgeben muss, stehen alle Open Source Projekte unter einer Open Source Lizenz.<sup>4</sup> Natürlich ist jeder Autor frei, für welche Geschmacksrichtung er sich entscheidet: Ob er zum Beispiel alle Urheberrechte abtritt, eine kommerzielle Nutzung ausschließt oder eben vorgibt, dass alle Bearbeitungen unter gleichen Bedingungen weitergegeben werden müssen.<sup>5</sup> Der Konsequenz des Letzteren sind sich wohl die wenigsten Künstler bewusst, die von Open Source Gebrauch machen. Denn wenn ich demnach Ressourcen nutze, die einer *Share Alike* Lizenz unterstehen, verpflichte ich mich dazu, mein darauf aufbauendes Werk wiederum unter derselben Lizenz der Öffentlichkeit zu überlassen.

Welche Auswirkungen hat ein solches Vorgehen auf den Kunstmarkt? Beschränkt sich das Prinzip von Open Source in der Kunst lediglich auf die Technik zur

---

<sup>3</sup>Ein von Cat Hope und John Ryan geprägter Überbegriff der Kunstformen New Media Art, Electronic Art, Computer Art, Internet Art, Post Media Art, Telematic Art, Virtual Art usw. Siehe Hope und Ryan 2014, S. 3-12.

<sup>4</sup>Eine Übersicht der von der Open Source Foundation anerkannten Lizenzen findet sich auf <http://opensource.org/licenses> (12.02.2015).

<sup>5</sup>Die Creative Commons <http://creativecommons.org> (12.02.2015) bietet eine Alternative zum Copyright an: Ein modulares Lizenzsystem, welches eine feinere Abstufung des «Alle Rechte vorbehalten» zulässt. Siehe Casserly und Ito 2011, S. 7-8.

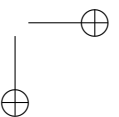
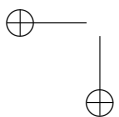
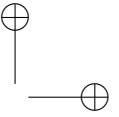
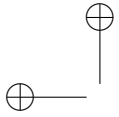


Herstellung oder findet es darüber hinaus Anwendung? Wenn ja, was wäre dann der Quellcode der Kunst, der offengelegt werden müsste?

Das Phänomen Open Source hat viele Seiten und diverse Schnittpunkte mit anderen Wissenschaften - dieses Thema umfassend abzuhandeln, übersteigt diese Arbeit bei weitem. Deshalb setze ich den Schwerpunkt einer ersten Annäherung bei der Frage, wie andere zeitgenössische Künstler mit dem Thema Open Source umgehen. Zuerst leite ich aus der Entstehungsgeschichte von Open Source einige Kern- und Kennwerte ab, um darauffolgend die !Mediengruppe Bitnik und Christian Waldvogel am Beispiel der gemeinsamen Ausstellung im Helmhaus Zürich<sup>6</sup> zu untersuchen. Schliesslich entwickle ich aus diesen Erkenntnissen einen Prototyp für den Umgang mit Open Source in meiner eigenen künstlerischen Praxis.

---

<sup>6</sup>Christian Waldvogel, *unknown* (Waldvogel 2014) und !Mediengruppe Bitnik, *Delivery for Mr. Assange* (Ryser 2014) im Helmhaus Zürich 14.02. - 06.04.2014.



## Was ist Open Source?

Der Begriff *Open Source* stammt aus der Softwareentwicklung und heisst soviel wie «offener Quellcode». Der Quellcode beschreibt den Ablauf eines Programms in einer für Menschen lesbaren Form. Meist ist dies ein strukturierter Text im Syntax einer Programmiersprache. Allerdings kann ein Computer mit dem Quellcode nicht viel anfangen. Damit er solche Programme ausführen kann, müssen sie in maschinenlesbare Sprache übersetzt, sprich kompiliert werden.

Aus der resultierenden Struktur von Einsen und Nullen ist es gänzlich unmöglich, den ursprünglichen Quellcode zu rekonstruieren. Viele Softwarehersteller veröffentlichen deshalb ihre Programme ausschliesslich in kompilierter Form als Binärdateien, den Executables. Auf diese Weise kann so genannte proprietäre Software als Produkt vermarktet werden.

Was uns heute selbstverständlich scheint, war nicht immer so: «In den frühen Tagen des Computers war alle Software quelloffen und frei».<sup>7</sup>

### Ursprung des offenen Quellcodes

Anfänglich bestand für Software kein eigenständiger Markt. Sie war kostenlose Zugabe zur Hardware oder wurde von ihren Nutzern gleich selbst geschrieben. Quellcode zirkulierte frei, Programme wurden laufend verbessert, weiterentwickelt und die Ergebnisse an die Nutzergemeinschaft zurück gegeben.

Erst in den 70er Jahren, so schildert es der Medienwissenschaftler Volker Grassmuck, zeichnete sich eine Trendwende ab. Unter dem Druck des vom US-Justizministerium eingeleiteten Kartellverfahrens gab IBM die Bündelung von Hardware mit Software, Peripheriegeräten, Wartung und Schulung auf. Diese Entkopplung schuf die Voraussetzungen für eine eigenständige Softwareindustrie - Software wurde erstmals zur Ware.<sup>8</sup>

---

<sup>7</sup>Grassmuck 2004, S. 202.

<sup>8</sup>Grassmuck 2004, S. 203.

Die darauffolgende vollständige Kommerzialisierung von Software hängt für Grassmuck mit der Erfolgsgeschichte von Microsoft (gegründet 1975) zusammen. Diese begann mit dem Betriebssystem DOS, das Microsoft nicht selbst entwickelt, sondern von einer anderen Firma aufgekauft hatte. Als IBM in den 80er Jahren ein Betriebssystem für den ersten Personal Computer suchte, fiel der Entscheid eher durch Zufall<sup>9</sup> auf DOS. Doch sicherte sich Microsoft dank der schnellen Verbreitung des PCs innerhalb von kurzer Zeit ein Monopol, das es mit der grafischen Benutzeroberfläche Windows und zahlreichen weiteren Anwendungen konsequent ausbaute. Dabei setzte es auf eine Strategie des *Bundling*: Die verschiedenen Microsoft-Programme bauen so aufeinander auf, dass sie schwer durch andere Software ersetzt werden können. Weil sie nur in kompilierter Form erscheinen, können sie nicht angepasst und weiterentwickelt werden. Auf diese Weise, so schliesst Grassmuck, machte Microsoft nicht nur seine Produkte, sondern auch sein System der kommerziellen Verwertung zum allgemeinen Standard: «Anfang der 80er war fast alle Software proprietär».<sup>10</sup>

Natürlich sträubten sich viele Urväter von Software gegen das auferlegte kommerzielle Korsett, das Copyright, den geschlossenen Quellcode und kommunikationswidrige Vertraulichkeitsvereinbarungen.<sup>11</sup> Der populärste Aufständische ist der exzentrische Software Freedom Aktivist und Programmierer Richard Stallman,<sup>12</sup> der 1984 seinen Job am Massachusetts Institute of Technology (MIT) kündigte, um ein Betriebssystem zu entwickeln, das wieder frei sein sollte. Denn nur wenn das Betriebssystem selbst - der Kern des Computers, der alle anderen Programme ausführt - *alle* Software-Freiheitsrechte einhält, kann es sie dem Anwender gewähren.<sup>13</sup>

Sein neues Betriebssystem sollte auf dem beliebten und weitverbreiteten *Unix* basieren. Dieses stammte aus der Zeit, in der Software noch frei war, doch wurde es vom amerikanischen Telekommunikations-Konzern AT&T allmählich kommerzialisiert. Der Plan war, Zeile um Zeile jedes einzelnen Teils des privatisierten Codes von Unix

---

<sup>9</sup>Das weiter verbreitete und bessere Betriebssystem für die damalige Generation von 8-Bit-Rechner war CP/M (Control Program for Microcomputers), das Gary Kildall entwickelt hatte. «Das Millionengeschäft mit IBM ging jedoch an Microsoft, das damals vor allem für das auf fast jedem Mikrocomputer verfügbare MS-BASIC bekannt war, weil Kildall einer Anekdote nach am entscheidenden Tag mit seinem Privatflugzeug unterwegs und damit nicht erreichbar war.» Grassmuck 2004, S. 205.

<sup>10</sup>Grassmuck 2004, S. 221.

<sup>11</sup>Diese auf engl. *Nondisclosure Agreements* verboten es explizit, Code frei anzupassen und untereinander auszutauschen.

<sup>12</sup>Die Webseite von Stallman bietet Einblick in sein vielseitiges politisches Engagement und weit über Software hinaus auch in sein intimstes Privatleben. Siehe <https://stallman.org> (17.12.2014).

<sup>13</sup>Stallman 2010, S. 9.

zu ersetzen. Stallman nutzte dazu die beginnende Vernetzung von Forschungseinrichtungen, Universitäten und Unternehmen über das Internet. Sein Projekt stellte er in Newsgroups vor und lud zur Mitarbeit ein. Der erste Coup, ein hocheffizienter C-Compiler *GCC* und ein sehr flexibler und anpassbarer Text Editor *Emacs*, stiess auf grosse Resonanz. Schnell fanden sich Gleichgesinnte, und schliesslich wurde zur Verbreitung der Programme und ihrer Ideologie die *Free Software Foundation* gegründet.<sup>14</sup>

Nun durften diese Programme keinesfalls das gleiche Schicksal treffen wie vormals Unix. Deshalb schrieb Stallman 1989 die erste Lizenz für freie Software, die *GNU General Public License* (GPL).<sup>15</sup> Copyleft schützt anders als das Copyright nicht den Rechteinhaber, sondern die Freiheit der Software selbst. Die GPL definiert vier Freiheiten, die dieses Prinzip des Copyleft gewährleisten: Ein Programm gilt nur dann als freie Software, wenn (1) sein Quellcode offen zugänglich ist, wenn (2) es kopiert und weitergegeben werden darf, wenn (3) Veränderungen vorgenommen und diese (4) wiederum weitergegeben werden dürfen.<sup>16</sup>

Auf der Grundlage der vier Freiheiten wurde das GNU-System kontinuierlich weiter entwickelt. 1990 fehlte nur noch der Kernel, die zentrale Verbindung zwischen Software und Hardware. Diese Aufgabe nahm der finnische Informatikstudent Linus Torvalds in Angriff. Bereits sehr früh veröffentlichte er das unfertige Programm im Internet und rief zur Kollaboration auf.<sup>17</sup> Aus allen Teilen der Welt meldeten sich Freiwillige, die bereit waren, Arbeit und Zeit ohne Bezahlung in ein gemeinsames Projekt zu investieren. Änderungen und Verbesserungen am Quellcode wurden von Torvalds unverzüglich eingearbeitet und auch täglich neu veröffentlicht. Torvald glänzte dabei nicht als Entwickler, sondern vorallem als Organisator und Motivator. Und so kam es, dass wider Erwarten Tausende von Menschen in einer offenen, locker gekoppelten Zusammenarbeit das komplexe Softwareprojekt Linux entwickelten.<sup>18</sup>

Mit dem Erfolg von Linux etablierte sich das Prinzip des offenen Quellcodes. Das kollaborative Entwicklungsmodell komplettierte das «Open Source Dreieck aus Technik, Recht und neuen Kooperationsformen»<sup>19</sup> und gewann rasch an Bedeutung. Zur Überraschung aller Free Software Enthusiasten entschied Netscape 1998, als

<sup>14</sup>Stallman 2010, S. 7-25; Vgl. auch Grassmuck 2004, S. 217-227 und Raymond 1999, S. 18-19.

<sup>15</sup>Die aktuellste Version ist die GNU GPL v3 (<https://www.gnu.org/licenses/>, 18.12.2014). Für ältere Versionen siehe <https://www.gnu.org/licenses/old-licenses/old-licenses.html> (18.12.2014).

<sup>16</sup>Siehe *GNU Free Software Definition*, Stallman 2010, S. 3.

<sup>17</sup>Siehe dazu Torvalds 1999.

<sup>18</sup>Siehe Raymond 2001; Vgl. auch Grassmuck 2004, S. 227.

<sup>19</sup>Deterding 2007.

erster proprietärer Softwarehersteller überhaupt, den Quellcode ihres gleichnamigen Internetbrowsers<sup>20</sup> unter einer Open Source Lizenz öffentlich zugänglich zu machen. Um weitere Unternehmen für die Sache zu gewinnen, wurde im selben Jahr die Open Source Initiative<sup>21</sup> um den Programmierer Eric Raymond gegründet. Zudem wurden neue liberalere Lizenzen<sup>22</sup> geschaffen, die speziell auf Unternehmen zugeschnitten waren. Der neue Name *Open Source Software* (OSS) verbreitete sich schnell, sorgte aber bald für Konflikte. Für Anhänger der alten Bezeichnung *Free Software* verkürzte er das umfassende Prinzip der Freiheit einer Software auf das technische Kriterium des offenen Quellcodes. In der Bezeichnung *FOSS* (Free and Open Source Software) beziehungsweise *FLOSS* (Free/Libre and Open Source Software) wurde ein Kompromiss gefunden, der sich aber bis heute noch nicht richtig durchgesetzt hat.

Ungeachtet der Bezeichnung steht Open Source für ein Konzept, dessen Erfolg mittlerweile kaum zu übersehen ist: Nebst einer Vielfalt von Code-Bausteinen, sogenannten Frameworks und Libraries, laufen Linux basierte Betriebssysteme auf vielen Firmenservern, Privatcomputern und Smartphones. Googles Chromium und Mozillas Firefox sind die am weitesten verbreiteten Webbrowser im Netz,<sup>23</sup> und Programme wie OpenOffice, Gimp, Thunderbird, VLC und Wordpress sind aus dem Alltag kaum mehr wegzudenken. Sogar bei Microsoft zeichnet sich ein Umdenken ab. Im Herbst 2014 startete Microsoft den Openness Blog<sup>24</sup> und veröffentlichte den Quellcode für die Web-Entwicklungsplattform DotNet.<sup>25</sup>

Open Source ist also mehr als nur ein Hype, wie die stetig wachsenden Marktanteile

---

<sup>20</sup>Heute bekannt unter dem Namen Mozilla Firefox (<http://www.mozilla.org>, 25.02.2015). Mozilla wurde von Netscape gegründet, um unter einem neutralen Namen den Quellcode zu veröffentlichen und um die Entwicklergemeinschaft zu pflegen. Hamerly, Paquin und Walton 1999, S. 91.

<sup>21</sup>Siehe dazu <http://opensource.org> (18.12.2014).

<sup>22</sup>Die GNU GPL Lizenz ist viral. Werden Programmteile, die unter dieser Lizenz stehen mit anderen kompiliert, unterliegt das Endprodukt und somit der gesamte Quellcode der GPL. Die GPL ist demnach nicht kompatibel mit anderen Lizenzformen und verhindert so gezielt, nur Teile eines Programms quelloffen zu machen und die anderen «geschlossen» zu verkaufen. Siehe Hamerly, Paquin und Walton 1999, S. 93.

<sup>23</sup>Gemäss dem Internet Statistik Dienst w3counter.com (<http://www.w3counter.com/globalstats.php>, 18.12.2014) sind zur Zeit Chrome (42.7%) gefolgt von Firefox (16.4%) und Internet Explorer (15.8%) die meist genutzten Webbrowser.

<sup>24</sup>Siehe <http://openness.microsoft.com/blog> (24.12.2014); Vgl. auch <http://www.microsoft.com/en-us/openness>. (24.12.2014)

<sup>25</sup>Der Quellcode kann unter <https://github.com/Microsoft/dotnet> (24.12.2014) eingesehen werden.

quelloffener Software eindrücklich belegen.<sup>26</sup> Doch warum ist das Phänomen so erfolgreich? Warum setzt eine wachsende Anzahl von Firmen und Regierungen<sup>27</sup> auf Open Source? Und was treibt einen Software-Entwickler dazu, in seiner Freizeit Software zu schreiben, nur um sie anschliessend zu verschenken?

## Ein Rückblick: Hackerethik und der Geist der Quelloffenheit

Die Computer Enthusiasten des MIT Modelleisenbahn Clubs (MIT Tech Model Railroad Club) nannten sich selber Hacker, nach der Aktivität der Programmierer, die auf der Computertastatur herum hacken.<sup>28</sup> Dieser Name beschreibt eine Gruppe von Menschen, die aus der seit 1945 herangewachsenen Computerkultur hervorging. Die vorwiegend aus jugendlichen Studierenden und Universitätsangestellten bestehende Gruppe hatte ihre eigene fachspezifische Sprache,<sup>29</sup> eigene Insiderwitze<sup>30</sup> und Dress-codes.<sup>31</sup> Hacker waren vom Computer fasziniert – programmieren war das Mittel zur Erkundung seiner Möglichkeiten. Sie waren aus blosser Wissbegierde und nicht aus finanziellen Interessen motiviert. Vermutlich dank ihrer wissenschaftlichen Prägung war es für sie selbstverständlich, ihre Forschungsergebnisse zu publizieren und sie anderen zur Examination und freien Weiterverwendung zu unterbreiten. So wurde Software gratis, frei und bedenkenlos unter den verschiedenen Forschungszentren ausgetauscht,<sup>32</sup> und es herrschte ein heiteres Klima des offenen Miteinanders.

Der hier beschriebene Typ von Hackern ist nicht zu verwechseln mit «Computer Kriminellen», wie man sie aus den Medien im Zusammenhang mit Computersi-

<sup>26</sup>Eine Studie im Auftrag der Europäischen Kommission im Jahr 2006 zeigte, dass der Marktanteil quelloffener Software kontinuierlich wächst und prognostiziert ihr eine entscheidende Bedeutung für die europäische Wirtschaftsentwicklung. Siehe Kleijn 2007.

<sup>27</sup>Sogar die Europäische Union hat eine Strategie für Open Source Software. Siehe «Strategy for internal use of OSS at the EC» unter [http://ec.europa.eu/dgs/informatics/oss\\_tech/index\\_en.htm](http://ec.europa.eu/dgs/informatics/oss_tech/index_en.htm) (24.12.2014).

<sup>28</sup>Levy 1984, S. 10.

<sup>29</sup>Siehe Jargon File, das Kompendium der Hacker-Ausdrucksweise unter <http://www.catb.org/jargon> (30.01.2015).

<sup>30</sup>Zum Beispiel die Hacker Bibel: «The Gospel According to Tux» siehe <http://www.armoredpenguin.com/~regan/penguins/tux.html> (30.01.2015). Tux ist der Name des Linux Pinguin-Maskottchens.

<sup>31</sup>Für einen Einblick in die Hackerkultur siehe Raymond 1999, S. 16; Vgl. auch Raymond 1996.

<sup>32</sup>So auch *Spacewar!*, eines der ersten Computer Spiele überhaupt und Relikt der Hackerkultur. Es wurde 1961 am MIT von Steven Russel für den DPD-1 entwickelt und verbreitete sich schnell in andere Forschungszentren, wo es ergänzt und weiterentwickelt wurde.

cherheit kennt. Die Ersteren waren Computer Liebhaber und Programmierer aus Leidenschaft, deren Ehrenkodex im Gegensatz zu den «Computer Kriminellen» unterschiedlicher nicht sein könnte.

Der Reporter Steven Levy versuchte als Erster, die neue computerzentrierte Subkultur zu beschreiben. Unter dem Begriff *Hacker Ethik* leitete er die nachfolgenden sechs Kernwerte ab. (1) «Access to computers - and anything which might teach you something about the way the world works - should be unlimited and total. Always yield to the Hands-On Imperative».<sup>33</sup> Nur wenn ein Hacker uneingeschränkten Zugang hat, kann er von bereits existierenden Ideen und Systemen lernen, um diese weiter zu entwickeln und zu verbessern. Diesen Drang zur Verbesserung sieht Levy indes als zentrales Kennzeichen eines Hackers, den sogenannten *Hands-On Imperative*: Programmierer sind darauf getrimmt, Programme in ihre Einzelteile zu zerlegen, sie zu analysieren und auf diesen Erkenntnissen aufbauend den Algorithmus zu verbessern oder neu zu erfinden. Dieses Prinzip weiten Hacker auf alle anderen Bereiche ihres Lebens aus. (2) «All information should be free».<sup>34</sup> Aufbauend auf dem Prinzip des Zugangs muss Information auch frei von Einschränkungen sein, damit ein Hacker das System verbessern kann. (3) «Mistrust Authority - Promote Decentralization».<sup>35</sup> Nur eine flache Hierarchie in einem offenen System bietet dem Hacker die oben genannten Freiheiten. Jegliche Art von Bürokratie erschwert seine Arbeit. (4) «Hackers should be judged by their hacking not bogus criteria such as degrees, age, race, or position».<sup>36</sup> Die Hacker sind leistungsorientiert, in Anbetracht ihres Könnens sind genannte Kriterien unwichtig. (5) «You can create art and beauty on a computer».<sup>37</sup> Computer können benutzt werden, um Kunst und Schönes zu schaffen. Für einen Hacker kann ein durchdachtes, effizientes und schön arrangiertes Programm durchaus ein Kunstwerk sein, denn schliesslich sind dies Qualitätsmerkmale für Quellcode. (6) «Computer can change your life for the better».<sup>38</sup> Ein Hacker ist so begeistert von der unbegrenzten Welt des Computers, dass er Andere daran teilhaben lassen möchte.

Die von Levy beschriebene *Hacker Ethik* ist stark mit der Situation der Hacker am MIT um 1984 verstrickt. Zum Beispiel mussten die Studenten die zur Verfügung stehende Zeit am einzigen Computer im Institut untereinander teilen. Gleichzeitig verstanden sich die Hacker aber als Vorreiter eines neuen Technologiezeitalters

---

<sup>33</sup>Levy 1984, S. 27

<sup>34</sup>Levy 1984, S. 27

<sup>35</sup>Levy 1984, S. 28

<sup>36</sup>Levy 1984, S. 30

<sup>37</sup>Levy 1984, S. 30

<sup>38</sup>Levy 1984, S. 33



und meinten es mit diesen Kernwerten, auch über die Universitätsgrenzen hinaus, durchaus ernst.<sup>39</sup>

Der finnische Philosoph und Computerwissenschaftler Pekka Himanen definierte 2001 den mittlerweile weit verbreiteten Begriff der Hackerethik in seinem gleichnamigen Buch etwas allgemeiner. Für ihn ist ein Hacker jemand, der seiner Leidenschaft folgt und dabei Gutes für alle schafft.<sup>40</sup> Himanen sieht darin eine neue Arbeitsethik, welche die protestantisch geprägte in Frage stellt. Statt der von der protestantischen Arbeitsethik betonten Kontrolle, steht bei der Hackerethik selbstbestimmtes Arbeiten im Vordergrund: «For hackers, passion describes the general tenor of their activity, though its fulfillment may not be sheer joyful play in all its aspects».<sup>41</sup> Der Hacker Codex, so Himanen, ähnelt dem Idealbild der Wissenschaft, einer wissenschaftlichen Gemeinschaft nach platonischem Vorbild also.<sup>42</sup> Himanen und Levy sind sich einig, dass das oberste Ziel Kreativität ist: «A hacker who lives according to the hacker ethic [...] gains the community's highest respect. This hacker becomes a true hero when she or he manages to honor the seventh and final value: [...] creativity - that is, the imaginative use of one's own abilities, the surprising continuous surpassing of oneself, and the giving to the world of a genuinely valuable new contribution».<sup>43</sup>

Zusammengefasst ist die Hackerethik also eine Gesinnung des freiwilligen und selbst gewählten Arbeitens, der Kooperation und des Teilens über alle Grenzen hinaus, der radikalen Offenheit und Transparenz<sup>44</sup> aller Informationen und der Freiheit, all diese Werte selbstbestimmt auszuleben, mit dem Ziel, zu einer besseren Welt beizutragen.

Mit diesem stolzen Credo voller Idealismus und waghalsigem Optimismus, gründete

---

<sup>39</sup>Siehe Levy 1984, S. 26-37.

<sup>40</sup>Himanen 2001, S. 3-6. Der australische Medienwissenschaftler McKenzie Wark vertritt in «A Hacker Manifesto» eine ähnliche Sicht wie Himanen. Seiner Ansicht nach ist alles und in allem Code. Der Hacker abstrahiert diese Datenfülle und schafft daraus Neues, sei es in der Mathematik, Musik, Kunst usw. Siehe McKenzie 2004, S. 13.

<sup>41</sup>Himanen 2001, S. 18

<sup>42</sup>Himanen 2001, S. 7-19.

<sup>43</sup>Himanen 2001, S. 141.

<sup>44</sup>Der Philosoph Byung-Chul Han steht solch radikalen Transparenzbestrebungen kritisch gegenüber. Durch Ideologisierung, Fetischierung und Totalisierung von Transparenz droht die heutige *Transparenzgesellschaft* in eine Kontrollgesellschaft umzuschlagen. Statt unsere Gesellschaft freier und demokratischer zu machen, erzeugt sie nur neue Zwänge und nährt ein Klima des Verdachts. Absolute Transparenz, so Han, ist eine Antiutopie, die längst im Begriff ist, Wirklichkeit zu werden: «Die unzähligen Überwachungskameras verdächtigen jeden von uns. [...] Das Internet erweist sich als ein digitales Panoptikum». Han 2012; Vgl. auch Han 2013.

der «letzte wahre Hacker»<sup>45</sup> Richard Stallman 1985 die Free Software Foundation und startete damit die Open Source Bewegung - überraschenderweise mit grossem Erfolg.

## Mit Offenheit Geschäfte machen

Ein puristisches Open Source Projekt hat keine finanziellen Absichten. Es ist weder Resultat einer Marktstudie, noch geht es aus einem gross angelegten Start-Up hervor. Es entsteht vielmehr aus einem kollektiven Bedürfnis heraus.

«Ein Programmierer kratzt sich dort, wo es ihn juckt»,<sup>46</sup> zum Beispiel, wenn ein Druckertreiber nicht so funktioniert wie erwartet. Der Legende nach hat sich Stallman dermassen über den vergeblichen Gang zum Drucker geärgert, dass er kurzentschlossen einen eigenen Druckertreiber schrieb, welcher über den Status des Druckvorgangs informierte.<sup>47</sup> Täglich entstehen unzählige ähnliche Projekte, da jederman ein Projekt starten kann. Entscheidend ist letztlich nur die Relevanz der Aufgabenstellung und der Nutzen für die Gemeinschaft (Community), denn nur einschlägige Projekte werden genug Aufmerksamkeit auf sich ziehen, um weitere Programmierer dazu zu bewegen, sich in das Projekt einzubringen.

Das ausschlaggebende Qualitätsmerkmal von Open Source Projekten ist nebst der Verbreitung die Grösse der Community. Denn nur wenn genügend Mitglieder die Sache vorantreiben, ist ein Weiterbestehen garantiert. Überhaupt ist Kollaboration der zentrale Aspekt, da das Schreiben von Software eine komplexe und zeitaufwendige Arbeit ist. Computer sind grundsätzlich dumm. Das Einzige worin sie brillieren, ist das Ausführen von immer gleichen Arbeitsschritten - dies aber tun sie in einer ungeheuren Geschwindigkeit. Somit ist es die Aufgabe des Software Entwicklers, eine vielschichtige Aufgabenstellung zu abstrahieren und in zig einfache Arbeitsschritte zu unterteilen. Diese müssen anschliessend möglichst detailliert beschrieben werden, damit ein Computer sie ausführen kann. Beide Arbeitsschritte, die Abstrahierung sowie das Niederschreiben, lassen sich im Team viel effizienter erledigen als im Einzelgang. Während des Schreibens solcher Instruktionen schleichen sich gerne

---

<sup>45</sup>Levy 1984, S. 419.

<sup>46</sup>Raymond 2001, S. 4.

<sup>47</sup>Raymond 1999, S. 32; Vgl. auch Stallman 2010, S. 9.

kleine Denk- und Schreibfehler ein, sogenannte *Bugs*,<sup>48</sup> die das Programm zum Abstürzen bringen können. Auch da gilt, je mehr geschulte Augenpaare den Programmtext überprüfen, umso mehr Fehler können frühzeitig erkannt und behoben werden. Und nistet sich trotzdem ein Bug ein, so werden ihn tausend Kammerjäger schneller finden als ein einziger.<sup>49</sup>

Gerade deshalb ist Open Source für Softwarefirmen ein interessantes Modell, denn viele Kundenprojekte kommen mit ähnlichen Aufgabenstellungen. Wer also geschickt arbeitet, isoliert einzelne Bausteine und kann diese dann immer wieder einsetzen. Proprietäre Softwarehersteller entwickeln ihre eigenen Code-Bausteine hinter verschlossenen Türen. Das führt immer wieder dazu, dass mehrere Unternehmen gleichzeitig ein fast identisches Produkt erfinden. Ein solches Vorgehen ist träge, teuer und unnötig aufwändig. Open Source bietet dazu eine einfache Alternative, die Zusammenarbeit. In neutralen Projekten entwickeln die verschiedenen Interessenten gemeinsam mit der Konkurrenz die Softwaregrundlage der Produkte. Dies spart personelle Ressourcen, erleichtert die Fehlerbehebung und garantiert eine innovative Weiterentwicklung. Verkauft wird nicht mehr die Software selbst, sondern Dienstleistungen wie IT-Beratungen, kundenspezifische Anpassungen oder der Unterhalt von Infrastrukturen.<sup>50</sup>

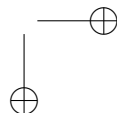
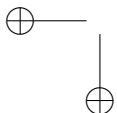
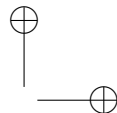
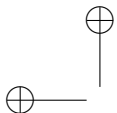
Eine solche Kollaboration funktioniert nur, wenn alle Parteien gleichgestellt sind: Alle haben das gleiche Mitspracherecht und uneingeschränkten Zugriff auf den Quellcode. Dieser kann nach Belieben verändert, kopiert und weiterverbreitet werden. Dazu muss jeder auf die Urheberrechte seines Beitrages verzichten, im Wissen darum, dass der Mehrwert in der Zusammenarbeit liegt. Deshalb braucht es einen Vertrag, der nicht den Urheber, sondern die Freiheit und Offenheit des Quellcodes schützt - die Open Source Lizenz.

---

<sup>48</sup>Die Bezeichnung «Bug» (auf Deutsch Käfer) für Softwarefehler und «Debugging» für deren Suche und Behebung, stammt der Legende nach von der Computerwissenschaftlerin Grace Hopper (1906-1992). Als Mitarbeiter an der Harvard Universität um 1940 als Ursache für einen Systemfehler des *Mark II* Computers eine in einem Relais feststeckende Motte fand, soll Hopper dies als *debugging* bezeichnet haben.

<sup>49</sup>Grassmuck 2004, S. 235-254.

<sup>50</sup>Grassmuck 2004, S. 329-357.



## Open Source in der Digitalen Kunst

Kunsthistorisch betrachtet finden sich erste Vorläufer der *Digitalen Künste* im frühen 20. Jahrhundert. Schon in Avantgarde-Strömungen wie Konstruktivismus, Futurismus und Bauhaus zeichnete sich eine Faszination für Technik und Maschinen-Ästhetik ab.<sup>51</sup> Zentral aber ist für die forschenden Künstler Cat Hope und John Ryan aber der Einfluss des Dadaismus. Viele Dada-Projekte entstanden nicht rein zufällig, sondern folgten formalen Anleitungen oder Regeln, Prinzipien, die sich auf die Digitale Kunst übertragen lassen: «Programming as a series of instructions and coding as the rules governing online environments are both vital concepts that lend shape to many digital arts projects today. Like their contemporary counterparts, Dadaists exploited communications technologies for conceptual, political and social commentary».<sup>52</sup> Von Dadaismus über Fluxus zu Konzeptkunst und Kunst im öffentlichen Raum – alle haben sie neue Formen von Performance, Theater und Installation hervorgebracht. Statt materiebasierte Kunstobjekte zu schaffen, erkunden sie Konzepte, Prozesse, Happenings und Interaktion.<sup>53</sup> Schliesslich waren es jedoch Ingenieure, die um 1950 begannen, Elektronik für künstlerische Zwecke einzusetzen, allen voran Ben Laposky (1914 – 2000) mit seinen abstrakten Oszilloskop-Bildern *Oscillons* (1953). Und nur wenig später folgte Michael Noll (\*1939) mit seinen computergenerierten Zeichnungen wie *Vertical-Horizontal No. 3* (1964). Er war Forscher an den *Bell Labs* in Murray Hill, New Jersey, dem Geburtsort von Transistoren, Laser und dem Betriebssystem UNIX.<sup>54</sup>

Die aufkeimende Zusammenarbeit von Kunst mit Natur- und Ingenieurwissenschaften der sechziger Jahre wurde von Projekten wie *Experiments in Art and Technology*

---

<sup>51</sup>Siehe Hope und Ryan 2014, S. 39.

<sup>52</sup>Hope und Ryan 2014, S. 40; Vgl. auch Taylor 2014.

<sup>53</sup>Solche sich in Bewegung befindende Kunstwerke bieten dem Rezipienten ständig neue Anknüpfungspunkte, um innere Beziehungen und Assoziationen zu schaffen. Je nach Lesart verändert sich dadurch das Werk – der Betrachter wird zum Mitschöpfer. Man spricht hier vom Konzept des *Offenen Kunstwerks*, ein von Umberto Eco geprägter Begriff. Siehe Eco 1973.

<sup>54</sup>Wands 2006, S. 24.

(E.A.T.)<sup>55</sup> weiter gefördert. Immer mehr Künstler begannen sich der neusten Technik zu bedienen, und so mündete das interdisziplinäre Miteinander schliesslich in Nam June's Videokunst *Participation TV* (1963), in Installationen wie *Legible City* (1988) von Jeffrey Shaws (\*1944) und Performances wie *Ping Body* (1996) von Stelarc (\*1946).<sup>56</sup> Mit der Verbreitung des Internets in den 90ern erhielten die Digitalen Künste weiter Aufschwung. Kunstwerke wie <http://www.wwwww.jodi.org/> von Joan Heemskerk (\*1968) und Dirk Paesmans (\*1965) sind sogar nur noch über einen Webbrowser zugänglich.<sup>57</sup>

Heute ist der Einsatz von Technik, Elektronik und Software in der künstlerischen Praxis weit verbreitet und immer häufiger anzutreffen. Institutionen wie das *MIT Media Lab* in Cambridge USA, das *Ars Electronica*<sup>58</sup> Center mit jährlichem Festival in Linz, das *Zentrum für Kunst und Medientechnologie* (ZKM) in Karlsruhe sowie das *Haus der Elektronischen Künste* (HEK) in Basel haben sich sogar auf diese Kunstrichtung spezialisiert. In und um diese Schmieden der Interdisziplinarität entwickeln sich die Digitalen Künste rasant weiter, und mit jeder neuen technischen Errungenschaft entstehen zig Kunstwerke, die diese nutzen oder kritisch hinterfragen. Es liegt also nahe, dass gravierende Veränderungen in der Technikwelt, wie etwa die Open Source Revolution, auch die Kunstwelt beeinflussen - bewusst oder unbewusst.

Die Frage ist also nicht, ob Open Source Einfluss auf die Digitale Kunst hat, sondern wie sich dieser abzeichnet. Wie wird Open Source Hard- und Software eingesetzt? Wie gehen die Künstler mit den Lizenzbestimmungen um? Wie wirken sich die Kernwerte Freiheit, Offenheit, Transparenz, Teilen und Kollaboration auf die zeitgenössische Kunst aus?

Im Frühjahr 2014 fanden im Helmhaus Zürich zeitgleich zwei Einzelausstellungen von Schweizer Künstlern statt, beides Vertreter der Digitalen Künste: im ersten Stock *unknown* von Christian Waldvogel, im zweiten *Ein Paket für Herrn Assange* von der !Mediengruppe Bitnik. Obwohl sich die Ausstellungen nicht explizit auf Open Source bezogen, ist der Umgang mit der Thematik in den ausgestellten Werken exemplarisch.

<sup>55</sup>E.A.T. wurde 1967 von den Ingenieuren Billy Klüver und Fred Waldhauer sowie den Künstlern Robert Rauschenberg und Robert Whitman gegründet. Ihr Ziel war die Förderung der Kollaboration von Künstlern mit Ingenieuren, indem sie ein Netzwerk zum Knüpfen von Kontakten aufbauten. Siehe Paul 2003, S. 16.

<sup>56</sup>Graham 2007, S. 96.

<sup>57</sup>Hope und Ryan 2014, S. 128.

<sup>58</sup>Bezeichnenderweise hat das Ars Electronica Festival im Jahr 1999 dem Open Source Linux Betriebssystem gar den Prix Ars Electronica verliehen. Siehe <http://residence.aec.at/rhizome/email/msg00043.html> (10.02.2015).

Um die zwei Standpunkte Bitnik und Waldvogel aufzuzeigen, analysiere ich in den folgenden Kapiteln je eine Arbeit. *Ein Paket für Herrn Assange* (2013) von Bitnik und *Random Planet Production Machine* (2013) von Waldvogel eignen sich besonders für einen Vergleich, denn sie wurden nicht nur zeitgleich am Helmhaus ausgestellt, sondern teilen auch das gleiche Entstehungsjahr. Zudem ist beiden Werken je ein Kapitel in den Katalogen und eine Seite auf den Künstlerhomepages gewidmet. Im nachfolgenden Vergleich gehe ich wie folgt vor: Zuerst beschreibe ich das Werk und seine Präsentation im Katalog und auf der Webseite. Anschliessend lege ich den Bezug von Werk und Künstler zu Open Source dar und ziehe schliesslich zusammenfassend Bilanz.

## **!Mediengruppe Bitnik - Ein Paket für Herrn Assange**

Die !Mediengruppe Bitnik ist ein Künstlerkollektiv, bestehend aus dem Schweizer Kernteam Domagoj Smoljo (\*1979) und Carmen Weisskopf (\*1976) sowie dem Journalisten Daniel Ryser (\*1979) und dem Londoner Filmemacher und Forscher Dr. Adnan Hadzi als Komplizen. Die künstlerische Strategie für ihre «Guerilla-Aktionen an der Grenze der digitalen Realität»<sup>59</sup> ist Hacking. So haben sie sich in *Surveillance Chess* (2012) Zugang zu einer Londoner CCTV Überwachungskamera beschafft, um den amtierenden Sicherheitsbeamten zum Schachspiel einzuladen. *Random Darknet Shopper* (2014) im Kunsthaus St. Gallen sorgte letzten Herbst in den Medien für Furore. Es ist ein Programm, das selbständig ein wöchentliches Budget von \$100 in Bitcoins auf dem Online-Schwarzmarkt verprasst. Nebst gefälschten Diesel Jeans und Nike Turnschuhen bestellte es ein HD-Scan eines ungarischen Passes und ein Päckchen Ecstasy Pillen.<sup>60</sup>

Der internationale Durchbruch gelang der Gruppe jedoch schon 2013 mit dem dreissig Stunden Live Mail Art Piece *Ein Paket für Herrn Assange* (2013). Am 16. Januar 2013 gab die Künstlergruppe ein Paket, adressiert an Wikileaks<sup>61</sup> Gründer und Internet Aktivist Julian Assange auf, der zu diesem Zeitpunkt mit politischem Asyl auf der ecuadorianischen Botschaft in London festsass. Im Paket befand sich eine Kamera mit GPS, die durch ein Loch in der Seitenwand alle zehn Sekunden

<sup>59</sup>Diener 2013.

<sup>60</sup>Siehe Live-Webseite zu Random Darknet Shopper unter <http://bitnik.org/r/> (17.02.2015).

<sup>61</sup>Wikileaks (<https://wikileaks.org/>, (04.02.2015) ist eine journalistische Nonprofit-Organisation, die sich auf die Veröffentlichung von Geheimdokumenten aus anonymen Quellen spezialisiert hat.

ein Foto schoss und auf eine Webseite lud. Auf Twitter liess sich die Reise des Pakets in Echtzeit mitverfolgen, was auch etwa 15'000 Interessierte taten.

Dieses Werk wurde an der gleichnamigen Ausstellung im Frühjahr 2014 am Helmhauhaus Zürich präsentiert (Abb. 1). In einem der Ausstellungsräume des zweiten Obergeschosses standen hinten links zwei grosse Flachbildschirme auf dem Boden. Der rechte zeigte im Zeitraffer die Schnappschüsse der Paketkamera - meist rötliches oder schwarzes Flackern, dann plötzlich ein Stuhl, ein Bein, ein anderes Paket, schliesslich Julian Assange mit einer Nachricht an die Zuschauer, auf kleine Zettel gekritzelt. Synchron dazu auf dem linken waren in einer fettgedruckten, serifenlosen, knallroten Schrift vor monochromem von dunkelrot ins schwarz abfallendem Hintergrund die Twitternachrichten der Aktion zu lesen. Um dem Museumsbesucher etwas von der Live-Spannung zu vermitteln, ertönte aus dem Gewirr von Kabeln, Mediaplayern und Lautsprechern hinter den Bildschirmen ein dumpf pulsierender Soundtrack, der an dramatischen Höhepunkten der Geschichte an Intensität zulegte oder ganz aussetzte. Rechts in dem ganz reinweiss gestrichenen Raum hing von der Decke eine ein Meter breite Hochglanzfotopapierrolle. Aufgedruckt auf der sich schräg aus der Ecke zur Zimmermitte hin erstreckenden Bahn waren noch einmal alle aus dem Innern des Pakets aufgenommen Bilder zu sehen. Verkleinert und nahtlos aneinander gereiht, 26 Bilder pro Reihe, liess sich der Rhythmus der ganzen Handlung als linienartige Grafik auf einen Blick nachvollziehen.

Die Geschichte um das Werk lässt sich im Katalog (Abb. 2) nachlesen, der wie die Ausstellung den Titel des Werks *Ein Paket für Herrn Assange*<sup>62</sup> trägt. Ryser, der Journalist der Gruppe, beschreibt in abgehackten Sätzen das Geschehen hinter den Kulissen: Von vorbereitenden Diskussionen mit Assange per Skype, vom Nervenkitzel während der Aktion, dem Lampenfieber beim Verfassen von Twitternachrichten bis zum abschliessenden Käsefondue in der ecuadorianischen Botschaft. Der Umschlag des Katalogs zeigt ein schwarzweisses Röntgenbild des Pakets und auf der Innenseite den vom GPS aufgezeichneten Weg auf einer Karte von London. Dann folgt ohne Inhaltsverzeichnis oder Vorwort der Text, immer auf der rechten Buchseite im gegen den Bund gerückten Faltersatz, formatiert wie ein Gedicht. Auf der linken Seite im ebenfalls auf den Bund ausgerichteten randabfallenden Breitformat ist je eine von der Paketkamera gemachte Aufnahme in chronologischer Reihenfolge zu sehen, mit kleinem weissen Titel am unteren Bildrand, dazu Datum, Zeit und die live Twitternachricht.

Auf der Webseite<sup>63</sup> wiederholen sich die Informationen noch einmal: Eine kurze

<sup>62</sup>Siehe Katalog zur Ausstellung: Ryser 2014.

<sup>63</sup><https://www.bitnik.org/assange/> (10.02.2015).



Zusammenfassung des Werks, gefolgt vom Webcam Feed, dem Twitter Archiv, den GPS Signal Aufzeichnungen, Fotografien des Pakets (Aussenansicht und Röntgenbild) und einer Auswahl an Reaktionen von Presse und Twitter (Abb. 3).

Der Bezug dieses Kunstwerks zu Open Source mag auf den ersten Blick nicht offensichtlich sein, doch sind es mindestens vier Punkte. Erstens, ganz explizit, der fettgedruckte Copyleft Hinweis zuunterst auf der Webseite (Abb. 4), der auch auf allen anderen Seiten der Bitnik Homepage zu finden ist. Zweitens die Verwendung von *Cubietruck* für die Präsentation der Videos im Helmhaus (Abb. 5). *Cubietruck*<sup>64</sup> (auch *Cubieboard3*) ist Open Source Hardware, ein Linux basierter Mikrocomputer, entwickelt in China und hier als Mediaplayer eingesetzt. Drittens der inhaltliche und thematische Bezug des Werkes zu *Hacktivism*<sup>65</sup> und Free Culture,<sup>66</sup> die beide auf der Tradition der Free Software Bewegung fussen. Und viertens der Komplize Adnan Hadzi. Er promovierte 2012 an der Goldsmith University in London mit seiner Dissertation zu «FLOSSTV - Free, Libre, Open Source Software (FLOSS) within participatory TV hacking»<sup>67</sup> und ist demnach Spezialist für Open Source Fragen.

Finden sich entsprechend auch einige der Open Source Kernwerte wieder? Das Werk ist über die Webseite frei zugänglich und wurde Online öffentlich uraufgeführt. In Bezug auf Transparenz gibt die ausführliche Dokumentation auf der Webseite in Kombination mit dem Bericht im Katalog tiefen Einblick in die Aktion. Doch der grösste Teil der zur Adaption und Weiterentwicklung nötigen Informationen bleibt unzugänglich. Der Inhalt des Pakets lässt sich mit Hilfe des Röntgenbildes nur erahnen. Es fehlen die Schaltpläne der Elektronik und der Quellcode der Software, welche die Bilder schoss und auf die Webseite lud. Möchte ein Anderer das Prinzip weiterentwickeln oder gar selber ein solches Paket versenden, bleibt ihm das verwehrt. Dabei wäre dies doch genau der Kernwert des Open Source Gedankens. Es deutet alles darauf hin, dass die Implementation hier rein oberflächlich erfolgte. Die Dokumentation dient in erster Linie der Vermittlung des Mail Art Piece und ist nicht getrieben von Freiheitsgedanken und Transparenzbestreben. So erhält

<sup>64</sup>Siehe <http://www.cubieboard.org> (08.02.2015).

<sup>65</sup>Hacktivism, ein Zusammenzug von Hack und Activism, ist die Nutzung von Computer und Computer Netzwerken als Protestmittel zur Förderung von politischen Interessen wie der Redefreiheit, Menschenrechte und Informations-Ethik. Populäre Vertreter sind zum Beispiel die Gruppe Anonymous, die sich über die Webseite 4chan (<http://www.4chan.org>, 08.02.2015) organisiert, oder der deutsche Chaos Computer Club (<http://www.ccc.de>, 08.02.2015).

<sup>66</sup>Eine Bewegung, die sich gegen die als zu restriktiv empfundenen Urheberrechte einsetzt, um die Kultur des Free Content zu fördern. Siehe Gründer von Creative Commons, Lawrence Lessing in: Lessing 2004, S. 276-286.

<sup>67</sup>Hadziselimovic 2012.

das stolze Copyleft auf der Webseite einen schalen Beigeschmack - es ist lediglich ein Marketing Gag in einer gross angelegten Verkaufsstrategie, die es versteht, sich geschickt der neusten Entwicklungen der Netzkultur zu bedienen und die Medienwelle zu reiten. Und dies geschieht ungeachtet der Tatsache, dass es erst die Beiträge der unzähligen Volontäre der Open Source Bewegung sind, die das Werk überhaupt ermöglichen.

## Christian Waldvogel - Random Planet Production Machine (RPPM)

Der Zürcher Künstler, Programmierer und Architekt Christian Waldvogel (\*1971) arbeitet an der Grenze von Kunst und Wissenschaft und interessiert sich besonders für die Weltraumforschung. Im Austausch mit Wissenschaftlern aus verschiedensten Disziplinen und durch seine Tätigkeit im Topical Team Arts & Science (ETTAS) an der European Space Agency (ESA) findet er Inspiration für seine Werke. Die Ausstellung *unknown*<sup>68</sup>, die im ersten Obergeschoss des Helmhaus Zürich zeitgleich zu Bitnik's stattfand, zeigt seine Auseinandersetzung mit Zufall - oder Chaos, wie es die Kosmologen nennen.<sup>69</sup>

Aus dieser Auseinandersetzung heraus entstand das Werk *Random Planet Production Machine* (2013). Auf einer Grundfläche von etwa einem Quadratmeter steht ein zwei Meter hoher Alurahmen (Abb. 6). Er ist hohl und in zwei Bereiche unterteilt. Oben befindet sich eine verschiedenfarbig geschichtete Kerze mit mehreren brennenden Dochten. Der schmelzende Wachs tropft auf das darunter angebrachte trichterförmige Kupferblech und fällt schliesslich durchs Zentrum hindurch nach unten ins Herz der *Random Positioning Machine*. Diese besteht aus zwei ineinander liegenden Rahmen, die um die rechtwinklig zueinander angeordneten Achsen drehbar sind. Die Drehung der beiden Rahmen wird von einem Mikrocontroller gesteuert, der dafür sorgt, dass die aus den beiden Achsrotationen resultierende Bewegung im Rotationszentrum zufällig ist. Dort sammelt sich der erkaltende Wachs und formt mit der Zeit einen farbigen Klumpen, den «Planeten». Ist eine Kerze niedergebrannt, erscheint ein Museumsmitarbeiter in einem weissen Kittel. Er löst den Planeten aus seiner Geburtsstätte, putzt die Wachsreste weg, setzt eine neue Kerze ein, und mit dem Aufflammen des Zündholzes beginnt der Schöpfungsprozess von vorne.

<sup>68</sup>Siehe Katalog zur Ausstellung: Waldvogel 2014.

<sup>69</sup>Waldvogel 2014, S. 7-9.

Was im Ausstellungsraum kommentarlos vor sich geht, wird im Katalog *unknown*. Die *Ordnung der Zufälligkeit* genauer beschrieben. Das Kapitel *Orientierungslos*<sup>70</sup> ist der *Random Planet Production Machine* gewidmet. Es enthält ein kurzes Interview mit dem Gravitationsbiologen Jack van Look, dessen *Random Positioning Machine*<sup>71</sup> Auslöser war für Waldvogels Werk. Darauf folgen kommentarlos sechs Seiten kleingedruckte Baupläne (Abb. 7) der RPPM, ein Foto der speziell für die Maschine angefertigten Printplatte und anschliessend, auf fünf Seiten zusammengepresst, der vollständige Quellcode der Steuerungssoftware in Mikroschrift (Abb. 8). Das Ende des Unterkapitels bildet schliesslich ein kurzes Essay von Waldvogel, in welchem er ausgehend von Giordano Brunos (1548-1600)<sup>72</sup> Sicht des Universums, sein Werk kontextualisiert. Weiter hinten im Katalog finden sich zudem 24 Seiten Fotografien der RPPM: Eine Installationsansicht, Nahaufnahmen einiger Kerzen und Planeten, ein Zeitraffer der Maschine in Betrieb und am Ende zwei Bilder der Mini-Performance, dem Eingriff des Schöpfers beim Neustart der Planetenproduktion<sup>73</sup>.

Auf der Webseite zum Werk<sup>74</sup> (Abb. 9) finden sich dieselben Informationen noch einmal in verkürzter Form: Der Werkbescrieb, die Kontextualisierung mit Giordano Bruno sowie eine Auswahl von Bildern der Maschine und eine Fotografie der Printplatte. Die Konstruktionszeichnungen der Maschine und der Quellcode der Software hingegen fehlen.

Waldvogel setzt bei der Umsetzung seines Werkes auf Arduino, ein Open Source Mikrokontroller System. Das Design des Arduinio Board<sup>75</sup> untersteht der Creative Commons Attribution Share-Alike Lizenz (CC BY-SA). Diese ermöglicht eine kommerzielle Nutzung mit der Bedingung, dass die Quelle genannt (Attribution) und alles darauf basierende unter der gleichen Lizenz veröffentlicht wird.<sup>76</sup> Weiter

---

<sup>70</sup>Waldvogel 2014, S. 13-36.

<sup>71</sup>Mit Hilfe der Random Positioning Machine erforscht Look an der Freien Universität Amsterdam das Verhalten von Mikroorganismen in der Schwerelosigkeit. Durch stetige zufällige Richtungsänderungen wird der Probe vorgetäuscht, im schwerelosen Raum zu existieren. Siehe Waldvogel 2014, S. 15.

<sup>72</sup>Ein Dominikanermönch und Astrologe, dessen kosmologische Theorien weiter gingen als das Kopernikanische Modell. Er war der Ansicht, dass die Sterne entfernte Sonnen sind, umgeben von eigenen Exoplaneten und somit vermutlich andere erdähnliche Planeten existieren. Zudem war er überzeugt, dass das Universum unendlich ist und deshalb keinen zentrischen Himmelskörper hat. Er wurde 1600 wegen Ketzerei auf dem Scheiterhaufen verbrannt.

<sup>73</sup>Waldvogel 2014, S. 112-135.

<sup>74</sup><http://www.waldvogel.com/narratives/RPPM> (10.02.2015).

<sup>75</sup>Damit ist das Layout der Printplatte und das Schema des elektrischen Signalverlaufs gemeint.

<sup>76</sup>Siehe <http://creativecommons.org/licenses/by-sa/2.5/> (10.02.2015).

unterliegt die Arduino Programmierumgebung<sup>77</sup> der GNU General Public Lizenz. Diese softwarespezifische Lizenz ist praktisch identisch mit der CC BY-SA, verlangt aber ausdrücklich, dass der Quellcode veröffentlicht wird.

Auch in diesem Werk finden sich einige der Open Source Kernwerte wieder. Wie schon bei Bitnik erfreut sich die *Random Planet Production Machine* grosser Transparenz. Katalog und Webseite dokumentieren die Arbeit und ihren Kontext ausführlich mit Text und Bild. Waldvogel geht sogar noch einen Schritt weiter, denn er veröffentlicht auch die Bauanleitung zu seiner Maschine und den Quellcode der Software. Er ist sich der Open Source Thematik durchaus bewusst und ist bestrebt, die Lizenzvereinbarungen zu erfüllen. Allerdings scheinen sich alle seine Bestrebungen lediglich auf das «Erfüllen» zu beschränken. Denn die technischen Zeichnungen sind so klein, dass sie zum Nachbau der Maschine höchstens Ansatzpunkte liefern. Das Layout und die Fotografie der Printplatte hilft ohne Schema und Komponentenliste kaum weiter. Auch der Quellcode in gedruckter Form ist absolut nutzlos – in der Zeit, die fürs fehlerfreie Abtippen aufgewendet werden müsste, lässt sich das Programm neu entwickeln<sup>78</sup>. Waldvogel missversteht offenbar, worum es bei Open Source wirklich geht. Das Einhalten von Vertrags- und Lizenzvereinbarungen steht eben gerade nicht im Vordergrund, wie dies bei proprietären Produkten der Fall ist. Die Lizenzen sind dazu da, die Freiheit der Soft- und Hardware zu schützen, damit Andere davon lernen und diese optimieren und weiterentwickeln können.

---

<sup>77</sup>Die Arduino IDE (Integrated Development Environment) basiert auf dem am MIT entwickelten Processing, einer Programmierumgebung für gestalterisch-künstlerische Arbeiten. Siehe <http://www.processing.org> (10.02.2015).

<sup>78</sup>Dass die Bauanleitung und der Quellcode lediglich mit dem Katalog erhältlich sind, ist indes kein Verstoß gegen die Lizenz. Frei in Open Source steht nicht für gratis, sondern bedeutet Freiheit im Sinne der uneingeschränkten Weiterverwendung.

## Copying is an Act of Love

Open Source steht, wie bereits dargelegt, für eine Arbeitshaltung basierend auf den Kernwerten Freiheit, Offenheit, Transparenz, Teilen und Kollaboration: der Freiheit aller, das Geschaffene nach Belieben zu verändern und anzupassen; der Offenheit der «Baupläne», für alle uneingeschränkt zugänglich; der Transparenz aller Quellen und deren uneingeschränkter Nennung; des Teilens des Geschaffenen, der Baupläne und der Quellen zugunsten der Weiterentwicklung und der Kollaboration als ultimativem Ziel all dieser Bestrebungen.

Open Source ist ein Gegenentwurf zum Urheberrecht.<sup>79</sup> Statt der Rechte des Urhebers schützen die Open Source Lizenzen die Freiheit des Geschaffenen. Denn nur, wenn das Geschaffene absolut frei ist, das heisst, jeder das gleiche Recht hat, damit zu machen, was er will, ist eine offene und locker gekoppelte Zusammenarbeit möglich. Dies ist absolut zentral, denn alle Beteiligten profitieren schliesslich von dieser Zusammenarbeit. Open Source heisst also, auf Individualrechte zu Gunsten des Kollektivs zu verzichten, im Wissen darum, dass der resultierende Gewinn aus der Kollaboration grösser ist als das Bestehen auf die eigenen Rechte.

Bezogen auf die Digitalen Künste beschränkt sich das Open Source Prinzip demzufolge nicht auf die Technik, sondern betrifft darüber hinaus den gesamten Prozess der künstlerischen Produktion. Es ist ein Aufruf zum Dialog, zur Kultivierung von Zusammenarbeit. Mit der Bildung von offenen Interessengemeinschaften rund um Kunstwerke könnten diese weiterentwickelt und im besten Fall zum Nährboden von neuen Ideen werden. Zumindest in der Programmierung funktioniert dieses Prinzip bereits.

Die !Mediengruppe Bitnik sowie Christian Waldvogel verpassen diese Chance, zumindest teilweise. Bitnik versucht, aus dem Hype um Open Source und dem Free Software Movement einen Gewinn für sich selbst zu schlagen. Sie geben nur so viele Informationen preis wie nötig, um im Medienhype mitschwimmen zu können.

---

<sup>79</sup>Bezüglich Urheberrecht und *Open Approaches* in der Kunst siehe Senn 2007; Vgl. auch Stalder 2010. Die Künstlerin Cornelia Sollfrank hat das Urheberrecht mit der Arbeit *net.art Generator* (2004) zum Thema ihres künstlerischen Schaffens gemacht, siehe Sollfrank 2001.

Zwar kann man über die sozialen Medien mit Bitnik in Kontakt treten, doch bleibt ihr Werk eine Blackbox. Eine Kollaboration im Sinne von gegenseitigem Austausch ist unter diesen Bedingungen nicht möglich.

Waldvogel, so scheint es, hat sich mit der Thematik eingehender befasst. Er ist sich der Open Source Lizenzvereinbarung bewusst und weiss, dass die Share Alike Klausel ihn dazu auffordert, seine Weiterentwicklungen an die Entwicklergemeinschaft zurück zu geben. Schliesslich verdankt er sein Werk zu einem grossen Teil denen, die das zugrundeliegende Mikrokontrollersystem entwickelt haben. Doch auch Waldvogel liegt wenig daran, in einen Dialog zu treten. Er veröffentlicht zwar die Baupläne und den Quellcode, aber in einem unleserlichen und untauglichen Format.

Was hält Waldvogel und die Bitnik-Gruppe davon ab, sich ganz auf Open Source einzulassen? Ist es die Angst davor, kopiert zu werden? Fürchten sie sich etwa vor finanziellen Einbussen?

Der New Yorker Künstler Golan Levin (\*1972) hat diese Angst überwunden. Alle seine Werke sind ausführlich dokumentiert und wie bei Bitnik mit Socialmedia-Anbindung Online zugänglich. Er veröffentlicht zudem wie Waldvogel seine Baupläne, allerdings in einer Form, die Weiterverarbeitung ermöglicht. Ein Beispiel dazu ist die Arbeit *Augmented Hand Series* (2014), eine schwarze Kiste, welche die in einen Schlitz hinein gestreckte Hand des Betrachters auf dem angebrachten Monitor in Echtzeit verfremdet (Abb. 10). Auf der Webseite<sup>80</sup> finden sich eine ausführliche Dokumentation mit Fotografien der Installation sowie Videos von partizipierenden Besuchern, eine Kontextualisierung der Arbeit und ein werkspezifisches Pressearchiv. Darauf folgt der Abschnitt «Credits & Acknowledgments» (Abb. 11). Hier listet er transparent auf, welche Personen an der Arbeit beteiligt waren, wer das Projekt finanziert hat, wo die Baupläne<sup>81</sup> zu finden sind und auf welchen Libraries und Frameworks<sup>82</sup> die Software aufgebaut wurde.

Levins Offenheit und Transparenz, vermutlich inspiriert durch das Studium in Kunst und Design am MIT, dem Geburtsort von Open Source, hat ihm nicht geschadet, im Gegenteil: seit 1999 ist er international regelmässig an Ausstellungen vertreten, hält Vorlesungen,<sup>83</sup> leitet Workshops, publiziert Bücher und gibt nebenbei Interviews.<sup>84</sup>

<sup>80</sup><http://www.flong.com/projects/augmented-hand-series> (11.02.2015).

<sup>81</sup>Der Quellcode kann unter [https://github.com/CreativeInquiry/digital\\_art\\_2014](https://github.com/CreativeInquiry/digital_art_2014) (11.02.2015) eingesehen werden.

<sup>82</sup>Libraries und Frameworks sind eine Form von Code Bausteinen, welche die Programmierarbeit erleichtern.

<sup>83</sup>Zum Beispiel an der Konferenz für Technology, Entertainment und Design (TED) im Februar 2009. Siehe Video unter [http://www.ted.com/talks/golan\\_levin\\_ted2009](http://www.ted.com/talks/golan_levin_ted2009) (11.02.2015).

<sup>84</sup>Hope und Ryan 2014, S. 144.

Es scheint, als ob Levin sich in einem ähnlichen Prozess befindet, der zur Zeit auch die Softwarebranche auf den Kopf stellt. Dort wird nämlich Code an sich immer mehr als Rohstoff verstanden, der in sich nur geringen Wert hat. Die Wertschöpfung erfolgt nicht aus dem Verkauf des kompilierten Quellcodes, der mit Open Source ja öffentlich zugänglich ist, sondern durch die dazu gelieferten Dienstleistungen. Man spricht hierbei von *Software as a Service* (SAAS).

Zeichnet sich in der zeitgenössischen Kunst etwa ein ähnliches Umdenken ab? Ansätze dazu wären bereits bei Marcel Duchamp (1887 - 1968) zu finden, der mit *Fountain* (1917) ein billiges Urinal zum Kunstwerk erklärte. Seine Leistung war die Idee an sich. Die Materialität des Kunstwerks hatte kaum einen Wert. In den 60er Jahren wurde dieser Gedanke von Fluxus und der Konzeptkunst weitergedacht und er wird im Zusammenhang mit Open Source möglicherweise wieder aktuell. Bestünde nämlich die künstlerische Leistung vor allem in der Schöpfung von Ideen, wären die «Baupläne», die diese Idee beschreiben, im weitesten Sinne der eigentliche Quellcode der Kunst, und das Kunstwerk demnach dessen kompilierte Form.

Somit wäre der Künstler, ähnlich dem Programmierer, ein Spezialist für die Ideenentwicklung und den Abstraktionsprozess von Idee zu Bauplan. Das würde eine stark an den Künstler gekoppelte Dienstleistung ergeben, die wertvoller und einzigartiger wäre als das Endprodukt. Das Kunstwerk würde zum Aushängeschild für den Künstler, der nunmehr von seinem Ruf als Ideenerbringer leben und nicht nur Kunstwerke, sondern auch die Dienstleistungen darum herum liefern würde - Ausstellungen kuratieren, Vorträge halten, Workshops leiten, Bücher publizieren, Interviews geben und so weiter. Das wäre ein Geschäftsmodell, das noch über die *Factory* von Andy Warhol (1928 - 1987) oder die Vermarktungsstrategien von Jeff Koons (\*1955) hinaus ginge: Kunst als Dienstleistung - *Art as a Service*, AAAS würde ich es nennen.

Ein solches Modell hätte gravierende Auswirkungen auf mein eigenes künstlerisches Schaffen. Indem ich mir bewusst mache, dass nicht ich meine Werke, sondern meine Werke mich vermarkten, kann ich mich von der Angst lösen, kopiert zu werden. Meine Baupläne lege ich nun gerne offen, ja kann andere sogar dazu auffordern, bei mir abzuschauen: Einerseits aus Dankbarkeit und Respekt gegenüber all denen, die vor mir selbstlos Zeit und Energie in Projekte investierten, ohne die ich meine Kunstwerke niemals realisieren könnte, andererseits aber auch im Wissen darum, dass eigentlich nur die Dinge nachgebildet werden, die ansprechen, qualitativ überzeugen und inspirieren. So muss ich mich vor dem kopiert Werden nicht mehr fürchten. Vielmehr wird das Kopieren für mich nun zu einem Gütesiegel,

einem *act of love*<sup>85</sup> an mich, den Autor.

---

<sup>85</sup>Die Bloggerin Nina Paley, bekannt geworden durch ihren Animationsfilm *Sita Sings the Blues* (siehe Anhang unter Animationsfilm), setzt sich mit der Website <http://www.copyheart.org> (18.02.2015) dafür ein, das Copyright-Symbol mit einem Herz und das Copyright-Statement mit «Copying is an act of love. Please copy.» zu ersetzen, um sich so gänzlich vom Urheberrecht zu lösen.



## Literaturverzeichnis

- Casserly, Catherin und Joi Ito (2011). *Creative Commons. The power of open*. Cambridge MA: Creative Commons Corporation.
- Deterding, Sebastian (2007). *Into the Great Wilde Open. Open Source jenseits von Software*. Bonn. URL: <http://www.bpb.de/gesellschaft/medien/opensource/63921/into-the-great-wide-open?p=all> (besucht am 18.12.2014).
- Diener, Andrea (2013). «Kunsthacks. Der Ausgang ist immer ungewiss». In: *Frankfurter Allgemeine FAZ.NET* Bd. 200. URL: <http://www.faz.net/kunsthacks-der-ausgang-ist-immer-ungewiss-12732369.html> (besucht am 17.02.2015).
- Eco, Umberto (1973). *Das offene Kunstwerk*. Hrsg. von Günter Memmert. Frankfurt a.M.: Suhrkamp Verlag.
- Graham, Beryl (2007). «Redefining Digital Art. Disrupting Borders». In: *Theorizing Digital Cultural Heritage: A Critical Discourse*. Hrsg. von Fiona Cameron und Sarah Kenderdine. Cambridge MA: MIT Press, S. 93–112.
- Grassmuck, Volker (2004). *Freie Software. Zwischen Privat- und Gemeineigentum*. Bonn: Bundeszentrale für politische Bildung.
- Hadziselimovic, Adna (2012). «FLOSSTV. Free, Libre, Open Source Software (FLOSS) within participatory TV hacking Media and Arts Practices». Diss. London: Goldsmiths University. URL: [http://research.gold.ac.uk/6922/1/MED\\_thesis\\_Hadziselimovic\\_2012.pdf](http://research.gold.ac.uk/6922/1/MED_thesis_Hadziselimovic_2012.pdf) (besucht am 10.12.2014).
- Hamerly, Jim, Tom Paquin und Susan Walton (1999). «Freeing the Source. The Story of Mozilla». In: *Opensources: Voices from the Open Source Revolution*. Hrsg. von Chris DiBona, Sam Ockman und Mark Stone. Sebastopol CA: O'Reilly Media Inc., S. 91–95.
- Han, Byung-Chul (2012). «Transparent ist nur das Tote». In: *Zeit Online* 3. URL: <http://www.zeit.de/2012/03/Transparenzgesellschaft> (besucht am 27.02.2015).
- (2013). *Transparenzgesellschaft*. Berlin: Matthes & Seitz.

- Himanen, Pekka (2001). *The Hacker Ethic. And the Spirit of the Information Age*. New York: Random House Trade Paperbacks.
- Hope, Cat und John Ryan (2014). *Digital Arts. An introduction to new media*. New York: Bloomsbury Academic.
- Kleijn, Alexandra (2007). «EU-Studie: Open Source ist gut für die Wirtschaft». In: *TelepolisMagazin der Netzkultur*. URL: <http://www.heise.de/open/artikel/EU-Studie-Open-Source-ist-gut-fuer-die-Wirtschaft-222037.html> (besucht am 25. 02. 2015).
- Lessing, Lawrence (2004). *Free Culture. How big media uses technology and the law to lock down culture and control creativity*. New York: The Penguin Press.
- Levy, Steven (1984). *Hackers: Heroes of the Computer Revolution*. New York: Anchor Press Doubleday.
- McKenzie, Wark (2004). *A Hacker Manifesto*. Cambridge MA: Harvard University Press.
- Paul, Christiane (2003). *Digital Art*. London: Thames & Hudson.
- Raymond, Eric (1996). *The new Hacker's Dictionary*. 3. Aufl. Cambridge MA: MIT Press.
- (1999). «A Brief History of Hackerdom». In: *Opensources. Voices from the Open Source Revolution*. Hrsg. von Chris DiBona, Sam Ockman und Mark Stone. Sebastopol CA: O'Reilly Media Inc., S. 16–20.
  - (2001). *The Cathedral and the Bazaar*. Hrsg. von Tim O'Reilly. 2. Aufl. Sebastopol CA: O'Reilly Media Inc.
- Ryser, Daniel (2014). *Ein Paket für Herrn Assange*. Ausst.-Kat. Zürich, Helmhaus Zürich, 14.02. – 06.04.2014. Basel.
- Senn, Mischa (2007). «Autorschaft und Open Approaches. Die Kulturrechtliche Perspektive». In: *Autorschaft in den Künsten. Konzepte - Praktiken - Medien*. Hrsg. von Corina Caduff. Zürich: Zürcher Hochschule der Künste, S. 240–251.
- Sollfrank, Cornelia (2001). «Copyright 2004». In: *net.art Generator. Programmierte Verführung*. Hrsg. von Esther Kiener. Nürnberg: Institut für moderne Kunst, S. 44–56.
- Stalder, Felix (2010). «Kritische Strategien zu Kunst und Urheberrecht». In: *Kunstforum International Bd. 200*, S. 111–117.

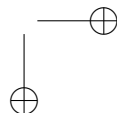
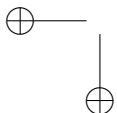
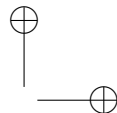
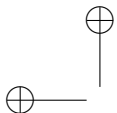
Stallman, Richard (2010). *Free Software - Free Society. Selected Essays of Richard M. Stallman*. 2. Aufl. Boston: Free Software Foundation Inc. URL: <http://www.gnu.org/doc/fsfs-ii-2.pdf>.

Taylor, Grant (2014). *When the machine made art. The troubled history of computer art*. Hrsg. von Francisco Ricardo. New York: Bloomsbury.

Torvalds, Linus (1999). «The Linux Edge». In: *Opensources. Voices from the Open Source Revolution*. Hrsg. von Chris DiBona, Sam Ockman und Mark Stone. Sebastopol CA: O'Reilly Media Inc., S. 51–54.

Waldvogel, Christian (2014). *Unknown. Die Ordnung der Zufälligkeit*. Ausst.-Kat. Zürich, Helmhaus Zürich, 14.02.-06.04.2014. Zürich.

Wands, Bruce (2006). *Arts of the Digital Age*. London: Thames & Hudson.



## Abbildungen



Abb. 1: !Mediengruppe Bitnik, Ein Paket für Herrn Assange, 2014, Helmhaus Zürich, Installationsansicht, Fotografie 2014 ([https://www.bitnik.org/media/news/exhibition\\_helmhaus/thumb/bitnik\\_helmhaus\\_028\\_1600\\_1280.jpg](https://www.bitnik.org/media/news/exhibition_helmhaus/thumb/bitnik_helmhaus_028_1600_1280.jpg), 17.02.2015).



Abb. 2: !Mediengruppe Bitnik, Ein Paket für Herrn Assange, 2014, Helmhaus Zürich, Fotografie des Katalogs zur Ausstellung (Ryser 2014, S. 14-15).

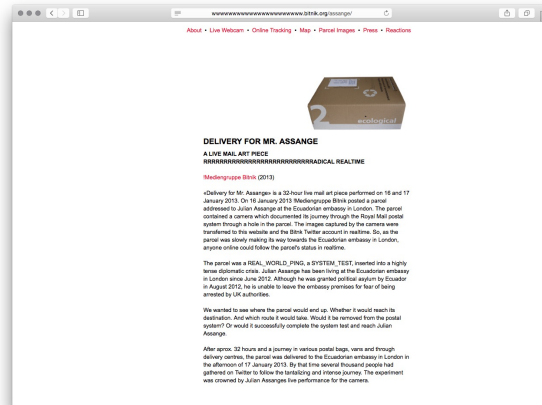


Abb. 3: !Mediengruppe Bitnik, Ein Paket für Herrn Assange, 2014, Screenshot der Webseite zum Werk (<https://www.bitnik.org/assange/>, 17.02.2015).

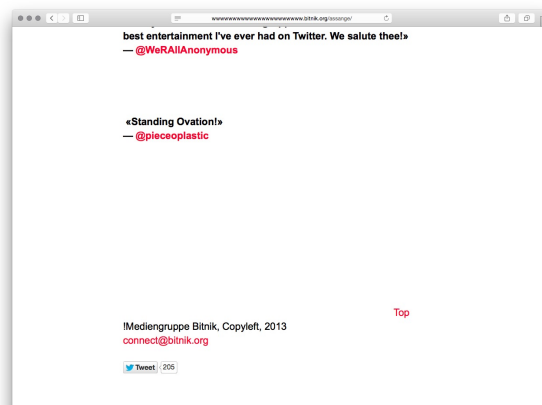


Abb. 4: !Mediengruppe Bitnik, Ein Paket für Herrn Assange, 2014, Detailaufnahme der Webseite zum Werk, Screenshot (<https://www.bitnik.org/assange/>, 17.02.2015).

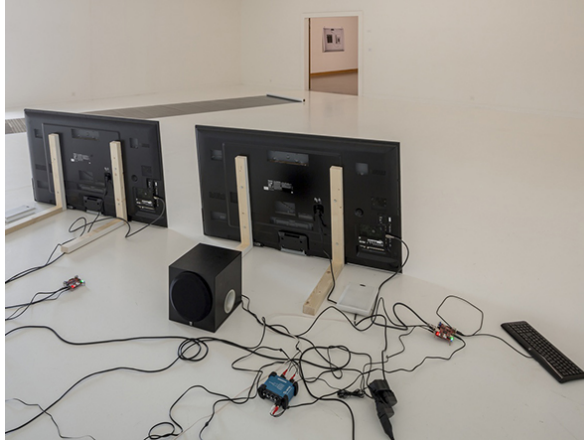


Abb. 5: !Mediengruppe Bitnik, Ein Paket für Herrn Assange, 2014, Helmhaus Zürich, Cubieboard links neben der Tastatur, Fotografie 2014 ([https://www.bitnik.org/media/news/exhibition\\_helmhaus/thumb/bitnik\\_helmhaus\\_034\\_1600\\_1280.jpg](https://www.bitnik.org/media/news/exhibition_helmhaus/thumb/bitnik_helmhaus_034_1600_1280.jpg), 17.02.2015).



Abb. 6: Christian Waldvogel, Random Planet Production Machine, 2014, Helmhaus Zürich, Installationsansicht, Fotografie 2014 ([http://waldvogel.com/cms/files/projects/helmhaus-zuerich/helmhaus\\_0005\\_MBCW.jpg](http://waldvogel.com/cms/files/projects/helmhaus-zuerich/helmhaus_0005_MBCW.jpg), 17.02.2015).



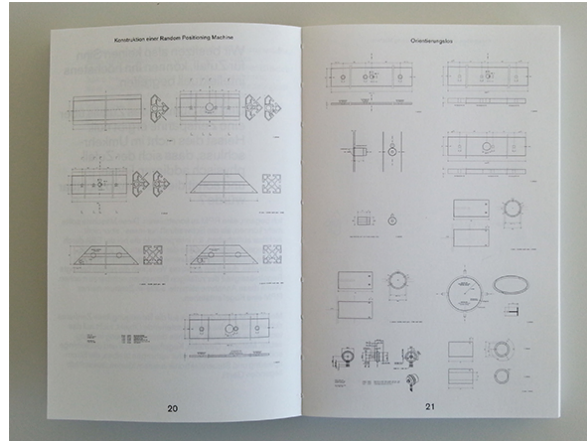


Abb. 7: Christian Waldvogel, Random Planet Production Machine, 2014, Helmhaus Zürich, Baupläne der RPPM, Fotografie des Katalogs zur Ausstellung (Waldvogel 2014, S. 20-21).

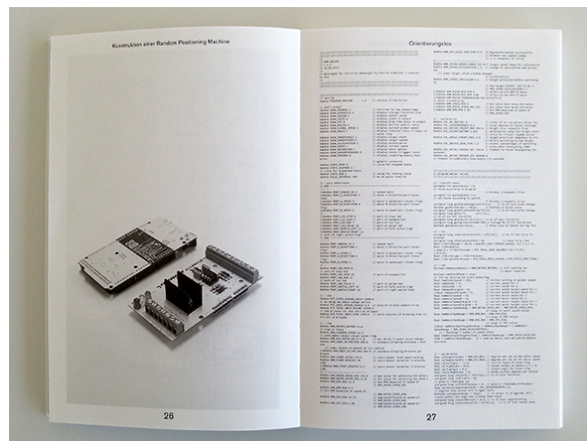


Abb. 8: Christian Waldvogel, Random Planet Production Machine, 2014, Helmhaus Zürich, links die Leiterplatte, rechts ein Ausschnitt des Quellcodes der Steuerungssoftware, Fotografie des Katalogs zur Ausstellung (Waldvogel 2014, S. 26-27).

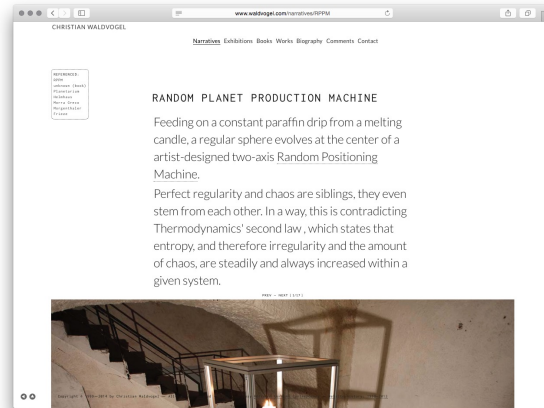


Abb. 9: Christian Waldvogel, Random Planet Production Machine, 2014, Screenshot der Webseite zum Werk (<http://www.waldvogel.com/narratives/RPPM>, 17.02.2015).



Abb. 10: Golan Levin, Augmented Hand Series, 2014, Installationsansicht, Fotografie 2014 (<http://www.flong.com/storage/images/projects/augmented-hand-series.jpg>, 17.02.2015).

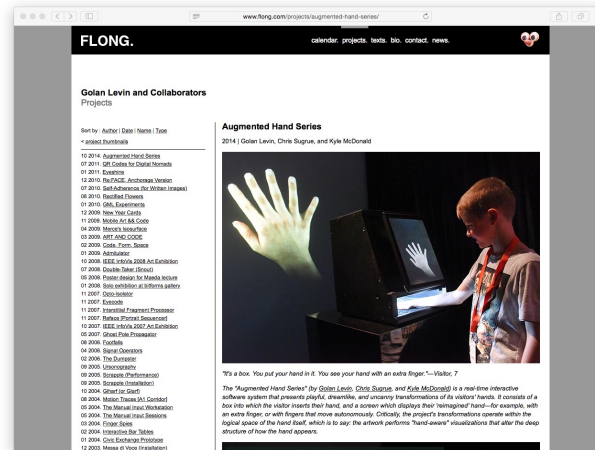
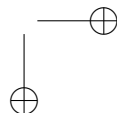
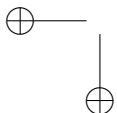
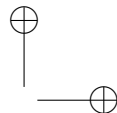
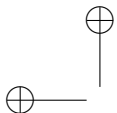
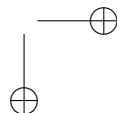
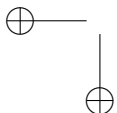
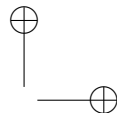
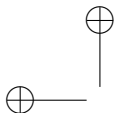


Abb. 11: Golan Levin, Augmented Hand Series, 2014, Detailaufnahme der Webseite zum Werk, Screenshot (<http://www.flong.com/projects/augmented-hand-series/>, 17.02.2015).



# Anhang



## Liste der erwähnten Open Source Projekte

Nachfolgend die Liste der in der Einleitung erwähnten Open Source Projekte. Die Reihenfolge entspricht derjenigen der Aufzählung. Die Auswahl der Projekte ist zufällig und nicht repräsentativ. Links zuletzt aufgerufen am 10.12.2014.

- Framework: Twitter Bootstrap (<http://getbootstrap.com/>)
- Webbrowser: Chromium (<http://www.chromium.org/>)
- Betriebssystem: Lubuntu Linux (<http://lubuntu.net/>)
- Mikrocontroller: Arduino (<http://www.arduino.cc/>)
- Programmierumgebung: Arduino IDE (<http://www.arduino.cc/> basierend auf Processing <http://processing.org/>)
- Computer System: Raspberry Pi (<http://www.raspberrypi.org/>)
- Strickautomaten: OpenKnit <http://openknit.org/build/>
- 3D-Drucker: RepRap (<http://reprap.org/>)
- Lexika: Wikipedia (<http://wikipedia.org>)
- Handprothesen: Open Hand Project (<http://www.openhandproject.org/>)
- Humanoide Roboter: Poppy Humanoid (<https://www.poppy-project.org/creatures/poppy-humanoid/>)
- Video Games: Sintel the Game (<http://sintelgame.org/>)
- Gamekonsolen: OUYA (<https://www.ouya.tv/>)
- CNC-Fräsmaschinen: Shapeoko (<http://www.shapeoko.com/>)
- Häuser: Wiki House (<http://www.wikihouse.cc/>)
- Möbelstücke: z.B. der Sketch Chair (<http://www.sketchchair.cc/>)

- Zeichensätze: Die Openfont Library (<http://openfontlibrary.org/>)
- Sciencefiction Novelle: Die Triologie von Stephen J. Sweeney *The Battle for the Solar System* (<http://www.battleforthesolarsystem.com/downloads/#knightsFirst>)
- Bier: Free Beer (<http://freebeer.org/blog/>)
- Pop-Musik: Das Album *Pulse of the Earth* von Hungry Lucy (<http://music.hungrylucy.com/album/pulse-of-the-earth>)
- Comic Strip: Mimi and Deunice (<http://mimiandeunice.com/>)
- Animationsfilm: Sita sings the blues (<http://www.sitasingingtheblues.com/>)
- Spielfilm: Valkaama (<http://www.valkaama.com/>)
- Filmfestival: Das Barcelona Creative Commons Film Festival, das 2010 erstmals unter dem Slogan *Copy this Festival* stattfand, erfreut sich seither jährlich steigender Besucherzahlen.
- Galerie: Open Source Galery in Brooklyn USA (<http://open-source-gallery.org/>)