

DISCORDIA CONCORDS: WWW.JODI.ORG

FLORIAN CRAMER

JODI NET

Wäre der heutige Kunstbetrieb nicht auf Zurschaustellungen fixiert – ob von Bildopulenz oder politischer Korrektheit – und vor allem nicht auf ausstell- und verkaufbare Objekte, wären jodi vermutlich anerkannt als die wichtigsten zeitgenössischen Künstler. Gewiß liefe solch eine Kanonisierung der ästhetischen Subversion von www.jodi.org zuwider; auch spricht gegen sie der Aufwand, mit dem jodi ihre Arbeit als Teil eines größeren Netzwerks präsentieren und schneller Konsumierbarkeit entziehen. Dies ist die eine Seite der Medaille; auf der anderen steht jodis Sorgfalt, ihrer Kunst vorab einen selbstgesteckten Kontext zu verpassen. Deutet man sie als Kontrollstrategie, so verkomplizieren sich die Gegensätze von Subversion und Kanonisierung.

Es ist, wie der Sinologe und Computer-Dichter John Cayley schreibt, „schwierig, im Sinne einer mehr oder weniger konventionellen Kunstkritik etwas Schnelles und Gewisses über eine Website zu schreiben, die selten dieselbe geblieben ist, wenn man sie erneut besucht.“¹ Als dieser Text geschrieben wurde, leitete <http://www.jodi.org> automatisch auf die Download-Seite von „Untitled Game“ um. Der Rest der Website ließ sich nur über Suchmaschinen finden, die auf Kritiker-Texte über [jodi.org](http://www.jodi.org) verwiesen, welche wiederum die versteckten Sektionen von [jodi.org](http://www.jodi.org) referenzierten – oder man wick gleich auf die netzkünstlerische Raubkopie von www.jodi.org auf <http://www.0100101110101101.org/home/jodi.org> aus. So unterminieren jodi die corporate identity ihrer eingängigen Netzadresse und verlangen Lesern Netz-Arbeit ab; eine Entzugsstrategie, die auch an die Schriften spätantiker Gnostiker erinnert, die jahrhundertlang nur als Zitate aus den Widerlegungs-Traktaten christlicher Theologen überlebten.

Seit dem 16. Jahrhundert sind Gedichte, die nach kombinatorisch-algorithmischen Regeln ihre Wörter vertauschen, als „Proteusverse“ bekannt, denn wie der griechische Meeresherr wechseln sie ständig ihre

Date: 10.7.2002.

¹[Cay01]

Erscheinung. www.jodi.org verlagert das proteische Prinzip von den internen Daten des Kunstwerks auf die Metadaten seiner Adressierung und Vernetzung. Diese proteische Logik regelt jedoch nicht nur den Zugang zur Website, sondern auch die Durch- und Ausgänge. Zum Beispiel weigert sich www.jodi.org, sich auf sich selbst zu begrenzen, indem es topographische Verbindungen zu diversen anderen Netzkunst-Servern zeichnet („map“, <http://map.jodi.org>), Web-Querverweise auflistet („CC“, <http://www.jodi.org/100cc/baklava/index.html>) und verborgene Sprunganweisungen auf interne Seiten mit verborgenen Sprunganweisungen auf fremde Seiten vermischt.

Zugleich konterkarieren diese Kartographierungen und Festschreibungen des Kontexts jodis proteisches Entgleiten einer festen Identität; eine Identität, die sich unweigerlich auch im durchschlagenden Einfluß ihrer Arbeit auf andere Netzkünstler festschreibt. Da jodi für die code-experimentelle Netzkunst, wie sie sich seit 1997 im E-Mail-Forum „7-11“ (<http://www.7-11.org>)² herausbildete, so wichtig waren wie Cézanne für den Kubismus oder Malewitsch für den Konstruktivismus, führt an ihrer Kanonisierung kein Weg vorbei.

JODI ART

Bei der Abfassung dieses Textes beförderte eine „Google“-Suche nach der Stichwortkombination „jodi.org“ und „noise“ 228 Web-Seiten zutage. Worin jedoch unterscheidet sich das Rauschen von www.jodi.org vom Rauschen und Zufall in den Avantgarde-Künsten der vorigen hundert Jahre, vom Pointillismus und Dada bis zu John Cage und Fluxus?³ – Der Unterschied ist sowohl ein medialer, als auch ein rhetorischer. In Dada-Dichtung, Hans Arps Zufallsbildern und John Cages Zufallsmusik strukturiert der Zufall das Werk immanent, nicht aber seine mediale Übertragung. www.jodi.org hingegen chaotisiert seine Datenübertragung nicht nur durch instabile Adressierungen, sondern liest und verhält sich auch, als bestünde es aus eigentlich intakten, nur durch Übertragungsfehler oder Computerabstürze beeinträchtigten Daten. Tatsächlich sind diese Übertragungsstörungen in den Daten selbst simuliert. Im Gegensatz zu Nam June Paiks störmanipulierten Fernsehern aus den 1960er Jahren spielen Jodis Störungen nicht in der Hardware und mit oft unvorhersehbaren Resultaten ab, sondern sind clevere Vortäuschungen von Unvorhersehbarkeit auf Software-Ebene.

²siehe Frederic Madres Beitrag zu diesem Katalog

³Thomas Skowronek, der an der Humboldt Universität Berlin zur Zeit eine Magisterarbeit über Rauschen in der Netzkunst schreibt, brachte diese Frage auf.

Denn das scheinbar chaotischer Verhalten der jodi-Website ist eine trompe-l'oeil-Simulation auf der Grundlage absolut linearer und deterministischer Codes: Texte in der Formatierungssprache HTML, einfache Bildanimationen im GIF-Format und sparsam eingestreute Zeilen in der Programmiersprache Javascript.⁴ Und während die indeterministischen Poetiken von Cage und Fluxus Zufall und Chaos für Mittel radikaler Freiheit hielten, ist ihre Implikation in jodis Kunst weitaus ambivalenter. Zwar inspirieren und befreien die scheinbaren Störungen die Imagination des Betrachters, zugleich aber locken sie ihn in Täuschungen, Irrwege und Sackgassen hinein. An die Stelle der naiven cagianischen Zufallsontologie tritt eine trickreiche Rhetorik simultaner Anarchie und Gefangennahme, ein neobarockes Concerto und eine discordia concors eines disziplinierten Schein-Chaos und einer chaotischen Schein-Disziplin.

JODI NET ART

Jodi sind bekannt für ihre Kunst der Disruptionen und Fehlcodierungen und dafür, jenseits aller technonaiven „Multimedia“-Kunst einen ästhetischen Blick auf Computer als mit sich selbst beschäftigte Generatoren kontingenter Datenströme geöffnet zu haben. Dennoch handelt ihre Kunst nicht simpel von einer (imaginären) Wahrheit unterhalb trughafter Software-Oberflächen. Was in Jodis Arbeiten als Code erscheint – zum Beispiel die kontingenten Nummerntabellen im Unterverzeichnis „Automatic Rain System“ von „betalab“ <http://www.wwwww.jodi.org/betalab/rain/> – ist Code, der auf anderen Code verweist, aber auch: Code, der nicht ist, was er zu sein scheint. Jodis Codes dienen typischerweise der Simulation von anderen Codes, Simulationen von Algorithmen durch Graphikanimationen etwa oder Simulationen von Bildschirmgraphik durch blinkenden Text. Am 22. Oktober 2001 schickten jodi einen umfangreichen Text auf die internationale E-Mail-Liste „Nettime“, der ein Kommentar auf damalige Krisenkriege und ihre Diskussion in netzkulturellen Foren zu sein schien:

```
$cd ug/models/soldier3
$origin 0 -6 24
$base base
$skin skin
```

```
$frame soldiery
$frame soldierd
```

⁴Das Phänomen statischer „Genotexte“ versus mobiler „Phänotexte“ in digitaler Kunst hat, mit ausdrücklichem Bezug auf Jodi, zuerst und ausführlicher als hier Inke Arns in ihrem Aufsatz [Arn01] beschrieben

```

/*
*/

void() army_fire;

void() army_stand1      =[      $soldierc,      army_stand2      ]
{ai_stand();};
void() army_stand2      =[      $soldierc,      army_stand3      ]
{ai_stand();};
void() army_stand3      =[      $soldierc,      army_stand4      ]
{ai_stand();};
[...]
```

Der Text (der hier nicht vollständig reproduziert, aber in seiner Originalfassung unter <http://amsterdam.nettime.org/Lists-Archives/nettime-1-0110/msg00120.html> gelesen werden kann) war zuallererst ein beeindruckendes Stück konkrete Poesie. Die militärische Ordnung, die sich ihm linguistisch und typographisch eingeschrieben hatte, ließ ihn wie eine obsessiv codierte Inventarliste oder strategische Planskizze erscheinen. Leser, die neben Englisch auch Programmiersprachen beherrschten, konnten außerdem bemerken, daß das Gedicht ein funktionaler Quellcode in der Programmiersprache „C“ ist. Tatsächlich handelte es sich um einen Teil des Quellcodes von jodis „Untitled Game“, der wiederum auf dem Quellcode des kommerziellen Computerspiels „Quake“ basiert. Indem jodis E-Mail seine Herkunft und Funktion unterschlug, machte sie zugleich die ästhetischen und politischen Subtexte scheinbar neutraler technischer Befehlsabfolgen sichtbar.

Im Jahr 1924 wies Tristan Tzara seine Leser an, „ein dadaistisches Gedicht zu machen“, indem sie die Wörter eines Zeitungsartikel ausschneiden, in einer Tüte mischen und „dann einen Schnipsel nach dem anderen [...] in der Reihenfolge, in der sie aus der Tüte gekommen sind“, herausnehmen sollten.⁵ Das so erzeugte Gedicht ist zwar ein Zufallsprodukt, nicht aber Tzaras Instruktion. Sie generiert dadaistische Sprache, ohne daß ihr eigener Sprachcode dadaistisch wäre. Noch konstruktivistischer wirkt das Gedicht „3 variationen zu ‘kein fehler im system‘“ des deutsch-bolivianischen Mitbegründers der konkreten Poesie, Eugen Gomringer, das den Satz „kein fehler im system“ permutiert:

kein fehler im system

⁵[Tza75]

kein efhler im system
 kein ehfler im system
 kein ehlfef im system
 kein ehlefr im system
 kein ehlerf im system
 kein ehleri fm system
 kein ehleri mf system
 kein ehleri ms fystem
 kein ehleri ms yfstem
 kein ehleri ms ysfem
 kein ehleri ms ystefm
 kein ehleri ms ystefm
 kein nehler ms systemf
 fkei nehler im system
 kfei nehler im system
 kefi nehler im system
 keif nehler im system
 kein fehler im system

Der Fehler in Gestalt des „f“ wandert in jeder Zeile um einen Schritt nach rechts, bis die ursprüngliche Ordnung wiederhergestellt ist. Da der Fehler dem Gedicht nach einem systematischen Prinzip eingeschrieben wird, gibt es zwar einen „fehler im system“, aber keinen Fehler im Gedicht.

Jodis Quellcode ist typographisch, linguistisch und semantisch viel reicher als Gomringers Verse. Von Tzaras Gedicht unterscheidet es sich zudem dadurch, daß es es seine poetische Collage nicht durch einen konträr gebauten Algorithmus erkaufte, sondern die Instruktionen selbst collagiert. Programmcode ist bei jodi nicht mehr Laborkonstrukt, sondern Pastiche und Ready-made. Im Gegensatz zur permutativen Proteuslyrik und zur generativen Kunst von Optatianus Porfyrius über die Lullisten bis in die zeitgenössische Kunst synthetisieren jodi ihre Instruktionscodes nicht mehr, sondern verwenden sie intertextuell, als vorgefundenes Zeichenmaterial.

NET ANTI NET

Andere Arbeiten auf www.jodi.org subvertieren die ästhetische Erwartung von opaken Quellcodes und verständlichen Outputs noch entschiedener als das oben zitierte Gedicht. „Location“ <http://www.wwwwwww.jodi.org> und „faq“ <http://www.wwwwwww.jodi.org/100cc/faq/index.html> erscheinen in der Standardanzeige des Web-Browsers als Buchstabenchaos, das erst in der Quellcode-Anzeige lesbar wird; dort entpuppen sich die blinkenden Codes als Typogramm der Zeichentrickfigur Stimpy (aus „Ren and

Stimpy“). So, wie die Website mit ihrer Zugänglichkeit und Unzugänglichkeit auf der Metadaten-Ebene ihrer Adressierung spielt, spielt sie auch mit sichtbaren und verborgenen Codes auf der Ebene des Dateiinhalts, macht das Undurchsichtige durchsichtig und umgekehrt.⁶

„Location“ und „faq“ beziehen sich intertextuell auf als „ASCII-Art“ bekannte Computer-Typogramme. ASCII, der „American Standard Code for Information Interchange“, der kleine gemeinsame Nenner aller Computerzeichensätze,⁷ wurde seit den 1970er Jahren benutzt, um in textbasierten Computernetzwerken Bildschirmgraphik zu emulieren, so daß die ASCII Art aus der Not technischer Beschränkung die klassische Figurendichtung neuerfand, wie aus der orphischen Lyrik des dritten vorchristlichen Jahrhunderts überliefert ist.⁸ Mit der Netzkunst von jodi und später den Teilnehmern der Mailingliste „7-11“ wurde ASCII Art von einer naiv-figurativen zu einer experimentellen und ironischen Kunstform umgeschrieben, die mit Fehlercodes, vorgetäuschten Systemabstürzen und Werbemüll-artigem Massenversand spielte. „Location“ und „faq“ kartographieren den Übergang von den figurativen zu den selbstreflexiven ASCII-Codes, in dem in ihnen die alte naive ASCII-Art buchstäblich zum Quellcode der neuen reflexiven ASCII-Netzkunst wird.

CODE ANTI CODE

Jodis besondere Fertigkeit, Daten eines Typs wie Daten eines anderen Typs aussehen zu lassen, manifestiert sich besonders in einer Kunstform, die eine echte jodi-Erfindung zu sein scheint: falsche Software, die in Wahrheit nur aus animierten Graphiken und blinkendem Browser-Text besteht. „Surgery/havoc“ <http://www.jodi.org/100cc/havoc/> sieht aus wie ein Jump-and-Run-Computerspiel im Stil der „Mario Brothers“ und bietet verschiedene Zoom-Funktionen, ist tatsächlich aber nur ein geschicktes Arrangement kleiner zappelnder Graphikdateien. Mit denselben Mitteln simuliert „BinHeX“ <http://www.jodi.org/100cc/hqx/i900.html> Systemabstürze und Computervirenbefall. In ständigem Bezug auf die alltagskulturelle Semiotik von Software-Interfaces gelingt es jodi, Softwarekunst – und somit auch generative Kunst – selbst dort zu schaffen, wo sie gar nicht algorithmisch programmieren; eine rhetorische Pointe, die das ganze definitorische Fundament beider Kunstformen hinterfragt.

⁶Diese Methode übernimmt und radikalisiert Ivan Khimins Projekt <http://imageless.net>. Die Website besteht aus extrem kleinen HTML-Dateien, die nur im Quellcode Information enthalten. (Zugang über <http://imageless.net/ASCII32/>)

⁷Die Geschichte des ASCII-Text rekonstruiert Dieter E. Zimmers Aufsatz [Zim97]

⁸[AE87]

Die prädigitale Netzkunst, die seit 1960er Jahren unter dem Namen „Mail Art“ bekannt wurde, förderte ihre offenen Strukturen durch konsequentes Mittelmaß. Jodi.org und die durch diese Website möglich gewordene Netzkunst zeigen, wie aus solchen Netzwerken auch herausragende Kunst entstehen kann. Was jodis Kunst zum ultimativen Beispiel zeitgenössischer „argutezza“ macht – also jener kunstvollen Gewitztheit und Urbanität, wie sie in Rhetoriken des 17. Jahrhunderts definiert wurde – ist die simultane Affirmation und Negation ihres Orts im Netzwerk. Denn sie selbst ragt viel zu sehr heraus im Geflecht ihrer selbstgeknüpften Beziehungen, deren Disseminationen ihnen zudem als Präventivwaffe dienen gegen unbotmäßige Lektüren und Kontextualisierungen wie diese hier.

LITERATUR

- [AE87] Jeremy Adler and Ulrich Ernst. *Text als Figur. Visuelle Poësie von der Antike bis zur Moderne*. VCH, Weinheim, 3 edition, 1990 (1987). 6
- [Arn01] Inke Arns. Texte, die (sich) bewegen: zur Performativität von Programmiercodes in der Netzkunst, 2001. <http://www.v2.nl/~arns/Lecture/performativ-code.html>. 3
- [Cay01] John Cayley. The Code is Not the Text, 2001. http://www.p0es1s.net/poetics/symposion2001/a_cayley.html. 1
- [Tza75] Tristan Tzara. Pour fair une poème dadaïste. In *Oeuvres complètes*. Gallimard, Paris, 1975. 4
- [Zim97] Dieter E. Zimmer. Schöne Gruse aus dem Netz. In *Deutsch und anders*, pages 272–292. Rowohlt, Reinbek, 1997. 6