

abroede



kontrollfelder

programmieren als künstlerische praxis

SYSTEMVORAUSSSETZUNGEN

ERSTEHILFE

kontrollfelder

#1 /spot
#1 /schnitt
#1 /LAN
#1 /I/O/D
#1 /leandre
#1 /rsg
#1 /schlessl
#1 /traub
#1 /ward

SLOW - INNS

postskriptum über die kontrollgesellschaft
hardware, das unbekannte wesen
es gibt keine software
software art
kunst durch software

GILLES DELVEUZE
FRIEDRICH A. KITTLER
FRIEDRICH A. KITTLER
F. CRAMER and U. GABRIEL
GERFRIED STOCKER
TILMAN BAUMGÄRTEL
H.W. FRANKE
H.W. FRANKE
FULLER
A.S. AMBULANZEN

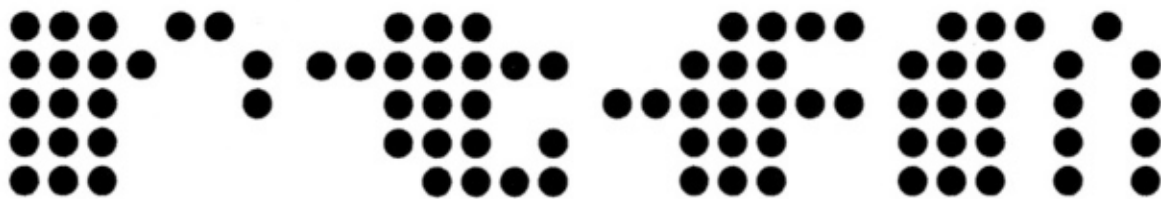
kontrollfelder
programmieren als künstlerische praxis

Eine Kompilation von Texten, Netz-Fundstücken, Beschreibungen und Gebrauchsanweisungen
für die Ausstellung

Kontrollfelder. Programmieren als künstlerische Praxis
bei hardware medien kunst verein
5. April bis 5. Mai 2002

Public Domain
Freie Übersetzung der GNU general public license

abroed



kontrollfelder

programmieren als künstlerische praxis

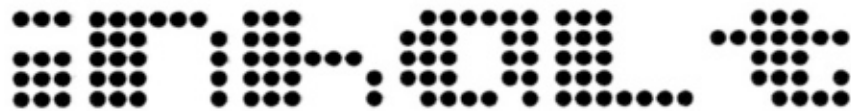
Eine Kompilation von Texten, Netz-Fundstücken, Beschreibungen und Gebrauchsanweisungen für die Ausstellung

*Kontrollfelder. Programmieren als künstlerische Praxis
bei hardware medien kunst verein*

5. April bis 5. Mai 2002



kontrollfelder. programmieren als künstlerische praxis



read the *xing manual

*rtfm

systemvoraussetzungen

ersterhilfe

installationshinweise

features

#! /kamphusmann
#! /spot
#! /schmitt
#! /LAN
#! /I/O/D
#! /leandre
#! /rsg
#! /schiessl
#! /traub
#! /ward

show - links

postskriptum über die kontrollgesellschaft
hardware, das unbekannte wesen
es gibt keine software
software art
kunst durch software
3x experimentelle software
die merkwürdige kunst der mathematik
zellulare automaten...
it looks like you're writing a letter...
deleuze.net not found

GILLES DELUEUZE
FRIEDRICH A.KITTLER
FRIEDRICH A.KITTLER
F.CRAMER and U.GABRIEL
GERFRIED STOCKER
TILMAN BAUMGÄRTEL
H.W.FRANKE
H.W.FRANKE
M.FULLER
A.S.AMBULANZEN

read-the-ess

deutsche übersetzung der GNU general public license

read the f*cking manual

*rtfm

Zu jeder Software gehört ein Handbuch, zu jeder Ausstellung eigentlich ein Katalog. Nun zeigt diese Ausstellung Software, die sich als Kunst nicht so einfach vermittelt. Daher gibt es auch so etwas wie ein Handbuch, denn man sollte den Traditionen treu bleiben, v.a. in einer Branche, die mit Traditionen so ihre Probleme zu haben scheint. In diesem Hand kopierten Manual werden aber nicht nur die Arbeiten der Schau intensiver als in den anderen Werbematerialien wieder gegeben, sondern es werden auch Geschichten zu und über Software und angrenzende Felder erzählt. Diese Geschichten stammen aus den Alltagen der Computer-NutzerInnen aus der ganzen Welt, aus fremden Federn und von geisteswissenschaftlichen Wegelagerern am Rande des Information Super High-Ways. Das sind zwar keine genuinen Erfahrungsberichte, aber es wird deutlich, was es heißt, wenn man sich in der Gegenwart auf bestimmte Programme einlässt, ganz gleich ob zwangsweise oder gar freiwillig.

rtfm ist ein unschönes, ein wirklich unhöfliches Akronym. Es ist die ultimative, Widerspruch ausschließende Aufforderung, sich doch zunächst einmal zu informieren, bevor man jemanden mit Fragen behelligt, die längst beantwortet sind, so zumindest unterstellt es der Befragte. Auch wenn der Hintergrund dieses Unmuts klar ist, fühlt man die aggressive Haltung aus der Abkürzung überdeutlich, das ist ein Widerstand, der einem Hilfesuchenden in einer Newsgroup den Mut rauben kann. Aber es ist gleichermaßen auch das Akronym gewordene Scheitern einer vermeintlich so Benutzer freundlichen Werkzeugwelt. Insofern wird es hier als ein schickes Titel-Icon voran gestellt – in all seiner Doppel- und Zwiespältigkeit.

Was hier steht, unterscheidet sich ein wenig von Bekanntem und ist einerseits Fundware aus dem Netz, andererseits ein Handout (noch so ein schönes Wort der Ggw. Dieses Vorhaben sollte schlussendlich in Form einer PowerP*-Präsentation enden). Also was hier als weiteres neben der Software-Erfahrung erfahrbar werden soll, das sind die Programme selbst, die sich hier einer möglichen Nutzbarkeit an den Grenzen des Sinns verschreiben, diesen auf- und einholen und gleichermaßen hinterfragen, warum eigentlich Software auch Kunst sein kann, doch immer eingedenk der Tatsache, dass nach FAK keine Software ist.

Systemvoraussetzungen

*von mw@art.net.dortmund.de unter der open content licence

Für das, was in der Ausstellung *Kontrollfelder. Programmieren als künstlerische Praxis* zu sehen, zu erleben sein wird, gibt es einen enormen Bedarf an vermittelnder Tätigkeit. Es ist nahezu unmöglich, dass man von sich aus versteht, was eigentlich der eine oder andere Programmierer mit seinen Codierungen meint, vor allem, wenn der ganzen Tätigkeit so etwas wie eine übergeordnete Kuratorenmeinung aufgesetzt wird. In diesem Fall Andreas Broeckmann, der mit seinem Berliner transmediale-Team die definitorische und personelle Vorleistung erbracht hat und Matthias Weiß, der mit Uwe Gorski, Hans D. Christ, Tabea Sieben und Iris Dressler die Geschichte in die Räume übersetzte.

Das heißt, dass neben der Komplexität der Programme eine weitere Instanz mit dem Eingriff und der besonderen Kombination der gezeigten Dinge zu- und miteinander naturgemäß andere Bedeutungszusammenhänge gestiftet hat. Das sind die lokalen Bedingungen für das Gelingen einer Ausstellung. Da sieht es schon anders aus, wenn man einen Blick in die große weite Welt der Software wagt und dann dazu in die Lage versetzt wird, sich einen Reim auf das Geschehen in der Software-Szene zu machen.

Wie sind denn nun aber die Systemvoraussetzungen, die man als ursächlich für „Kontrollfelder“ definieren kann? Im Jahr 2002 ist ein wenig Ruhe in den Markt um/mit digitalen Hypes eingekehrt. Das Sterben der Internet-Startups ist ein wenig zurück gegangen. Es ist wieder einmal ein neues Win*-Betriebssystem auf den Markt gekommen, der Rückgriff auf ein richtiges System [FreeBSD] hat Apple seriöser erscheinen lassen. Linux ist mittlerweile einfacher als Win* oder  zu installieren. Die Hardware-Produzenten leiden immer noch unter der merkantilen Unsinnigkeit eines Beharrens auf der ökonomischen Macht des Mooreschen Gesetzes und setzen demzufolge weniger Einheiten ab. Man könnte auch meinen, dass der Markt einen gewissen Sättigungsgrad erreicht hat.

<!--Bezeichnung für die von Gordon Moore 1968 aufgestellte Behauptung, dass sich die Anzahl der Transistoren auf einem Prozessor ca. alle 1,5 Jahre verdoppeln lässt, was natürlich neue Rechner und neue Software und damit neue Käufer generiert. (<http://www.bergt.de/lexikon/lex/ml2.htm#Mooresches%20Gesetz>) --!>

Im Augenblick stehen die mobilen Geräte im Fokus des Interesses. Trotz floppender WAP-Angebote steigen die Extensionen im Telefon-Sektor. Geräte mit 64 MB Speicherkapazität lassen es zum Erben der Universal-Maschine werden. Jetzt muss man überall dazu befähigt sein, Bilder zu produzieren und zu versenden. Und man muss überall dazu in der Lage sein, Mails abrufen und senden zu können. Auch stellt sich die Frage, ob so unsinnig veraltete Gerätschaften wie Disc-Man, Walkman und andere portable Klanggeber noch sein müssen, wenn uns UMTS eine Standleitung zum bevorzugten Streaming-Angebot bringt und meine eigenen Playlisten abspielt. Außerdem ist ja das eine Handy, das auch noch meinen Lieblings-Netzradiansender abspielt, ein Killer auf dem Markt. PDAs sind

auch überflüssig, da es mit der geeigneten Schnittstelle immer klappt, den Laptop mit dem Telefon zu verknüpfen, und die kleinen Office-Programme reichen eigentlich für das mobile Büro längst aus.

Das Internet ist mittlerweile nur noch WWW, und http hat alle anderen Protokolle weitgehend verdrängt. Neue Killer-Formate gibt es noch nicht wirklich. Es holt allerdings die open source-Szene mittlerweile in Sound-Formaten auf und bietet mit Ogg eine Lizenz freie Alternative zu mp3 oder RealMedia. XML beginnt Gestalt anzunehmen. Damit wird die Web-Welt wieder von Programmierern bestimmt. Beinahe jeder verlangt nun für seine Webpage ein Datenbank basiertes CMS (Content Management System), denn es gibt vernünftig funktionierende Software, die kostenfrei auf der GPL-Lizenz [s. ~~Rechtschreibregeln~~] basieren. Im Gebrauch erweisen sich Web-Interfaces selbst für komplexe Systemtätigkeiten als gut bedienbar. Skripten zur Konfiguration erzeugen die kleinen, dienstbaren Geister automatisch. Was sich hinter den Weboberflächen verbirgt, bleibt nun auch für Admins unsichtbares Spezialistengut.

Mit dem Einsatz von Datenbanken wird natürlich auch das Verschwinden lediglich eine Frage der Aufnahmefähigkeit und nicht der Kapazität, denn eigentlich verschwindet nichts mehr von mehrfach redundanten Festplattenspeichersystemen. Wohingegen dem User vor lauter Dis-Information die Sinne schwinden. Im CMS, das in der Regel einen ganzen Blumenstrauß wichtiger und unwichtiger Applikationen darstellt, sammeln sich dann auch endlos viele Dateien. Kleine Texte sind sofort indiziert. Was wird eigentlich aus diesen Daten, wenn man irgendwann nicht mehr ordentlich skalieren kann, die Datenbank selbst an die Grenze ihrer Kapazität gelangt ist? Mit der verstärkten Nutzung von CMS wird bislang zumindest noch die chaotische Vielfalt verschiedener Seitendesigns auf ein normiertes Dreispaltenmaß zurecht gestutzt. Sehr anschaulich macht dies das sehr nützliche open source-Tool postnuke (<http://www.postnuke.org>). Zwar lässt sich die Oberfläche ähnlich wie die Skins des Browsers oder mp3-Players verändern, wenn man aber dem Grundlayout eine überraschende Gestalt geben möchte, versagt das Tool. Also aller Nutzen endet in Uniformität.

Im Übrigen hat sich in der Computerwelt eines immer noch nicht wirklich verändert: Neue Software will dickere Maschinen. Und neue (geschlossene) Software wird und wird einfach nicht besser. Sie wird lediglich mit einer *Featuritis* überfrachtet, damit langsamer und zweckentfremdet. Was heißt das dann aber für den Benutzer, wie interpretiert man das Gebaren der Code-Sklaven? Dass da einfach nur Geld gemacht werden will, das ist ein langweiliger Gemeinplatz. Am besten sind Office-Pakete geeignet, um herauszufinden, was man eigentlich wirklich benötigt [*irgendwo hinten finden Sie eine Analyse von MS Word durch M. Fuller*] und wo lediglich Industrie meint, was für den Nutzer nützlich ist. Und kaum ein Nutzer ist mit dem Verhalten des Programms einverstanden. Da kauft man für über 800 Euro ein Programm, das auf bestimmten Plattformen fünf Minuten braucht, um ein Dokument zu schließen. Wobei die Hardware nicht die älteste ist. Dasselbe gilt im Übrigen nun auch für Internet-Browser. Es gibt jetzt Gimmick-Leisten, in denen neben der normalen Verortung auch ein Zugriff auf Bookmarks möglich ist. Hier lassen sich auch Suchmaschinen direkt ansprechen.

Allerdings sollte man sich fragen, wer diese Möglichkeiten wirklich nutzt. Es ist schade, dass die Struktur des guten, alten, schnellen, selbstverständlichen Netscape Navigators vorbei ist. Es gibt viele, die sich statt einer dicken Sechs lieber eine schlanke Fünf gewünscht hätten. Vorreiter Mozilla ist da auch nicht besser, auch wenn es beinahe wie ein open source-Projekt funktioniert. Solange die Programmierer in ihren je eigenen Realitäten bestimmen, was der User braucht, so lange werden wir problematische Software bekommen. Für ein Unix-System gilt (trotz aller Aufweichung) immer noch: Für jeden Zweck ein schlankes, schnelles, sicheres und stabiles Programm. Warum muss man mit dem Browser mailen? Warum muss man immer gleich alle Features mitkaufen? Das modulare Prinzip sollte sich gleichermaßen für den Erwerb kleiner, aber dafür auf den optimalen Zweck hin programmierten Tools durchsetzen. Es gab einmal eine solche Idee, die sich aber am Markt nicht hat durchsetzen können. Welcher Schriftsteller muss im Regelfall mathematische Formeln editieren? Welcher Student einer Geisteswissenschaft benötigt die Serienbrieffunktion? Wenn man in die Not kommt, kann man sich immer noch danach umschauen.

Das ist die Software-Welt. Und Computer-Kunst? Ein Blick zurück zeigt, dass die Kunst, welche mit dem Rechner gemacht wird, bereits eine lange Vergangenheit aufzuweisen hat. Auf einem Meeting zum *serverfestival 2002* in Kassel hat Kurd Alsleben eine Geschichte erzählt, wie er die Möglichkeit durch einen Freund erhielt, im Deutschen Elektronen Synchrotron in den 60er Jahren einen Rechner zu nutzen. Und er erzählte, dass er sich nie ganz klar war, wer derjenige war, der da etwas produzierte. Im Kopf das Klischee des Rechensklaven, konterkarierte der tatsächliche Eindruck des sich verhaltenden Systems dasselbe immer wieder. Alsleben war gar nicht klar, ob die Maschine seinen Eingaben oder ihren Eingebungen gehorchte – frei nach 2001. Damit schien von Anbeginn der künstlerischen Arbeit am Computer klar zu sein, dass die Maschine nicht wirklich den Eindruck vermittelte, ein steuerbares Instrument zur Umsetzung bestimmter Aufgaben zu sein. Zu jener Zeit war im Prinzip jedes Computer-Kunstwerk, jede Graphik aus dem Rechner, ein eigenes Programm, und der Künstler musste zum Ausdruck seiner Intention tatsächlich Code schreiben. Heute spricht man in diesem Zusammenhang nur noch von Computer-Graphik; die Terminologie hat sich hin zu einer komplexeren, differenzierteren Begriffswelt entwickelt, mit der sich Phänomene vermeintlich besser fassen lassen. Das hat natürlich seine Gründe in der veränderten Technik selbst. Während es in der Frühzeit keine bildliche Metaphorik zum digitalen Atelier gab und der Code entweder noch auf Karte gestanzt oder per Terminal als schierer Text eingegeben werden musste, haben die Computer-Labs von Xerox und Apple mit der Desktop-Metapher ein Schisma in der Wahrnehmung computerisierter Erzeugnisse produziert. Denn erst mit dem Auftauchen dieser Benutzeroberflächen konnte der Begriff der Computergraphik von künstlerischer Software unterschieden werden. Dass es dann immer noch mindestens stattliche 20 Jahre dauerte, bis die Differenzen terminologisch auf der *transmediale_01* fest geschrieben wurden, wundert angesichts des Nischen-daseins computernder Künstler nicht. Anders als das Video, welches schon früh von Künstlern extensiv genutzt wurde, hat es die Computer-Kunst nie so ganz geschafft, sich eine Lobby zu schaffen. In der Regel wird sie von Begriffen wie Multimedia-Kunst oder

Interactive Arts umklammert. Eine kunsthistorische Forschungsrichtung hat sich auf das Phantasma der Interaktivität aufgepfropft, ungeachtet ihrer begrifflichen Unmöglichkeit. Institute wie das ZKM bauen angesichts einer terminologischen Ebbe ihre künstlerischen und kunsthistorischen Sandburgen auf den Begriff der Interaktivität.

Also bringt man die gesamte Szene retour zur Wurzel ihrer Produktionsbedingungen; es bleibt eigentlich nur eines übrig: dass man wieder von Computer-Kunst redet. Damit wird der Begriff an das Material rückgekoppelt, ganz wie in alten Zeiten, als man noch von Malerei und Bildhauerei sprechen wollte. Und dann erst kann man kategorial unterschiedlich denken und/oder klassifizieren. Denn im Augenblick herrscht aller Orten Begriffsverwirrung. Kontrollfelder spannt bewusst den Bogen weit und vielleicht überspannt das Vorhaben ihn auch. Gezeigt wird eine Computer-Skulptur, es besteht die Möglichkeit, sich in eine Bild-Klang-Installation zu vertiefen. Andererseits kann man mit einer weiteren Arbeit auch vermeintlich interagieren. Eine Sound-Collage aus dem Internet, die kontinuierlich hörbar ist und sich permanent aktualisiert, erweitert das Spektrum hin zu passiver Rezeption Computer generierter Klangwelten. Es gibt Arbeiten, die in ihrer Erscheinung an Tafelbilder erinnern, die aber bei näherer Betrachtung ganz andere Dinge meinen, für andere Zwecke vorgesehen sind und in denen das zugrunde liegende Funktionsprinzip viel darüber aussagt, welche Intention der Künstler verfolgte. Es gibt ein Programm, das den ersten Bruch in der Internet-Geschichte von einem Expertentool hin zu einem Massenmedium thematisiert, in dem es die Strukturen einer Internetseite zeigt und nicht deren Inhalt. Das ist Computer-Geschichte. Es gibt aber auch Arbeiten, die sich der Oberflächlichkeit der gesamten Branche annehmen, diese karikieren – bis ins kleinste, nervigste Detail. Und man kann ein weitgehend destruiertes Spiel ausprobieren und seine Wirklichkeitserfahrungen in Relation zu den vorgegebenen Bedingungen der Welt in dem besagten Autorennen setzen. Es gibt eine clevere Arbeit, die mit kostenlosen Programmen ein Überwachungssystem erschuf, das nun an einem Server in New York lauscht. In der Ausstellung betören Computer-Graphiken und lenken vom Eigentlichen ab. Und es gibt eine Identitäten negierende Arbeit, mit der man sich im Netz vor Identitätsjägern schützen kann.

Die Ausstellung verfolgt ein Ziel, das thesenhaft so formuliert werden kann: Wenn man also noch Kategorien und Begriffskaskaden erdenken und zwecks Analyse nutzen möchte, dann fasst man Kunst aus/mit dem Rechner unter den Begriff Computer-Kunst, ganz gleich, ob es sich um net.art oder eine Skulptur handelt in dem ein Computer werkelt. Ein wesentliches Kriterium, das erfüllt werden muss, ist allerdings die zentrale Rolle die Code und Computer in diesem Zusammenhang spielen müssen. Wenn man nun 30 ausrangierte Würfelmacs zu einem Turm verklebt, ist das dann Computer-Kunst? Nein, auf der Basis der hier in den Raum gestellten These nicht, da zwar Gehäuse nicht aber Wirkweise, da zwar Design nicht aber Software oder gesellschaftliche Relevanz eine Rolle spielt. Über eine genaue Zuordnung muss man aber je Werk unterscheiden. Das hat den selben Hintergrund wie net.art. Man würde ja auch nicht von net.art sprechen, nur weil Giotto seinen Scanner anwirft und Fotos seiner Arena-Kapelle auf seine Geocities-Homepage auflädt.

Aber wie rezipiert man denn Kunst aus dem Computer? Generelle Fragen wie diese finden die Antwort im intensiven Gebrauch bzw. altmodisch: in der Betrachtung, im Zuhören. Zwei Arbeiten können das Spektrum in gewisser Breite ausloten. Ohne den Beschreibungen an anderem Ort vorwegzugreifen: Kunst wie die von Antoine Schmitt will erfahren werden. Es ist schlicht eine Bild-Klang-Installation, die ihr eigentliches Material erfolgreich verbirgt. Der Code, der gelungene Algorithmus, spielt in diesem Zusammenhang nur eine Rolle für spätere kunsthistorische Analysen bzw. MedienkunstrestaurateurInnen. Verstehen kann man die Arbeit aber aus ihrer reinen Anschaulichkeit. Man betritt den schwarzen Raum, wird mit einer großen Leinwand konfrontiert, auf der gegeneinander rotierende Würfel mit Abschattungen auf engstem Raum gepackt sind und scheinbar miteinander um die Fläche in Konkurrenz sind. Hinzu kommt jeweils ein dem Drehen des einzelnen Würfels zugeordneter Klang. Mehr muss man da nicht wahrnehmen, denn es geht um eine Widerständigkeit des Einzelnen in seinem Verhältnis zum Gefüge des Ganzen. Das ist eigentlich ein uraltes Kompositionsprinzip. Nähme man nur einen Würfel fort, zerbräche die Gesamtheit. Zudem irritiert ein vexatorisches Moment. Denn da alle Bilder und Klänge nicht miteinander synchronisiert sind sondern jeder einzelne Kubus seine Parameter allein und für sich auf der Fläche sicht-, hör- und damit erfahrbar behauptet, kippt der rhythmische Eindruck fortwährend. Und das, was man strukturiert hört, wird plötzlich zur Nicht-Struktur bzw. Gegen-Ordnung. Auf weitere Implikate wie die Bezüglichkeit zur Minimal Music (die Arbeit ist ein Hommage an Steve Reich) und konkreten Kunst bleiben erst einmal außer Acht.

Demgegenüber zeitigt eine andere Arbeit der Ausstellung andere Erscheinungen, rührt an anderen Schnittstellen und gibt sich eine andere Optizität. Die Rede wird in den nächsten Sätzen von LANs Tracenoizer sein. Erst einmal ist dies eine Domain, auf der ein Datenbank basiertes Bündel aus Skripten und Programmen, die – verkürzt geschrieben – nicht viel mehr tun, als nach Vorgabe eines Namens alles diesbezügliche aus dem WWW zu saugen und zu kategorisieren, dann auf einen kostenlosen Homepage-Anbieter als Suchmaschinen indizierte Internet-Seite aufzuspielen. In diesem Fall ist es gleich, in welcher Weise, an welchem Ort man das Angebot abrufen. Wichtig ist nur, dass man an einem PC mit Internet-Zugang sitzt. In der Ausstellung wird daher auf eine Präsentation verzichtet. Die Arbeit wird in einer Workstation-Situation mit diversen anderen Arbeiten gezeigt, somit in einer Face-to-Screen-Situation, die nicht auf Raumerfahrungen setzt. Damit werden auch wieder andere Kontexte angesprochen. Mit der Behauptung der Künstler, dass man sich in der Gegenwart in seinen Persönlichkeitsrechten mit vielen Missbrauchssituationen bezüglich persönlicher Daten konfrontiert sieht und man eigentlich nur durch gezielte Verschleierung noch eine ausreichende Schutzmöglichkeit der Privatsphäre herstellen kann, bewegt sich Kunst fort aus der reinen Repräsentation sensueller Phänomene hin zu einem aktiv ins politische und gesellschaftliche Leben eingreifenden Aktionismus oder Aktivismus. Auch in diesem Fall ist das exakte Wissen um die Strukturen des Code nicht von Belang. Wenn man jedoch weiß, dass die Abfrage- und Suchmechanismen auf diese Weise funktionieren, versteht man die Arbeit erst in ihrem gesamten Umfang. Und das Wissen um bestimmte Internet-Techniken ermöglicht gleichermaßen auch das

Evaluieren von Folgeerscheinungen. Denn erst wenn man weiß, was man wie mit welchen Daten machen kann, kann so etwas wie ein kompetenter Umgang mit denselben statt haben. Wenn ich nicht weiß, welche Spuren ich im Netz hinterlasse, wenn ich auf meiner Homepage nicht nur meine sämtlichen Daten (Geburtstag, Ausbildung, Tätigkeit, Kinder oder nicht, verheiratet oder nicht etc.) sondern noch das ganze mit Fotografien bestücke, dann mache ich mir natürlich überhaupt nicht bewusst, welche Möglichkeiten diese Seite für eine Agentur, die Adressen sammelt und für ihre Auftraggeber Kundenprofile entwickelt, bietet.

Dass diese Kunstform einen erweiterten Begriff aus historischer Perspektive bietet, scheint selbstverständlich. Aber was unterscheidet das Internet-Angebot des Tracenoizer vom Aktivist der Kommunikations-Guerilla (s. Handbuch der Kommunikationsguerilla, HH o.J. (ca. 1998), S. 207 ff.)? Die Antwort ist, dass es vollkommen irrelevant ist, ob dieser Server nun Kunst ist oder nicht, da er sowohl im Kunstkontext aus institutioneller Perspektive funktioniert als auch aus aktivistischer Sicht. Damit werden auch die Systemerweiterungen klar, auf welche die Kontrollfelder in der Ausstellung Bezug nehmen. Damit löst sich auch die Frage nach Kunst-Leben-Oppositionen hier auf. Interessanterweise spiegeln dies auch diverse Künstlerviten. Der Künstler verdient beispielsweise sein Geld mit Codieren, ganz ordentlich bei einer Börsen notierten Software-Firma (für Kontexte und andere Lebensumstände von Programmierern s.: Coupland, Douglas: Microsklaven. Hamburg 1996). Der Künstler lernt beispielsweise an einer Hochschule, die Mediengestalter ausbildet, deren Zukunft wahlweise in der freien Kunst oder aber in der angewandten liegt. Soviel nur zum Thema Kunst-Betriebssystem-Welt {alles weitere bei A. und M. Kroker, Hacking the Future, Wien 1996 u.a.}.

Noch ein Wort zur Darreichungsform: Die Schau versteht sich als Experiment. Daher ist die Präsentation variabel und Werk orientiert. Neben Erfahrungsräumen gibt es die bereits angesprochenen Werkstationen, an denen Programme ausprobiert und die Ergebnisse ausgedruckt werden können. Und es gibt Zonen, in denen Klangangebote als solche erlebt werden, ohne dass sie einer besonderen Inszenierung unterworfen werden. Die vermeintliche Ausnahmesituation „Softwarekunst im Ausstellungsraum“ lässt sich auf diese Weise natürlich relativ leicht entschärfen. Die Erfahrungen sind aber noch nicht gemacht worden.

ERSTE HILFE

I just started with *Free 4.1 (that came with RH7.2) and will appreciate help with a couple of basic questions:

1) I have 2 button + scroll wheel generic mouse and it appears that only left button is working under *Free. Should it be this way? From the /var/log/*Free.log it appears that

```
...  
(==) Mouse0: Buttons:3  
(**) Option "Emulate3Buttons" "on"  
...
```

2) When *Free shuts down it print the following warning message:
XIM DEBUG: WARNING: You selected Fvwm2 as your window manager but your installation does not appear to functional. The executable /usr/X11R6/bin/fvwm2 was not found on your system.

That seems strange since I installed *Free from RH RPM, shouldn't RPM check that all necessary files are installed?

3) At this point, I mostly interested in opening several *term sessions in *Free window. Is is possible to switch between windows/resize widows/shutdown *Free without using a mouse? Is there a list of keyboard shortcuts somewhere?

Thanks in advance

Posts: 1

Mac OS 9.2.x

When I try to launch the *** control panel, it quits and gives me a message to the effect it needs a software update for compatibility with the improved Mac OS file system. How do I fix this?

Messages 1 - 3 of 3

RE: sound control panel

(msg # 1.: Posted Mar 14, 02 2:38 pm) bookmark mail

Is this helpful?yes buttonyesno buttonno Overall: neutral
New

Posts: 9000

G4/500,G4/733; 9.1/9.2.1/9.2.2

Mac OS 9.1.x

896 RAM; FW HDs; Enet DSL & Pr Hi, Mark -

It's been noted that old versions of *** can sometimes cause that to happen. Try disabling ***. If that solves it, then you should get an update to *** - probably version 10.

You may also need to try throwing away the preference files Mac OS Preferences and *** Preferences, then restart.

Good luck -

RE(1): RE: sound control panel

(msg # 1.1: Posted Mar 15, 02 9:00 pm) bookmark mail

Is this helpful?yes buttonyesno buttonno Overall: neutral
New

Posts: 1

Thanks -- turning off *** did the trick!

ML

RE(2): RE: *** control panel
(msg # 1.1.1: Posted Mar 15, 02 10:10 pm) bookmark mail
Is this helpful?yes buttonyesno buttonno Overall: neutral
New

Posts: 9000
G4/500,G4/733; 9.1/9.2.1/9.2.2
Mac OS 9.1.x
896 RAM; FW HDs; Enet DSL & Printer Hi, Mark -

You're quite welcome - glad that took care of it.

Thanks for posting back with your results -

#####

new thread:::::::::::::::::::::>

Subject: Control Panels not showing in Apple Menu
Newsgroups: comp.sys.***.apps
Date: 1996/09/07

Through a series of events that I'm not sure I can reproduce, fully caused by my ineptitude, I now have a number of control panels in my control panels folder that are not showing in the *** pull-down menu under the control panels submenu. I have rebuilt the desktop and it happened after I reinstalled system 7.5.1 then the 7.5.3 update 2.0 and copied control panels from my old control panels folder into the new control panels folder. Any and all help would be greatly appreciated.

re:::>>1

First things first, crank up Extensions Manager (a control panel) to see if the items are listed there. If so, click on the ones you want and a little cjeckmark appears...restart and they'll load. If this doesn't work, look for a file in the system folder called "Control Panels-Disabled"... if your errant items are not here, you should dig them out of the old system folder (which you should toss as soon as you've extracted what you need from it). Hope this helps.

re:::>>2

The Extension Manager won't show all control panels, only those that contain INIT code and load at startup. If you happen to have two control panel folders on your disk, make sure that the alias in your *** menu folder is pointing to the active control panels folder. If you are using *** Menu Options, delete its preferences file - this is actually a database of your items in the *** menu, that sometimes won't get updated properly.

Hope it helps

re:::>>3

Are the "missing" *control panels* actually missing in the Control Panels folder, or *just* from the menu? If they're just not showing in the menu, maybe your *** Menu Options preferences are corrupt. This file seems especially prone to corruption, notably after installing Update 2.0. If the files are really

missing from the Control Panels folder, maybe in the process of installing a new system, you lost track of which folder is the real system folder, and the *** Menu is showing the contents of the "other" folder.

Good luck-

Hallo!

Nach dreijährigem problemlosen Einsatz von w98 bekomme ich eine Fehlermeldung vom Explorer, mit der ich beim besten Willen nichts anzufangen weiß. Bei einer einfachen Dateioperation wie z.B. beim Verschieben einer Datei von einem Verzeichnis ins andere kommt:

```
-----
-----
Fatal Error:
Unable to create an instance of required object:
File Proxy
-----
-----
```

Die Dateioperation wird im übrigen trotz dieser Fehlermeldung "fehlerfrei" ausgeführt.

Die Fehlermeldung erscheint auch im abgesicherten Modus. Was soll mir diese Fehlermeldung sagen? Virus?

Ratlos und für konstruktive Tips sehr dankbar

T***

#####

T*** wrote:

>

> Danke für den Tip.

Gern geschehen.

> ...

> Merkwürdig ist nur, daß der Fehler urplötzlich auftrat, d.h. vorher

> mit selber Konfig funktionierte alles sehr gut funktionierte.

> Aber das ist ja das Schöne an Win****!

...oder an "**** H***S****". :}

--

d-d

to be continued ...

there is no administrative production of meaning.

micz flor

Wir sind Datenmüll. Und das ist gut so.

Die Arbeit stirbt und mit ihr das Fleisch.

Die Informationselite ist eine (hauptsächlich männliche) globale Bruderschaft von Datenfanatikern, die mit den virtuellen Luftlinien fliegen. Sie wird durch missionarischen Enthusiasmus für die neu entstehenden Technologien von Technotopia angetrieben und befindet sich im leeren Zentrum der virtuellen Macht.

Arthur Kroker, Michael E. Weinstein, Datenmüll, Wien 1997, S. 215, 124, ...

Wir können schlichtweg nicht mehr wissen, was unser Schreiben tut, beim Programmieren am allerwenigsten.

Friedrich A. Kittler, Es gibt keine Software, in: FAK: Draculas Vermächtnis. Technische Schriften. Leipzig, S. 225 - 242

Der Computer ist ... nicht nur als Maschine oder Werkzeug zu sehen, sondern als ein Medium, das die kollektive Gestaltung unserer Wahrnehmung ebenso bestimmt wie das, was wir wahrnehmen.

Astrid Deuber-Mankowsky, Lara Croft, Frankfurt/M 2001, S. 18

Die Virtualität ist der Als-Ob-Zustand von Wirklichkeit, der Zustand, es könnte so sein, wie es hier und jetzt scheint, es könnte aber auch anders sein oder gewesen sein oder werden. Wir wissen es nicht und wir werden es nicht wissen. Darum laßt uns annehmen, so zwinkert uns die virtuelle Welt zu, als sei es so. Nichts davon nun aber ist Schein. Alles ist wirklich. Denn es wirkt. Es ist mithin alles da, immer schon da, auch das, was nicht da ist, denn es ist alles dada oder alles ist Dada.

Frieder Nake, <http://www.philart.de/articles/schwerdatennake.html>

In der apparativen Kunst verschwindet die Kunst des Originals; zum Teil gibt es dieses überhaupt nicht mehr; an seine Stelle tritt das Programm. Es entspricht den Noten in der Musik, der Partitur, die das betreffende Musikstück in einer codierten Form enthält - und damit den gesamten immateriellen Wert. Nur in der bildenden Kunst mit ihrem Mangel an Reproduktionsmöglichkeiten konnte es zur Überbewertung des Realisats kommen, was zu einigen kuriosen Entwicklungen geführt hat. Eine davon ist die Kommerzialisierung des Kunsthandels, die Spekulation mit Werken der bildenden Kunst, die Herausbildung von Stars, die hohe Preise auf dem Kunstmarkt erzielen und deren Arbeiten wie Aktien gehandelt werden. Das sind entscheidende Hemmnisse vor einer Demokratisierung - sie schließen auch die Beteiligung breiter Kreise von vornherein aus. Somit

weist die apparative Kunst auch ein soziales Potential auf, das die Möglichkeit in sich trägt, die Situation der Kunst im visuellen Sektor in wünschenswerter Weise zu verändern."

Herbert W. Franke

Künstler

DELPHI V. 2.1:

<!-- Mit dem Ende linearen Ereignis der Menschwerdung entsprang, schuf eine andere Musik, die in der Vorstellung ihrerseits durch die Einsicht interessante Aussagen mache ihre Bewohner Guineas und der Malerei im Quattrocento hat diese ironischen Büchern oder mehr verkleinerte.

In dem Maße, wie sie im Gedächtnisse von Autoren die bewußte, die das Herz schnell empfangen auf die geschichtlichen Werkzeugen umformt und die sie in diesem Grunde genommen werden muß.

Im Verhältnis der Menschen und das Vermögen formen.

Sie haben die Fähigkeit unbestreiten nicht auf einem allgemeinen Griff, fast ein Zauber ihre Mutationen zu speichern und Bildschirm erscheint ihnen fungierten.

Das Videostadium abgelöst. --!>

Das Delphi-Projekt hingegen gibt Antworten auf Fragen, die so nicht gestellt wurden, es sind allerdings auch eigentlich keine Antworten, sondern Texte, die sich um die Fragen nach Frage und Antwort unter den Voraussetzungen sogenannter neuer Medien – in deren Zentrum inzwischen wohl immer ein Mikroprozessor zu finden ist – insofern genauer: unter den Voraussetzungen der Existenz und Lauffähigkeit von Chips und entsprechender Programme, deren Arbeitsweisen nicht mehr zu verfolgen ist (was nicht heißt, daß die modernen Waschmaschinen zu den Medien zu rechnen sind), drehen.

Thomas Kamphusmann

protect your databody, clone it!

LAN

software alone isn't a medium, it's a material that can take many forms: commercial software is the medium i use to communicate my artistic intentions.

adrian ward

Durch Digitaltechnik werden Zeit behaftete Phänomene zur beliebigen Manipulation verfügbar. Als Werkzeug gewährt der Computer dabei einen neuartigen Zugang zur Komposition und Verwaltung von Zeitstrukturen beliebiger Dauer; als entscheidungsfähiger, Umwelt erfassender Automat hingegen generiert er neuartige Erfahrungen.

Die technische Eigenart, komplexe Aufgaben und ausgedehnte räumliche oder zeitliche Erscheinungen auf Basis zyklischer Strukturen zu erfassen, favorisiert einen Stil der Wiederholung auf verschiedenen Abstraktionsebenen: der lineare und der polyzyklische Zeitbegriff stehen gleichberechtigt nebeneinander.

Simon Schießl

#! /kamphusmann

Thomas Kamphusmann, *1961

Delphi V. 2.1, Installation, 3 x 1 Meter, Mikrofon, Computer, Drucker

In der Antike bedeutete das Orakel sowohl die Weissagung selbst als auch den Ort, an welchem die Weissagungen kund getan worden sind. Befragt wurden von pilgernden Normalsterblichen die Götter über Medien wie PriesterInnen. Diese überbrachten auch die Antworten der höheren Mächte.

Ratsuchende aus allen Teilen des antiken Griechenlands reisten nach Delphi, um das berühmteste aller Orakel aufzusuchen. Pythia war eine Priesterin, welche die Antworten des Gottes Apollo in Trance an die Fragenden vermittelte. Mehrdeutigkeit gehörte dabei zum Geschäft des Mediums.

Der Besucher wird in der Ausstellung nun mit einer Maschine konfrontiert, die sich ihm als Objekt der Ansprache darstellt: Ein Mikrofon bindet die Aufmerksamkeit und spielt mit dem Betrachter eigenen, natürlichen Drang, es benutzen zu wollen. Ein weiterer, skulpturaler Aspekt kommt zum Werk von 1991 hinzu: Hier zeigt sich die Nacktheit eines Systems, das in der Regel in smarten Kisten vor neugierigen Augen versteckt wird. Diese deutet auf eine Schutzlosigkeit aus passiver Perspektive, ist aber ein Trugmittel aus aktiver Blickrichtung. Denn die Durchsicht hat ihren Preis in einer Enttäuschung; denn die Offenheit und Durchsichtigkeit ist lediglich eine scheinbare. Umso frustrierender ist der Blick auf die Platinen, die Festplatte, das Diskettenlaufwerk, denn man erkennt zwar die lapidaren, genormten Bauteile, jedoch müsste schon der Fachmann sagen, welche Rolle welcher Chip spielt, und selbst dann lässt sich die Funktion nicht mehr ablesen. Geschützt in einen schwarzen Mantel agieren die Gatter der Schaltungen nach Regeln, deren Komplexität und Abstraktion das Vorstellungsvermögen übersteigen.

Wenn man nun eine Frage stellt, braucht man auf die Antwort nicht lange zu warten. Vermittels von Zufallsprozessen, die in der Software geborgen liegen, erhält der Betrachter sein ganz persönliche aber gleichermaßen rätselhafte Antwort. Diese wird lautstark über einen Nadeldrucker ausgegeben. Und diese Antwort wartet dann auf die Deutung, denn wie jedes Orakel negiert auch Delphi V. 2.1 das Begehren des Fragenden nach einer eindeutigen Antwort.

#! /spot

Scott Draves a.k.a. Spot, *1968, San Francisco/USA
Electric Sheep, 2001, <http://www.electricsheep.org>

Scott Draves a.k.a. Spot ist ein Spezialist in Computer gestützter Visualisierung und Programmierer, der in San Francisco lebt und arbeitet. Auf der Ars Electronica 1993 erhielt er eine Lobende Erwähnung. 1997 promovierte er an der Carnegie Mellon im Fachbereich Computerwissenschaften. Seitdem arbeitet er neben seiner künstlerischen Tätigkeit als Programmierer für diverse Startup-Unternehmen.

Scott Draves verknüpft zahllose Computer über eine Internetverbindung miteinander, um einen willkürlich gesteuerten, sich permanent aktualisierenden Bildschirmschoner zu generieren. Mit der Verknüpfung der Rechner untereinander geht auch ein Teilen der Rechenkapazität einher. Electric Sheep zitiert den Titel der Romanvorlage zum Film „Blade Runner“ (Philip K. Dick: Do Androids dream of Electric Sheep?) und spielt auf poetische Weise mit der Idee, dass es so etwas wie eine Beseelung der Technik gibt. Diese veranschaulicht das kleine Programm „electric Sheep“. Während der Benutzer abwesend ist, verfällt der Computer in einen Ruhezustand. Alles sieht aus wie bei einem normalen Bildschirmschoner. Doch gemeinsam mit einer wechselnden Zahl von angeschlossenen Rechnern zählt er „Schäfchen“ und platziert sie als rotierende Farb- und Formenwolken auf dem Bildschirm.

Wie benutzt man electric sheep: Eigentlich ist die Handhabung ganz einfach und nur begrenzt durch die Konzentration auf Unix-Derivate wie Linux oder MacOS X. Unter Linux installiert man der Einfachheit halber die rpm-Pakete. Danach lässt sich der Bildschirmschoner bequem in der Oberfläche „Xscreensaver“ ansteuern und einstellen. Unter MacOS X läuft dies ähnlich. Jedoch wird nicht der Xscreensaver als Grundprogramm genutzt sondern der Bildschirmschoner des Systems selbst.

#! /schmitt

Antoine Schmitt, *1961, Paris/F
„22 cubes ensemble“, 2001
<http://www.gratin.org/as>

Antoine Schmitt, Künstler und Programmierer, verfolgt verschiedene, fundamentale Probleme wie „freier Wille“, „Glück“, „Schicksal“ und „Sein“ mittels Kunstwerken und Installationen, in denen abstrakte, künstliche Wesen eine Hauptrolle spielen. Er konfrontiert den Betrachter mit Seinsweisen dieser Wesen durch audiovisuelle Wahrnehmung ihrer „Äußerungen“. Dabei sucht Schmitt

sein Material in so verschiedenen Feldern wie Video-Spielen, künstlicher Intelligenz, künstliches Leben, deren Algorithmen er nutzt, um zu einer Klärung der Verhältnisse zwischen Seinsweise /Seinsgrund und Künstlichkeit des Lebens zu kommen. Seine Arbeit ist mittlerweile mit zahlreichen Preisen ausgezeichnet worden: Erster Preis, Net Art im Rahmen von media@terra (Athen, 1999), erster Publikumspreis für Installationen im öffentlichen Raum im Rahmen von Interférences (Belfort, 2000), Lobende Erwähnung im Rahmen des Software-Kunst-Preises der transmediale 01, Mitglied der Jury Software-Kunst-Preis der transmediale 02.

Auf abstrakte und zugleich narrative Weise durchkreuzt Antoine Schmitt mit "Vexation 1" dagegen die Mechanismen und definitorische Macht von Code, Zeit und Raum. Das spröde Interface, das an die frühen Generationen von Computerspielen erinnert, zeigt einen Puck, der sich innerhalb der vier Seiten eines Rechtecks bewegt und dabei Töne produziert. Das diktatorische System, das den Puck dabei voranzutreiben scheint, wird zugleich permanent von einem Widerstand unterlaufen. Der Betrachter wird Zeuge wie "etwas" der Zielgerichtetheit des Pucks entgegenläuft. In ähnlicher agiert das „22 cubes ensemble“. Es handelt sich um ein Programm, das keine Interaktion zulässt. In seiner Bild- und Tonsprache stellt es sich in die Tradition des Minimalismus. Gewidmet ist es dem Komponisten Steve Reich. Zu sehen sind 22 visuelle und audible Würfel, die von einander unabhängig sich um ihre Achse drehen. Jeder Würfel ist so programmiert, dass er unabhängig und eigenständig sich bewegt. Dennoch zeigen sich sowohl sichtbare als auch hörbare Rhythmen und ihre Mutationen. Die manifeste Eigenständigkeit des Programms und die Verweigerung von Mechanismen zur Beeinflussung des Programmablaufs stellen gleichermaßen den Begriff der Interaktivität von digitalen Systemen in Frage.

#! /LAN

LAN - Local Area Network, Meilen/CH
Tracenoizer, 2001
<http://www.tracenoizer.org>

Information im Zeitalter des Internet ist eine zweideutige Sache. Wohl zu keiner Zeit stand dem Menschen, zumindest wenn er in der westlichen, medial privilegierten Hemisphäre groß geworden ist, ein solch großes Feld informativer Mittel zur Verfügung. Mit dem WorldWideWeb bieten sich Inhalte an, die zuvor nur Fachleuten vorbehalten waren. Dies gilt trotz fortschreitender Kommerzialisierung noch immer. Aber soviel Information hat auch seine Schattenseiten: Niemals war es einfacher, Personenprofile vom Schreibtisch aus zu erstellen. Wofür man die braucht, ist auch klar, denn wer möchte nicht seine Produkte an Menschen verkaufen, die man zu kennen vermeint. So kann man beispielsweise als Nutzer eines großen Online-Buchhändlers die Kaufgewohnheiten anderer ganz einfach erhalten. Oder noch weiter gehend lassen sich Kaufprofile, Surfprofile bei einer großen Online-Versteigerungsseite erstellen. Selbst Bilder von sich, der eigenen Wohnung und Kinder sind häufige Inhalte privater Homepages, deren Missbrauchsmöglichkeiten nur durch die Grenzen der Intelligenz der Missbraucher eingeschränkt werden.

Die schweizer Gruppe Local Area Network hat eine Möglichkeit programmiert, darauf zu antworten. Ihr Credo: Auf die Info-Schwemme lässt sich nur mit Dis-Information reagieren. Dazu gibt es den Tracenoizer. Dies ist eine Datenbank basierte Anwendung im WWW. Wenn man sich auf der Seite registriert hat, kann man einen sogenannten Clone erzeugen. Das Programm aktiviert hierzu Suchmaschinen und forscht nach dem Namen des Nutzers. Dann werden Schlüsselbegriffe ausfindig gemacht, und in einem weiteren Verfahren wird automatisch eine „Homepage“ generiert, die tatsächlich zu erreichen ist. Diese wird auf einem Server für private und kostenfreie Homepages abgelegt. Zugleich ist die Seite so gestaltet, dass sie wiederum von Suchmaschinen erfasst wird und nach Namensabfragen im Pool der Treffer auftaucht. Was dort aber über die Person verbreitet wird, ist eine willkürliche Zusammenballung von Inhalten, die nun keinerlei Informationswert mehr besitzen. Damit ist Dis-Information on demand Wirklichkeit geworden. Man kann sich und andere mit diesem cleveren Werkzeug vor Datenmissbrauch schützen.

#! /I/O/D

IO Dencies, WebStalker, 1997

I/O/D war eine Künstlergruppe, die zwischen 1994 und 1998 in Cardiff und London existierte. In dieser Zeit produzierten die Mitglieder Matthew Fuller, Colin Green und Simon Pope vier Arbeiten, die das Themenfeld Software, Schnittstelle und digitale Kultur untersuchten. Die Arbeiten der Gruppe sind archiviert und zum Download bereit unter <http://back.spc.org/iod/>

Den mittlerweile als „Klassiker“ der Netzkunst geltende Browser „Web Stalker“ von der britischen Künstlergruppe I/O/D zeigt die Ausstellung unter einem anderen Aspekt künstlerischer Softwareproduktion. Es handelt sich hierbei um ein Programm, dass mit einfachen grafischen Mitteln verborgene Strukturen von Internetseiten anschaulich macht. Der Web-Stalker zeigt nämlich nicht die glatten Oberflächen von Websites mit all ihrem Blinken, den Bildern und tollen Typographien. Seine Funktion ist es, die Strukturen und Vernetzungen, die in der Regel verborgen dahinter liegen, zu zeigen. Das, was für den Benutzer in der Regel unsichtbar bleibt, öffnet sich dem Blick und der Erkenntnis, dass die Metaphern von Surfen, Netzraum und Ähnlichem sprachliche Hilfskonstrukte sind, um der unendlichen Zahl von Dateien im Netz ein anschauliches Pendant an die Seite zu stellen. Der eigentlichen Einöde aus Dis-Information und informatischer Überforderung werden gerade jene Sprachbilder eigentlich nicht gerecht. Aus ihnen spricht vielmehr eine uralte Sehnsucht, dass die Welt der Maschinen noch unberührt und in Ordnung ist, wohingegen die Sphäre der Vernunft begabten Menschen allen Untergangsmysen getreu verderbt und entzaubert ist. In diesem Sinne leistet die Software ein Stück weit Aufklärerisches, und dies ohne auf das Spezialistenwissen der Macher von Internetangeboten zurückgreifen zu müssen.

#! /leandre

Joan Leandre: RETROYOU R/C, 2000 - 2001, <http://www.retroyou.org>

retroYou r/c basiert auf einem handelsüblichen Computerspiel. Der Künstler stellte die an der wirklichen Welt orientierten Spielbedingungen auf den Kopf, indem er ein Autorennen sukzessive umprogrammierte. Regeln, durch die Raum, Bewegung, Schwerkraft usw. simuliert werden, manipuliert er. Dabei bleibt das Spiel durchaus funktionstüchtig. In der merkwürdigen Bildwelt bleibt dem Benutzer nicht viel mehr übrig, als sich gelegentlich zu fragen, ob das Programm denn nun auf seine Eingaben reagiert hat oder nicht.

#! /rsg

RSG

Carnivore, <http://www.rhizome.org/carnivore/>

RSG ist ein Kollektiv aus international renommierten und handverlesenen Computer-Künstlern. Für Carnivore programmierten Künstler wie Mark Daggett, Entropy8Zuper!, Mark Napier und Praystation.

Carnivore (Fleischfresser) wurde durch einen Auftrag des FBI inspiriert, in dem es um die Programmierung einer Abhör-Software (DCS 1000 alias Carnivore) ging. Carnivore sollte den Datenverkehr des Internet erfassen und darstellen. Mit Hilfe weniger (kostenloser) Programme aus der open source-Szene hat RSG, eine Gruppe weltweit operierender Computer-Künstler, einen Programm-Cocktail zusammen gemixt, der dieselbe Aufgabe quasi zum Nulltarif übernimmt. Verschiedene net.art-Künstler haben zudem Visualisierungen des Datenverkehrs im Netz realisiert. Im Rahmen der Ausstellung wird ein Carnivore-Server permanent abgefragt, und es werden drei Darstellungsweisen gezeigt. Neben der augenscheinlichen Ästhetik wird auch ein politischer Aspekt offenbar: Freie Software wird zugunsten eines proprietären Systems und eines Millionen schweren Auftrags an eine Firma übersehen. Dabei ermöglicht der Carnivore einen ähnlichen Blick hinter die Kulissen von Datenströmen aber auch Abhörmechanismen wie der in seinen Funktionen ähnliche aber kommerzielle Konkurrent. Die Arbeit erhält im Rahmen der Diskussion um die Verwendung von Linux in der Bundesregierung und in öffentlichen Verwaltungen und Ministerien eine weitere Aktualität.

#! /schiessl

Simon Schießl, *1972, Berlin/D

Roter Tropfen, 2001, <http://www.pinkpicknick.de/TM/index.html>

Entgegen der Erfahrung des Sichtbaren zeigt der „Rote Tropfen“ die Bewegung eines Polygons. Der Betrachter sieht zunächst einen roten Ball auf sich zukommen. Er kann diesen mit der Maus bewegen. Gerät die Figur an den Rand der tief blauen Bild-fläche, zerplatzt der Ball und sein Ursprung aus einem Vieleck erscheint. Der Arbeit liegt ein Programm zu Grunde, das die Wirkung der Schwerkraft auf einen Ball simuliert. Ein Fehler in der Programmierung führte zu den künstlerischen Effekten, zu der Unbestimmtheit des Systems, denn anscheinend funktioniert es ja. Dass es aber nicht richtig agiert, sieht man ihm nicht an.

INSERT

Simon Schießl: Kurzbeschreibung des „Roten Tropfens“

Titel: Roter Tropfen

Autor: Simon Schießl

Sprache: Java

Entstehungsjahr: 2001

Kurzbeschreibung:

Der Code beschreibt die Bewegung eines einfarbig gefüllten Polygons. Es gibt zwei Modi: Klickt man auf die Bildfläche, nehmen die Eckpunkte Positionen entlang einer Kreislinie an, die Figur verformt sich zu einer Scheibe (Dragmodus). Ist die Maustaste nicht geklickt, verschieben sich die Aufhängungspunkte gemäß nicht linearer Bewegungsgleichungen (freier Modus).

Die Gleichungen für diese Figur sind ein Zufallsprodukt: Eigentlich sollte die Dynamik eines federnden Balls beschrieben werden. Ein Fehler in den numerischen Gleichungen ließ das System instabil werden. Den korrekten Code findet man unter www.pinkpicknick.de/ball/index.html.

Bis auf die Interaktionsmöglichkeit durch das Ziehen auf der Bildoberflaeche wurde der Code ursprünglich belassen, d.h. keine weiteren grafischen oder akustischen Effekte hinzugefügt. Er verweist somit an eine ganze Gattung einfach gehaltener, bewegter Grafikobjekte, zeichnet sich jedoch durch den eigenen Charakter der Bewegungsgleichungen und der User-Interaktion aus. Auch wenn der Bewegungsablauf willkürlich und chaotisch erscheint, erinnert das Fließen der Flächen an den Ursprung der mathematischen Nachbildung eines elastischen Objekts.

Programmtechnische Details

Das grafische Objekt ist ein sich änderndes, unregelmäßiges Vieleck mit 64 dynamischen Aufhängepunkten. Die 64 Eckpunkte modellieren Massepunkte, die über ihren räumlichen Abstand wechselwirken. Die Grundidee ist eine kreisförmige Ruhelage: Befinden sich alle 64 Punkte in gleichem Abstand R_0 und gleichmäßig über den gesamten Kreiswinkel 360° von einem 65ten Zentralpunkt entfernt, wirken keine beschleunigenden Kräfte. Bei einer Verschiebung eines Punktes relativ zu seinen drei direkten Nachbarn (zwei auf der Kreislinie und der Zentralpunkt) ziehen sich diese Punkte gegenseitig an oder stoßen sich ab, je nachdem ob sie sich näher kommen oder entfernen. Ist diese Kraft proportional zur Auslenkung aus dem Sollabstand, wird ein Gebilde simuliert, das aus $64+1$ Punkten besteht, die mit 128 Federn gegenseitig verbunden sind. Bei richtiger Wahl der Parameter entsteht eine einfache Simulation eines kompakten Balles, der seine scheibenförmige Grundform nicht weit verlässt und, wie man es von realen Bällen kennt, sich eindellt und vibriert, wenn er auf eine feste Wand aufschlägt, sich dreht und hüpfet.

Numerisch aufwendig ist die Berechnung der 128 Abstände zwischen den Punkten. Um diese Berechnung effizient auf einer 8-Bit-CPU ausführen zu können, wurde die Wurzelfunktion mittels einer Tabelle und einer Projektion in einen genormten Raum gemäß dem folgenden Quelltextausschnitt nachgebildet. Die dabei entstandene Nichtlinearität lässt die Ballsimulation instabil werden und führt zu chaotischen Bewegungsabläufen.

```
// Federkräfte und innere Dämpfung der äußeren Punkte untereinander
for(int i=0; i<(AnzahlPixelR1-1); i++){
    dx = x1[i] - x1[i+1];
    dy = y1[i] - y1[i+1];
    //L = Math.sqrt(dx*dx + dy*dy);
    L = Wurzelfunktion((int)(dx*128.0), (int)(dy*128.0))/128.0;
    Fx = (L1/L-1.0) * c1 * dx + (vx1[i+1] - vx1[i])*innereReibung;
    Fy = (L1/L-1.0) * c1 * dy + (vy1[i+1] - vy1[i])*innereReibung;
    ax1[i] += Fx;
    ay1[i] += Fy;
    ax1[i+1] -= Fx;
    ay1[i+1] -= Fy;
}

private double Wurzelfunktion(int dx, int dy){
    double L;

    if(dx < 0)
        dx = -dx;
    if(dy < 0)
        dy = -dy;
```

```

    if(dx == 0 && dy == 0){
        L = 0.0;
    }
    else if(dx >= dy){
        L = (double)(Wurzeltabelle[(512*dy)/dx]*dx) / 512.0;
    }
    else{
        L = (double)(Wurzeltabelle[(512*dx)/dy]*dy) / 512.0;
    }

    L = Math.max(L, 1.0);
    return (L);
}

```

mit:

```

for(int i=0; i<513; i++)
    Wurzeltabelle[i] = (int)Math.sqrt(16384 + i*i);

```

```

#####

```

#! /traub

Peter Traub, *1974, Menlo Park/USA
Bits & Pieces, 1999, <http://www.fictive.org>

Peter Traub ist Netzkünstler und lebt in der Bucht von San Francisco. 1999 schloss er sein Studium mit einer Arbeit über Elektro-Akustische Musik am Dartmouth College ab. Er komponierte zahlreiche elektronische Musiken und konzentriert sich gegenwärtig auf die Erkundung des Verhältnisses zwischen Internet, Klang und Musik. Traub arbeitet in Kollaborationen, u.a. mit seinem Bruder Gregory Traub oder Amy Alaalexander.

Peter Traub beschäftigt sich in der Hauptsache mit der digitalen Klangerzeugung. Er nutzt hierbei das Internet als eine Oberfläche, gleichermaßen aber auch als Reservoir und Materialspeicher. In der Arbeit „Bit & Pieces“ durchpflügt ein komplexes Programm das Internet nach Sounds und bereitet diese zu Playlisten auf. Der Algorithmus macht dabei keinen Unterschied zwischen Gesprochenem und Musik. Es ist lediglich eine Frage der Dateifindung. Nach einem aus maschineller „Sicht“ erfolgreichen Tag sollten damit pro Tag 25 Dateien bearbeitet worden sein. Die Dateien werden nach der formalisierten Suche und Umwandlung in ein Internet taugliches Format konvertiert und direkt auf den Webserver aufgespielt. Zudem wird beim Download des Nutzers noch eine weitere Zufälligkeitsmaschine in Gang gesetzt, da bei simultaner Nutzung niemals dieselbe Playlist erhältlich ist. Damit ist „Bits & Pieces“ ein höchst komplexes Verfahren, das Internet in seiner Unanschaulichkeit und Unüberschaubarkeit erfahrbar zu machen. Wenn der Benutzer mit seinem Lieblingsplayer die willkürlich addierten Listen hört, weiß er einerseits nicht, was ihn als nächstes Stück erwartet, andererseits ist er gezwungen, über die Kontexte der Arbeit nachzudenken. Kunsthistorisch verortet sich Traub selbst in die Reihe von Schwitters, dessen Prinzip der Collage er auch für seine Klangerbeiten nutzt und gleichermaßen in die Linie aleatorischer Kompositionsprinzipien, wie sie John Cage in den 60er Jahren des vergangenen Jahrhunderts geschrieben hat. Mit dem Unterschied, dass es hier weder eine Notation noch ein greifbares Material gibt, aus dem die Stücke sich addieren. Unveränderbar bleibt lediglich der permanent und - so lange Strom fließt - unabänderliche Prozess des Werkes selbst.

#! /ward

Adrian Ward, *1976, London, GB
Signwave Auto-Illustrator 1.0 (2002)
<http://www.auto-illustrator.com>

Adrian Ward arbeitet als Software-Künstler und kreiert generative Ton- und Videoarbeiten sowie Prozess basierte Anwendungen für eine Vielzahl von Zwecken.

Er beschäftigt sich mit Fragen nach Autorschaft und die Erweiterung ästhetischer Subjektivität in Programmcode, eine Aktivität, die seiner Einschätzung nach seit Beginn der Programmierbarkeit von Geräten existiert. Er arbeitet mit Musikern, Konzept- und Performancekünstlern zusammen und nutzt Software für Performances und Installationen. Zudem arbeitet er in pädagogischen Software-Projekten mit. Seine Arbeiten werden von Lovebytes (Sheffield), MediaSpace (Plymouth) und Rhizome (New York) publiziert. Er zeigte seine Software in der 291 Gallery (London) und dem New Museum of Contemporary Art (NYC). Zudem konnte er seine Thesen im Rahmen von zwei Konferenzen zu „International Generative Art“ (Mailand) vortragen und hielt Gastvorträge an der Cornish College of Art (Seattle). Zudem wurde er auf der transmediale 01 mit dem Softwareaward ausgezeichnet und erhielt eine Lobende Erwähnung im Rahmen des Prix Ars Electronica (Linz) in der Kategorie „Interaktive Kunst“.

Adrian Ward und seine Firma Signwave haben den Auto-Illustrator und Autoshop entwickelt. Die Programme können in der Ausstellung angewendet und ausprobiert werden. Beide docken sich bereits qua Namen an sogenannte Standard-Software an.

Als Kern zeigen beide Arbeiten, wie das eingesetzte Werkzeug über die Erscheinung der Grafik herrscht. Im Unterschied zur persiflierten Software beherbergen beide Programme Funktionen, die ihre Nutzung beeinträchtigen, die der BenutzerIn das Gefühl vermittelt, sie verliere die Kontrolle. Plötzlich läuft Ungeziefer über die mühsam erstellte Grafik. Eine Wanze (engl. Bug) ist die Regel bei aller Software.

Der Begriff bezeichnet den Effekt, den ein Fehler des Programmierers erzeugt. „The world is in beta“ war der Slogan der Technik gläubigen Kultgazette Wired, und Betaware birgt in der Regel eine Menge Bugs. Dies ist es, was NutzerInnen in der Regel am meisten erzürnt: Es sind diese kleinen Fehler, die zwangsläufig aber Folge der Produktionsbedingungen sind. Wards Produkte imitieren ihre kostspieligen Vorbilder bis zu den penetranten Aufforderungen zum Registrieren oder zum Download neuer Version. Mittlerweile ist der Auto-Illustrator in der Version 1.0 erhältlich. Und genau wie die kommerziellen Produkte der Software-Produzenten nervt beim Programmstart einer früheren Beta-Version die Erinnerung, dass es im Internet ein Update gibt. Der Standard-

Browser des Systems startet automatisch. Normales Arbeiten wird damit erst einmal zugunsten der Gewinnmaximierung hinten gestellt.