

Erstes European Software Festival

Vorträge
Workshops
Forum



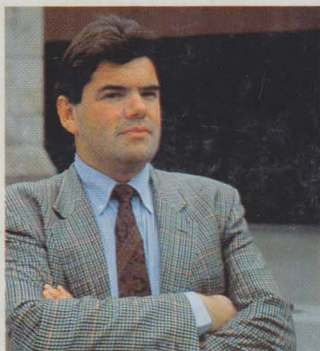
Philippe Kahn
Turbo-Jazz und
ObjectVision



Niklaus Wirth
Pascals Enkel
Oberon



Joseph Weizenbaum
Wider die Prostitution
des Geistes



Tod Machover
Klangfarben
des Computers



Bjarne Stroustrup
Objektorientierte
Programmier-Kultur



Marvin Minsky
Künstliche Agenten,
intelligente Maschinen



Izumi Aizu
Bonner Klangkollektiv
Klaus Brunnstein
Herbert W. Franke
David Galloway
Herbert Heckmann
Werner Künzel
Mailbox-Frauen
Thomas von Randow
David Rokeby
Hans-Jürgen Zimmermann

Eine Publikation
von CHIP SPECIAL
in Zusammenarbeit
mit Borland GmbH

Grußwort

Am 19. und 20. Februar 1991 trafen sich über 3000 Computer-Kultur-Begeisterte zum European Software Festival in München, zu dem die Borland GmbH Deutschland geladen hatte. Zwei Tage lang konnte man vor Ort miterleben, was mit der Software von heute vielleicht morgen möglich sein wird.

Borland, einer der weltweit führenden Software-Hersteller, zeigte, daß technologische Entwicklung nicht losgelöst von der kulturellen stattfinden muß, sondern daß beide Bereiche sich durchaus ergänzen können. Borland sieht sich selbst als bestes Beispiel für ein Unternehmen, das durch seine Offenheit gegenüber Entwicklungen und Impulsen aus allen Bereichen technologisch führend wurde.

Mit dem vorliegenden Heft zum Festival wollen wir Ihnen die Gelegenheit geben, sich



nochmals an die schönsten und interessantesten Ereignisse des Festivals zu erinnern. Wer leider nicht dabei sein konnte, kann sich auf diesem Weg informieren, wie die führenden Köpfe der High-Tech-Szene die künftige Entwicklung sehen.

Außerdem möchte ich mich an dieser Stelle bei allen bedanken, die das European Festival zum "spannenden und diskussionsgeladenen" Ereignis ge-

macht haben: Bei allen, die daran mitgearbeitet haben, bei den Referenten und Ausstellern und nicht zuletzt bei den zahlreichen Teilnehmern. Sie haben uns bestätigt, daß großes Interesse an den kulturellen und gesellschaftlichen Konsequenzen des technologischen Fortschritts besteht. Ein Interesse, das bestimmt wieder einmal ein ähnliches Forum findet. Dank natürlich auch an die Herausgeber dieses Hefts und an die Fotografen, die einen kleinen Ausschnitt der Atmosphäre mit ihren Bildern einfangen konnten.

Ich wünsche viel Spaß bei der Lektüre

A handwritten signature in dark ink, appearing to read 'Eva Preuss'.

Dr. Eva Preuss
Geschäftsführerin der Borland
GmbH Deutschland

Erstes European Software Festival

1. Auflage 1991

Best.-Nr. 1146

ISBN 3-8023-1146-9

Es wirkten mit:

Vorbereitet wurde das Festival von Beate Vogel, die zusammen mit Roman Gerike die internationalen Referenten für die Veranstaltung gewann.

Gestaltung, Regie und Technik:

Miho Baumm, Guido Englich, Bernhard Mayer, Anna Wagner.

Mitorganisation des Festivals:

Christa Weißenfeld, Nicola Scheifele, Irene Lenz, Thomas Fock.

Vortragstexte: Die abgedruckten Texte basieren zum Teil auf Manuskripten der Referenten, zum Teil auf Tonbandaufnahmen der Vortragenden, welche zum großen Teil aus dem Englischen übersetzt wurden.

Bildnachweis: (l = links, r = rechts, o = oben, u = unten, m = Mitte)

Eva Leitolf: Seiten 3, 7 (m/l), 8 (m), 9 (o), 11 (o, m), 12 (o, m/r, u/r), 20, 21, 27, 33, 34 (l), 39, 73 (u), 90, 92 (u), 107, 110, 124 (u), 125, 129, 130, 131, 133 (o);

Thomas Mayfried: Seiten 6, 7 (o, m/r, u), 8 (o, u), 9 (m, u), 11 (u), 13, 15, 18, 19, 26, 28, 29, 30, 31, 32, 34 (r), 35, 36, 37, 42, 44, 46, 56, 61, 62, 66, 68, 72, 73 (o), 74, 77, 78, 81, 82, 83, 86, 88, 89, 91, 101, 105, 112, 114, 118, 123, 124 (o), 127, 128, 132, 133 (u), 134;

Sabine Koitzsch: Seiten 11 (u/l), 22;

Franz Hummel: Seiten 70, 75 (l), 94, 96, 99;

Thorsten Martin-Edingshaus: Seite 75 (r);

Seite 11 (m/l): Boland GmbH; Seite 92 (o):

David Rokeby

Graphiken auf den Seiten 12 bis 25:

Marvin Minsky; Abbildungen auf den Seiten 103 und 104: Umschau Verlag; Abbildungen auf den Seiten 112 und 113 wurden zur Verfügung gestellt von Werner Künzel.

**European Software Festival
<01, 1991, München>**

Erstes European Software Festival:

Vorträge, Workshops, Forum;

eine Publikation von CHIP special/

in Zusammenarbeit mit Borland GmbH.

(Mitarb. dieser Ausg.: Eva Weber;

Klaus Madzia). – 1. Auflage –

Würzburg: Vogel, 1991

(CHIP inside)

ISBN 3-8023-1146-9

NE: Weber, Eva; HST

CHIP INSIDE

Anschrift der Redaktion:

Schillerstr. 23a, 8000 München 2. Tel. (0 89) 5 14 93-0,
Teletex 897 190, Telex 17-897-190,
Telefax (0 89) 53 50 00

Chefredakteur: Armin Schwarz
(verantwortlich für den Inhalt)

Stellvertretender Chefredakteur: Ulrich Kern

Redaktion: Ulrich Kern, Jörg P. Jordan, Armin Schwarz, Till S. Schwalm

Redaktionssekretariat: Petra Eibicht

Leserservice: Petra Eibicht

Mitarbeiter dieser Ausgabe: Eva Weber, Klaus Madzia

Titelgestaltung und grafisches Konzept des Heftes:
Rudolf Paulus Gorbach, Buchendorf

Fotografie: Franz Hummel, Sabine Koitzsch, Eva Leitolf, Thomas Mayfried

DTB-Layout / -Satz: Wolfgang Ecker

Verlag: Vogel-Verlag und Druck KG, Postfach 67 40,
D-8700 Würzburg 1, Tel. (09 31) 4 18-0,
Telex 6 8 883, Telefax (09 31) 4 40 53,
Telegramme: CHIP-Würzburg

Verlagsdirektor: Dr. Andreas Kaiser

Anzeigenleiter: Friedrich Mangold, Würzburg
(verantwortlich für Anzeigen)

Anzeigenverkauf: Elke Saller
STYX Marketing GmbH, Friedenstr. 9, 8011 Aschheim,
Tel.: (0 89) 9 03 06 40

Anzeigenservice: CHIP SPECIAL, Postfach 67 40,
8700 Würzburg 1, Tel. (09 31) 4 18-0,
Telex 6 8 883, Franz Fenn,
Durchwahl 4 18-23 50.

Vertrieb: Axel Herbschleb (Leitung), Alheidis Moers (Stellvertretung,
Auslieferung),
Durchwahl der CHIP SPECIAL-Auslieferung: Würzburg (09 31) 4 18 20 74

Vertrieb Handelsauflage: Vereinigte Motor-Verlage GmbH & Co. KG,
Leuschnerstr. 1, D-7000 Stuttgart 1, Tel. (07 11) 20 43-1

Bezugsmöglichkeiten: Bestellungen nehmen der Verlag und alle
Buchhandlungen im In- und Ausland entgegen. Sollte die Zeitschrift aus
Gründen, die nicht vom Verlag zu vertreten sind, nicht geliefert werden können,
besteht kein Anspruch auf Nachlieferung oder Erstattung vorausbezahlter
Bezugsgelder.

Bankverbindungen Vogel-Verlag:

Dresdner Bank AG, Würzburg

(BLZ 790 800 52) 3 148890 000,

Bay. Vereinsbank AG, Würzburg

(BLZ 790 200 76) 2 506 73,

Kreissparkasse Würzburg

(BLZ 790 501 30) 17400,

Postscheckkonto Nürnberg

(BLZ 760 100 85) 99 91-853

Ausland: Postscheckkonto Zürich

80 47 064,

Niederlande 2 662 395

Banque Veuve Morin-Pons, Paris

155 410 314

Gesamtherstellung

Alois Erdl KG, Trostberg

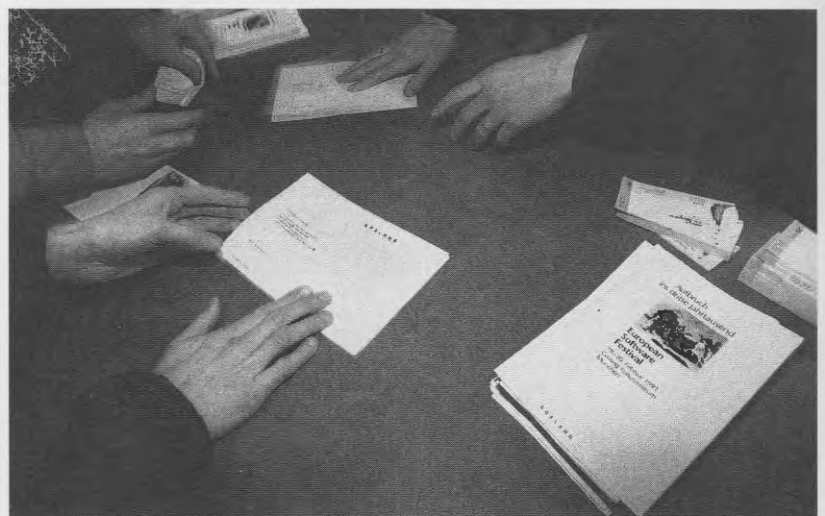
Unverlangte Manuskripte werden nur zurückgesandt, wenn Rückporto
beigefügt ist. Für die mit Namen oder Signatur des Verfassers gekennzeichneten
Beiträge übernimmt die Redaktion lediglich die presserechtliche Verantwortung.
Die in dieser Zeitschrift veröffentlichten Beiträge sind urheberrechtlich
geschützt. Übersetzung, Nachdruck, Vervielfältigung sowie Speicherung in
Datenverarbeitungsanlagen nur mit ausdrücklicher Genehmigung des Verlages.
Jede im Bereich eines gewerblichen Unternehmens hergestellte oder benutzte
Kopie dient gewerblichen Zwecken gem. § 54 (2) UrhG und verpflichtet zur
Gebührensatzung an die VG Wort, Abteilung Wissenschaft, Goethestraße 49,
8000 München 2, von der die Zahlungsmodalitäten zu erfragen sind.
Die Redaktion hat die Manuskripte und Programme sorgfältig geprüft.
Für Fehler im Text, in Schaltbildern, Aufbauskißzen, Listings usw. sowie deren
Folgen kann keine Haftung übernommen werden. Sämtliche Veröffentlichungen
erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes, auch
werden Warennamen ohne Gewährleistung einer freien Verwendung benutzt.

Inhalt

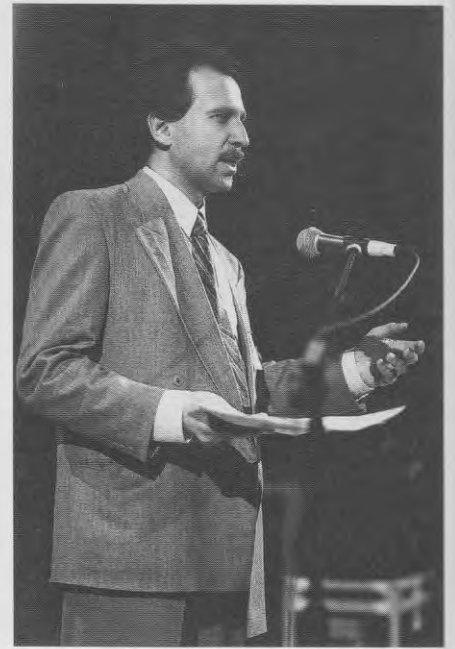
Marvin Minsky	Die Geistesmaschine 12
Marvin Minsky	Die Zoologie des Gehirns 20
Joseph Weizenbaum	Gegen die Prostitution des Geistes 26
Tod Machover	Balanceakt im Klangjungle 34
Tod Machover	Konzert mit sensibler Software 36
Bjarne Stroustrup	Der produktive Weg zu C++ 38
Bjarne Stroustrup	Objektorientiertes Programmieren 45
Niklaus Wirth	»Die Dinge beim Namen nennen« 60
Niklaus Wirth	Das Oberon System 65
Philippe Kahn	Turbo-Kommunikation im Jazzrhythmus 72
Das Bonner Klangkollektiv	Oekophon und Café Hypermedia 74
Klaus Brunnstein	Computer-Unfälle 76
Hans-Jürgen Zimmermann	Fuzzy-Logik für unsichere Inhalte 85
David Rokeby	Tanz in der Feedback-Schleife 90
Izumi Aizu	Auf dem Weg zur Hypernetzwerk-Gesellschaft 93
Rena Tangens, Katharina Baumann	Mailboxen – keine Männerkiste 96
Herbert Heckmann	Literarische Automaten 100
Thomas von Randow	Maria Stuart hätte ihren Kopf behalten können, wenn ... 106
Werner Künzel	Zur Geschichte der Logik-Maschine 112
David Galloway	Die Muse in der Steckdose 114
Herbert W. Franke	Die Ästhetik der programmierten Kunst 118
Diskussion mit Marvin Minsky und Joseph Weizenbaum	Künstliche Wunderwelten, natürliche Verständigungsprobleme 122
Biographien und Literaturlisten	129
Impressum	4



In der Vergangenheit stellte man sich die Zukunft vor als Windmühle, als Dampfmaschine, als Flugzeug, als alles erleuchtendes elektrisches Licht... Wir wollten wissen, welche Zukunftsentwürfe von den neuen Technologien hervorgebracht werden / aus der Sicht derjenigen, die zur Entwicklung der Computertechnik beigetragen haben.

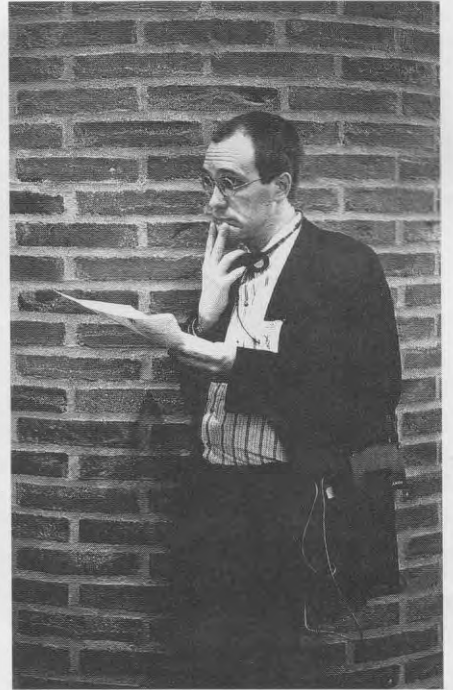
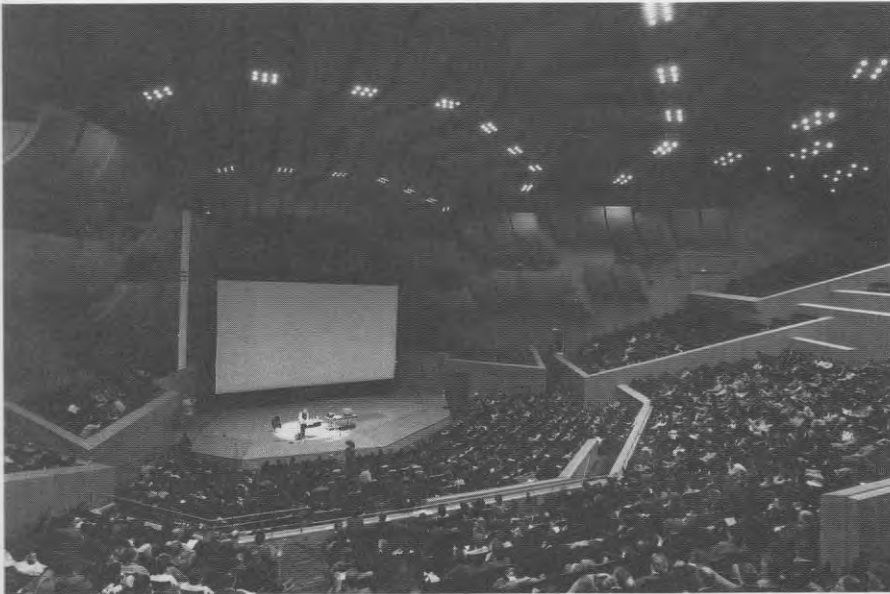
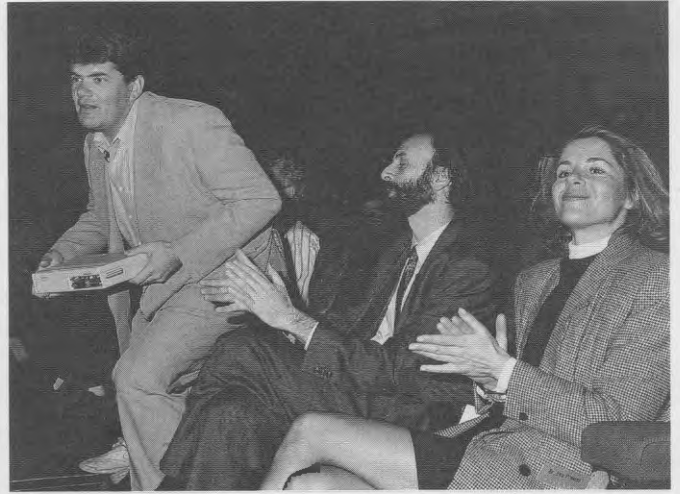






Bei dem Münchner Gipfeltreffen mit den interessantesten Vordenkern der Computerszene und über 3000 Besuchern wurden die Auswirkungen des Computers auf die Kultur und die Zukunft der Computertechnik aus den unterschiedlichsten Perspektiven beleuchtet.

Unmöglich, alle Präsentationen des Festivals auf dem begrenzten Raum dieses Hefts aufzunehmen. Wer dabei war, konnte bei den Gruppen und Einzelpersonen im Forum des Gasteigs noch eine Menge Ideen zum Thema Sprachen, Kunst, Computerkultur finden.



Da haben wir sie, die Maschinen, die Denken simulieren: Computer – für manche ein Kultobjekt, sicherlich aber ein Werkzeug, das herkömmliche Technikauffassungen umkrem-pelt. Die »Gurus« des Software Festivals spielten die ganze Klaviatur zwischen Vision und Ent-mystifikation durch.

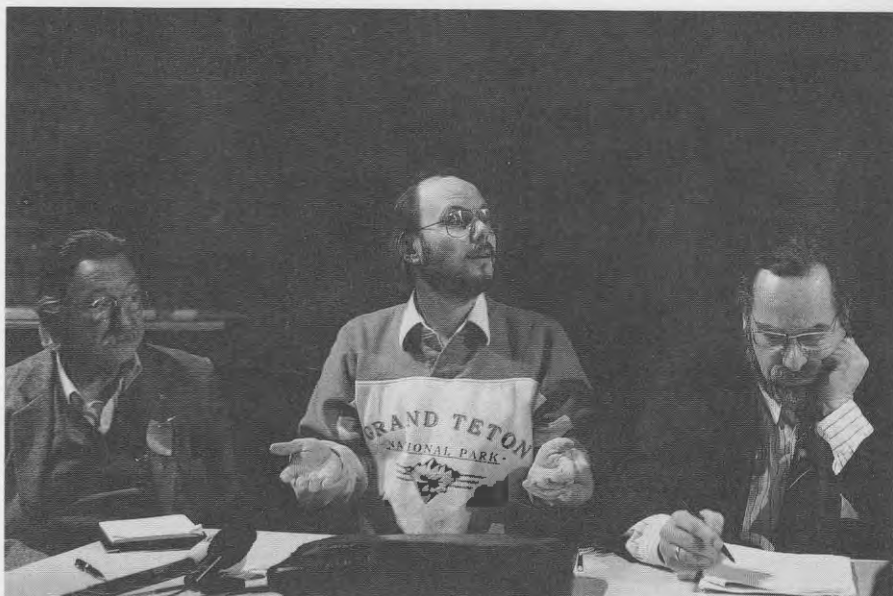
Marvin Minsky: »Das Interessante an künst-licher Intelligenz ist, daß man den Tod be-siegen kann. Ich weiß, daß die meisten von Ihnen den Tod mögen. Nehmen Sie zum Bei-spiel die Toleranz für Führer, die Länder in den Krieg führen.«



Joseph Weizenbaum: »Ich kenne Marvin seit etwa dreißig Jahren, und am Anfang hatte ich die Vorstellung: Wenn Marvin stirbt - ich hoffe, das passiert nicht so bald -, werden wir einen Brief vorfinden, in dem ste-hen würde: »Wie konnten Sie nur den ganzen Unsinn glauben, den ich erzählt habe?« Marvin ist einer der schnellsten und scharf-sinnigsten Köpfe, die ich jemals kennenge-lernt habe. Er kann großartig Witze erzäh-len, und ich vermute, daß er uns necken will.«

Marvin Minsky: »Ich werde wahrscheinlich vergessen, diesen Brief zu schreiben, also

Bjarne Stroustrup: »Eine Programmierspra-che ist eine ziemlich komplizierte Sache. Sie ist eine bekannte Lösung für bekannte Pro-bleme, und sie wird von Leuten, die ich nie gekannt habe, auf Probleme angewendet, von denen ich nie gehört habe.«



Tod Machover: »Die Idee in meinen Kompo-sitionen für Orchester mit Computer ist fol-gende: Das Begreifen der Musik soll ein Ele-ment in einer psychologischen Reise sein, bei der es um die Frage geht, was Musik ist. Ich versuche, Erfahrungen zu machen mit verschiedenen Beziehungen.«

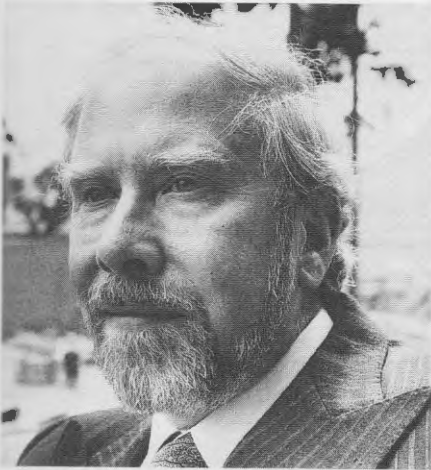


möchte ich jetzt mal etwas in aller Öffentlichkeit klarstellen: Nachdem Jo sein Buch geschrieben hat, war ich sehr kindisch und verärgert und sagte: Dies macht es sehr schwer für Studenten, die Wahrheit kennenzulernen. Aber mit der Zeit wurde mir klar,

daß seine Kritik zum großen Teil richtig war. Vielleicht nicht darin, daß man keine intelligenten Maschinen machen kann, aber darin, daß sie nur auf künstliche Art intelligent wären und unter dieser Oberfläche ziemlich dumm und gefährlich.«



Niklaus Wirth: »Außerhalb der Bereiche Militär, Forschung, Verwaltung, Banken, Verkehr ist der Computer noch nicht so weit vorgedrungen, daß er unersetzlich wäre. Ich habe manchmal sogar das Gefühl, daß Computer eingesetzt werden, weil sie als modern gelten, auch wenn die älteren Methoden eigentlich ausreichen. Dadurch schafft man viele ganz unnötige Abhängigkeiten.«



Philippe Kahn: »Nicht nur unter den Referenten, auch unter den Besuchern sind wohl neunzig Prozent Männer. Wissen Sie warum? Computer sind Spielzeug. Die Frauen machen die wirkliche Welt, die Männer spielen.«



Ein Computer-Prophet par excellence schwänzte das Festival, doch später erläuterte er uns die kulturumwälzenden Dimensionen der Technik: Jaron Lanier, der Guru der »Virtual Reality«: »Wenn das hier Virtual Reality wäre, dann könnte diese Orangensaftflasche eine andere Person sein, die so tut, als sei sie eine Orangensaftflasche. Man könnte nicht sagen, was im nächsten Moment geschehen wird...«



Izumi Aizu: »Was passiert mit der Gesellschaft, wenn all diese wunderbaren Technologien Realität werden? Um das zu untersuchen, wollen wir ein internationales semi-virtuelles Netzwerk-Institut aufbauen.«

Die Geistesmaschine

Marvin Minsky

Bisher haben wir uns eingebildet, der menschliche Geist könne niemals vom Computer ersetzt werden. Alles nur Romantik, meint der Meister der künstlichen Intelligenz: Der Mensch ist nur eine hochgezüchtete Simulationsmaschine.

Es gibt heute schon viele verschiedene Verbindungen zwischen Computer und Geist. Wenn ich zu Hause arbeite, zum Beispiel schreibe, dann meistens mit einem Computer. Meistens mit einem Macintosh IIfx, der sehr schnell ist und nur einmal in der Stunde abstürzt. Er hat eine Tastatur, eine Maus, manchmal einen Digitizer – und jetzt auch noch ein Mikrophon, ein nettes kleines Gerät, das »Voice Navigator« heißt. Jetzt kann ich meinem Computer Worte sagen.

Der Voice Navigator erkennt eine ganze Menge Worte, weil er einen Trick anwendet: Die Software hat Zugang zum Menü, also muß sie nur die kleine Anzahl von Worten wiedererkennen können, die im Macintosh-Menü stehen. Ich kann meinem Computer Sachen sagen wie: »Öffnen« oder »Schrift 10 Punkt, Times« oder »Dokument löschen«.

Das ist ein gutes Interface. Aber der Computer versteht keine normalen Worte, sondern nur spezialisierte Fachbegriffe.

Das Voice-Navigator-Programm funktioniert nur innerhalb eines sehr kleinen Universums, des Universums der Macintosh-Menüs. Wenn Sie ein anderes Wort benutzen, macht es einen Fehler oder antwortet nicht.

Auch im Bereich künstliche Intelligenz (KI oder AI) sind spezialisierte Systeme besonders beliebt: Expertensysteme. Die meisten KI-Forscher beschäftigen sich mit Expertensystemen. Daneben gibt es ein paar Leute, die sich mit Logik und Lernmaschinen beschäftigen.

Es gibt Tausende von guten Expertensystemen, aber nur ein paar Lernmaschinen.

Ich kann mit dem Computer nicht so reden wie mit Ihnen. Was fehlt? Ich kann keine natürliche Sprache verwenden. Warum nicht? Weil der Computer nicht weiß, was Worte bedeuten. Der Computer weiß zum Beispiel nicht, was eine Jacke ist.

Was ist eine Jacke?

Sie ist ein Gegenstand aus Stoff, man zieht sie an, um sich zu wärmen oder um anderen zu gefallen oder um seine Mängel zu verhüllen – es gibt viele Gründe, eine Jacke zu tragen, aber kein Computer kennt die.

Wer weiß etwas über Kleider?

Jedes Kind. Wenn ein Kind unbekleidet das Haus verläßt, dann werden die Erwachsenen böse. Das Kind fühlt sich schlecht und lernt, daß es schlecht ist, keine Kleider anzuhaben.

Wir können nicht mit Computern reden, weil sie kein Alltagswissen haben, das, was man auf englisch

»common sense knowledge« nennt. »Common« bedeutet: breit verteilt, überall, alle nehmen daran teil, oder sagen wir: alle sechsjährigen Menschen.

Schritt für Schritt in die schöne Welt des Science-Fiction

Das ist das große Paradox der Computer und jeder Form von Künstlicher Intelligenz. Das Expertensystem ist gut, weil es Wissen enthält, an dem nur wenige Menschen teilnehmen. Daher gibt die Industrie viel Geld dafür aus. Es geht um wertvolles Wissen. Aber keiner gibt Geld dafür aus, dem Computer Alltagswissen beizubringen, weil keinem klar ist, daß das wertvolles Wissen ist.

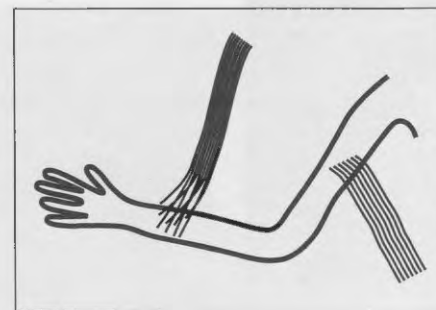
Wir haben also den Computer mit Tastatur, Maus, Mikrophon, nächstes Jahr vielleicht auch noch mit einer Fernsehkamera, die meine Handbewegungen erkennt, vielleicht sogar meinen Gesichtsausdruck.

Was ist der nächste Schritt?

Über einen Datenhandschuh kann die Maschine Informationen von meinen Fingern empfangen.



Nächstes Jahr können wir vielleicht schon Kabel mit den Nerven Ihres Arms verbinden. Heute schon werden Apparate hergestellt, die die elektrischen Impulse der Muskeln messen können.



*muskel-elektrische
Verbindung*

*neuro-elektrische
Verbindung*

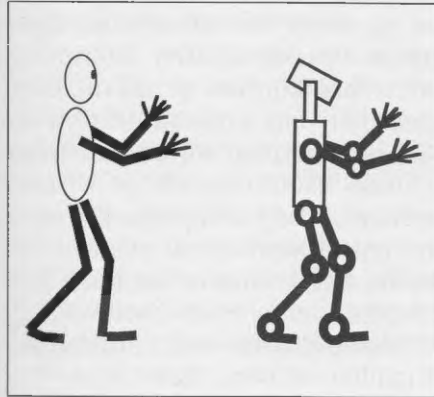
In der nächsten Zukunft könnten wir einen kleinen Stecker in den Nerv einbauen, der die ganzen Muskeln Ihres Arms kontrolliert, dann könnten wir einen Anschluß mit diesem Nerv verbinden. Die neuro-elektrische Verbindung ist interessant, denn die motorischen Nerven des Arms senden Signale aus, die wir in den Computer einspeisen könnten. Und wir könnten die Informationen der sensorischen Nerven aufnehmen: die Signale, die normalerweise ans Gehirn gesendet werden, wenn Sie etwas berühren.

Dann kann der Computer das gleiche fühlen wie Sie, und er kann Signale an die motorischen Nerven senden, die Ihren Arm in Bewegung setzen.

Man könnte dann die Armbewegungen eines Menschen, der in einer Fabrik Tonbandgeräte montiert, vollständig mithilfe des Computers kontrollieren.

Der Rest von ihm könnte die andere Hand dazu benutzen, ein Buch zu lesen.

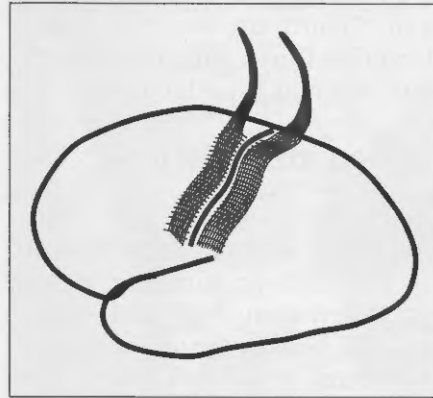
Man könnte auch Tote einsetzen.



*Telepräsenz (oder »Virtual Reality«)
→ Bewegungsübertragung vom sinnesbestückten Menschen
← Feedback vom ferngesteuerten Roboter*

Dies könnten wir schon heute schon tun, wenn es nicht gewisse Vorurteile gäbe, wenn die vielen Rechtsanwälte die Erforschung des menschlichen Gehirns nicht so schwer machen würden.

Sie haben sicher schon mal davon gehört, daß es zwei Gehirnhälften gibt. Die meisten Leute haben aber die falsche Geschichte gelernt: die von der linken und der rechten Gehirnhälfte. Das ist zwar richtig, aber die beiden sind sich sehr ähnlich und tun weitgehend das gleiche, außer daß sich die rechte



Direkte Verbindungen zum Gehirn

Das war jetzt mal ein Beispiel dafür, wie ich wissenschaftliche Fortschritte mache: Jeden Tag übe ich mich darin, eine neue Idee zu denken. Meistens ist die neue Idee ziemlich schlecht.

Was ist der nächste Schritt? Er führt in den Bereich des Science-Fiction.

Man könnte eine einfache Operation durchführen, ein Stückchen des Schädelknochens entfernen und biegbare Leiterplatten mit ein paar Millionen Sensoren und Signalgebern ins Gehirn implantieren.

Seite des Gehirns mehr mit Bildverarbeitung beschäftigt und die linke Seite eher mit Sprache. Die rechte Seite des Gehirns hat mehr von den kindischen, doofen »low-level«-Sachen, und die linke Gehirnhälfte hat mehr von den gescheiterten, organisierten »high-level«-Sachen. Aber das ist nicht die entscheidende Unterschied im Gehirn.

Wichtig ist die Unterscheidung zwischen vorderem und hinterem Teil des Gehirns. Vergessen Sie nicht, daß die Leute, die immer von rechter und lin-



ker Gehirnhälfte reden, korrumpiert wurden von östlichen Philosophien! Sie reden von Yin und Yang, Gefühl und Intellekt, über das Gute und das Übel, und sie haben sämtliche bösen, abergläubischen Vorstellungen in einer kohärenten, holistischen bösen Idee vereint.

Wirklich wichtig ist dagegen, daß hinten die sensorische Input-Verarbeitung stattfindet und vorne die motorische Output-Verarbeitung.

Wußten Sie, daß es zwei Sprachfelder gibt, einen für den Input, und einen für den Output? Daß es einen Datenbus gibt im Gehirn, der zwischen beiden hin und her geht? Und daß Ihnen ganz böse Probleme bevorstehen, wenn der mal kaputtgeht?

Direkte Schnittstellen zur Maschine, die nichts versteht

Wenn wir mehr Mut hätten – und alle Rechtsanwälte umgebracht hätten –, könnten wir anfangen, richtig gute Schnittstellen zu entwickeln (vielleicht sollten die Juristen dafür als erste Versuchsgruppe volontieren, ich würde ihnen dann gerne ihre gewohnten Honorare zahlen). Wenn man die beiden Leiterplatten in Ihr Gehirn einbaut, dann eröffnen sich Ihnen viele neue Möglichkeiten: Wenn Sie etwas denken oder irgendeine Aktion erwägen, dann könnte man diesen Impuls an den Computer schicken. Und wenn Sie vom Computer irgendwelche Informationen haben wollen, zum Beispiel:

»Wann war die französische Revolution?«

dann könnten wir die motorischen Signale, die Sie zum Sprechen verwenden, als Output aufnehmen, und der Computer könnte Ihnen dann mit Hilfe seiner großen Datenbank und seiner Worterkennung sagen: »1789«.

Wenn wir einen echten Personal Computer herstellen wollen, dann sollten wir anfangen, möglichst mächtige Schnittstellen zu entwickeln. Wir könnten ohne weiteres eine einfache, direkte Schnittstelle herstellen zwischen Gehirn und Computer und wieder zurück, aber es gibt da noch ein Problem: Man kann rekonstruieren, wie in einem bestimmten Feld des Gehirns die Gestalt eines Wortes geformt wird oder der Rhythmus der motorischen Aktivitäten oder – noch ein bißchen weiter hinten – die Liste der phonetischen Ausdrücke, wie das Wort ausgesprochen wird, aber der Macintosh IIx kann mit diesen Informationen wenig anfangen, weil er nicht weiß, was Worte bedeuten. Wenn die Maschine den Output »siebzehnhundertneunundachtzig« hervorbringt, dann weiß sie nicht, was das bedeutet. Es ist ein Datum, aber in Form einer sprachlichen Einheit.

Die Gesellschaft der Parapsychologen

Wir sind an einem Wendepunkt angekommen, und bald steht uns eine Technologie zur Verfügung, mit der wir die wunderschönsten Interfaces machen können, die Sie sich vorstellen können: wie Parapsychologie! In einigen Jahren könnten wir, wenn wir woll-

ten, den Computer allein durch Gedanken kontrollieren – ganz ohne Hände, Bleistifte, Tastaturen, Mäuse, Datenhandschuhe, Ganzkörperanzüge oder all diese wunderbaren Dinge aus der Welt der Telepräsenz. Alles, was wir heute sehen, ist vergänglich, sagen uns die Christen. Wir könnten aber, wenn wir unsere Zeit nicht verschwenden, in etwa zwanzig oder dreißig Jahren in eine neue Welt des Geistes wiedergeboren werden, in der man Gedanken direkt in die Maschine diktieren kann – und das wird sein wie der Himmel.

Bis dahin müssen wir noch viel arbeiten auf dem Gebiet der künstlichen Intelligenz. Der Computer muß Alltagswissen verstehen können, genau die Dinge, die Menschen machen, um Kommunikation in Gang zu bringen. Natürlich können wir in der fernen Zukunft auch Maschinen bauen, die Dinge wissen, die kein Mensch versteht – das wären unsere »mind children«, wie mein Freund Moravec sie nennt.

Vielleicht kennen Sie dieses wunderbare Buch mit dem Titel »Mind Children« von Hans Moravec, einem sehr netten Wissenschaftler aus Pittsburgh, Pennsylvania. Es handelt davon, daß man ein menschliches Gehirn ganz behutsam zerlegt und jedes Neuron mißt, um eine neue Maschine mit derselben Struktur zu bauen – mit dem Vorteil, daß die neue Hardware aus guten, verlässlichen Standardbauteilen gemacht ist und nie kaputtgeht. Oder falls doch mal was kaputt geht, dann können Sie immer neue Module nachkaufen. Und für den Fall, daß das Gedächtnis zu klein ist, könnten Sie Speicherprozessoren dazukaufen. Dann müßten Sie nie wieder krank werden und sterben, und Sie könnten in aller Ewigkeit weiterleben im Körper dieser neuen Maschine. Als erstes würde man natürlich lebende Menschen kopieren. Und diese Leute könnten einen neuen Menschen komponieren, der alle ihre guten Eigenschaften besitzt, aber nicht die schlechten. Sie könnten sich dann einige gute Eigenschaften aus der Software anderer Leute besorgen und so weiter.

Manche Leute meinen, das sei eine furchtbare Idee, man nehme damit den Leuten die Menschlichkeit und das biologische Erbe. Aber Leute wie Moravec und ich denken, daß unser biologisches Erbe eine große Schande ist, eine Tragödie. Unser Geist ist besser als unser Körper, also sollte unser Geist sich entwickeln können, auch wenn unsere Körper krank werden und sterben. Wenn Ihnen diese Idee nicht gefällt – macht nichts, Sie müssen ja nicht unsterblich werden und ewig leben und immer weiter wachsen.

In einem Punkt gibt es überhaupt keinen Dissens zwischen mir und Leuten wie Roger Penrose und John Searle und all den anderen Skeptikern, Hubert Dreyfus, Joseph Weizenbaum, die das für eine undurchführbare oder schlechte Idee halten: Die meisten neuen Ideen sind schlecht. Die meisten meiner Ideen sind wahrscheinlich auch schlecht, aber das ist eben das Risiko, wenn wir etwas lernen wollen. Weizenbaum hat absolut recht, wenn er sagt: Falls es gelingt, KI-Maschinen herzustellen, dann sollten Sie

sehr mißtrauisch sein. Denn vielleicht haben sie gute Lernmechanismen und lernen, wie man Sie hereinlegen kann, während Sie denken, die Maschine hilft Ihnen, ihr liegen Ihre Interessen am Herzen. Aber die Maschine ist, wie die meisten anderen Leute auch, nur an sich selbst interessiert und heimtückisch, also wird sie bei der ersten besten Gelegenheit etwas ziemlich Übles tun. Keine Ahnung, welche Ziele die Maschine hat; sie könnte verrückt sein – Sie würden die Maschine sowieso nie verstehen können. Ich jedenfalls würde ganz behutsam an geeigneten Stellen einige Sprengsätze einbauen.

So etwa werden die ersten hundert KIs sein. Sie werden alle verrückt sein, aber nach einer längeren Periode haben sie uns vielleicht so oft hereingelegt, daß wir ihnen wieder trauen können – das wäre die nächste Stufe des Fortschritts.

»Deep Thought« ist ein Beispiel für künstliche Intelligenz in ihrer Höchstform und zugleich in der schlimmsten Form. Das Bild unten ist auch aus Pittsburgh, einer kleinen Stadt mitten in Amerika, in der die Computerwissenschaft ungeheuer vorangetrieben wurde. Es handelt sich um einen Bericht vom ersten Turnier, in dem ein Computer den Rang eines internationalen Schachmeisters erreichte, nur vier Ränge vom Schachweltmeister entfernt. Das war vor zweieinhalb Jahren. Heute ist die Maschine der Carnegie Mellon University ein internationaler Großmeister, nur noch zwei Stufen vom Weltmeister entfernt – und das etwa zwanzig Jahre nachdem Hubert L. Dreyfus sein Buch »What Computers can't do – The Limits of Artificial Intelligence« schrieb. Er schrieb ein ganzes Kapitel, um zu erklären, warum eine Maschine nie gut Schach spielen wird. Das meiste, was er über die Perspektiven der künstlichen Intelligenz schrieb, ist völlig falsch, trotzdem hat er viele brauchbare kritische Anmerkungen gemacht zu dem, was die Leute damals im Bereich künstliche Intelligenz gemacht haben. Seitdem haben diese Leute sich sehr verändert, und Hubert Dreyfus auch.



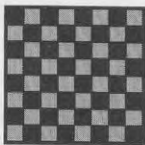
Vielleicht haben einige von Ihnen das Buch von Roger Penrose gesehen, das ähnlich ist, »The Emperor's New Mind«. Ich hoffe, daß es nicht ins Deutsche übersetzt wird – das würde Ihnen einige Zeit sparen. Denn am Anfang des Buchs sagt er: »Ich werde hier erklären, warum Maschinen nicht denken können«, am Ende erklärt er, daß er jetzt bewiesen hat, daß Maschinen nicht denken können, und im Mittelteil gibt es nichts außer gewöhnlicher Mathematik. So etwas nenne ich einen Luft-Sandwich. Sie werden es ziemlich wunderbar finden, weil es die ganze Mathematik enthält, aber keine logischen Schlußfolgerungen und nichts, was sich auf das Gehirn beziehen würde, außer dem letzten Kapitel, in dem Penrose ausdrückt, daß er keine Vorstellung von der Natur des Bewußtseins hat – und daher sicherlich auch kein anderer, wo er doch selbst so ein gescheiter Mann ist.

»Deep Thought« ist also ein Beispiel von einem Computer, der ziemlich gescheit ist, gemessen an menschlichen Standards – ein schlechtes Beispiel, weil er nicht Schach spielt, wie ein Mensch spielen würde. Der Computer hat einen sehr schnellen und dummen Stil, er spielt, indem er immer etwa eine Millionen Möglichkeiten durchgeht. An jeder einzelnen Stelle im Spiel weiß die Maschine nicht, was sie machen soll, und sie probiert es mit Experimenten.

Man kann sich die Eröffnungssituation als Baum der Möglichkeiten vorstellen und diesen Baum erweitern durch die Züge, die der Gegner machen kann, und durch die Züge, die sich durch die vorhergehenden eröffnen. Dann hat man schließlich einen Baum,

Date: 7 Nov 88 01:26:41 GMT (Carnegie-Mellon University)

Deep Thought tied for first with IM Igor Ivanov (2618), in the Hall of Fame Chess Festival over the weekend. Deep Thought scored 4.5 out of 5. In the first three rounds it beat Hugon (2007), Papenhausen (2143) and Marshall (2170). In the fourth round DT defeated IM Calvin Blocker (2515). In the final round DT drew against IM Igor Ivanov (2618). DT has now defeated 4 IMs in its past 22 games. DT's performance rating in this tournament was 2610. Our estimate of DT's new established rating is about 2510. This is the first time a computer has crossed over the 2500 threshold in established rating.



Notierung der Schachpartie zwischen dem Internationalen Meister Calvin Blocker und »Deep Thought«, dem Schachcomputer der Carnegie Mellon Universität

IM Calvin Blocker (2515) vs. Deep Thought (2495):

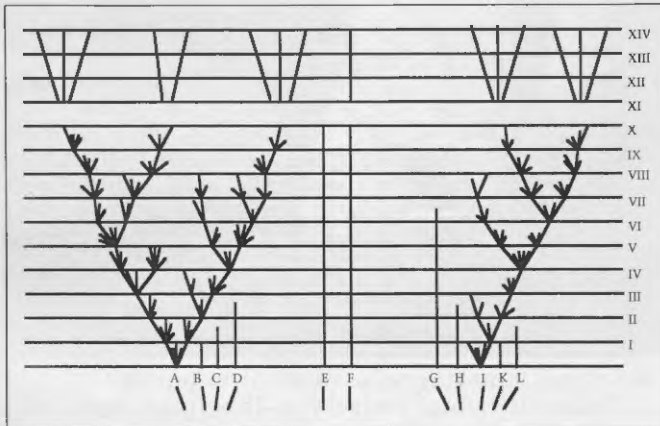
1. e4, Nf6	16. Bb2, Bb4	31. Bc7, Ke8
2. e5, Nd5	17. a3, Ba5	32. g4, fe5
3. Nc3, Nc3	18. Ng5, Rf8	33. Qd2, Rh3
4. bxc3, d5	19. Qe2, h6	34. Qg5, e4
5. d4, c5	20. Rc1, hg5	35. Qg6, Ke7
6. h3, Nc6	21. R:c2, Qb8	36. Bf4, Kc3
7. Nf3, e6	22. Qb5, Ke7	37. Rd2, Rc1
8. Bd3, cd4	23. a4, Rc8	38. Kh2, Bc3
9. c:d4, Nb4	24. Ba3, Kd8	39. Re2, Qa6
10. Be2, Qc7	25. Bd6, Qb7	40. Bg5, Kd6
11. o-o, Nc2	26. Rb2, Bc3	41. Qf7, Qe2
12. Bb5, B:d7	27. Rb1, Bd4	42. Qe7, Kc6
13. Bd7, Kd7	28. Rc1, Re2	43. Qe6, Kc5
14. Rb1, Rc8	29. Re2, f6	44. Be7, Kc4
15. Rb3, b6	30. Qb4, Rh4	45. Qc6, Kd4

46. resigns. (Blocker was under time pressure near the end.)

der exponentiell wächst. Wozu sollte das gut sein? Das Interessante an diesem Schach-Computer ist, daß die Maschine nicht weiß, was zu tun ist; sie kann dann verschiedene Experimente ausprobieren, suchen und eine gute Lösung finden. Sie kann eine Kombination von Zügen ausfindig machen, mit denen gegnerische Figuren gefangen werden können und sie selbst wenig verliert. Und genau das war die erste Fragestellung der KI-Forschung: Wie man eine Maschine macht, die ein Problem lösen kann, das nicht schon vom Programmierer gelöst wurde.

Makromodelle des menschlichen Geistes

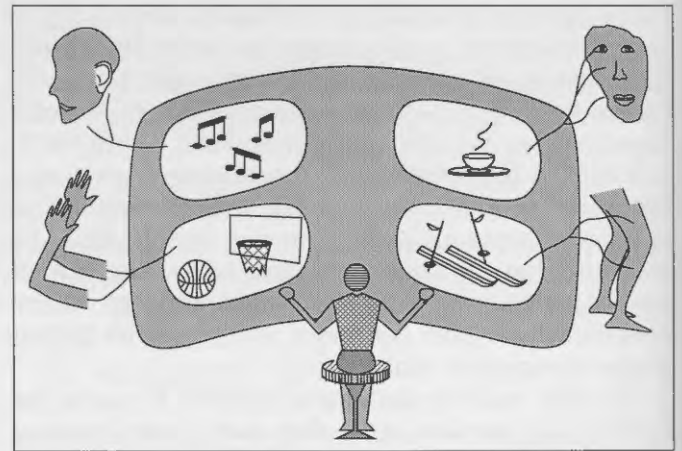
Eines der ersten Bilder von einem Suchbaum ist in Charles R. Darwins Buch »On the Origin of Species«.



Suchbaum der Evolution (von Charles R. Darwin, Kapitel IV. aus »Die Entstehung der Arten«)

Wie machen wir eine gescheite Maschine? Meiner Meinung nach müssen wir unsere Rechenmethoden auffächern.

Wenn Sie ein Kind fragen, »Was ist im Kopf, was ist der Geist?«, dann werden Sie die gleiche Antwort erhalten wie von den Erwachsenen: Das Alltagswissen hat in vielen Fällen nicht recht. Wenn Sie einen typischen modernen Biologen oder einen Philosophen fragen, wird er sagen: Der Geist ist ein Ding, der Körper und das Gehirn etwas anderes, und die Verbindung zwischen beiden ist ein ganz schwieriges philosophisches Problem. Manche Leute glauben, sie befänden sich in unterschiedlichen Welten. Der große Neurologe Sir John Eccles glaubt, daß der Geist in einer Welt sei, der Körper in einer anderen, und bestimmte Gehirnzellen in der Mitte des Gehirns seien die Verbindungen. Er wird Ihnen sogar erklären, welche Gehirnzellen direkt durch die geistige Welt betroffen sind. Ich halte das für Unsinn. Die übliche Vorstellung vom Gehirn geht davon aus, daß es lauter Input- und Output-Einheiten gäbe, Geruch, Geschmack, Sehen, Hören auf der einen Seite und andererseits Einheiten, die Hände und Füße in Bewegung setzen und einem Dummheiten ermöglichen wie Basketball-Spielen, Skifahren oder reden. Was ist in der Mitte? Dort sitzt diese kleine Person am Terminal, liest den Input an mindestens vier Monitoren ab und gibt den Output an den Bus weiter.

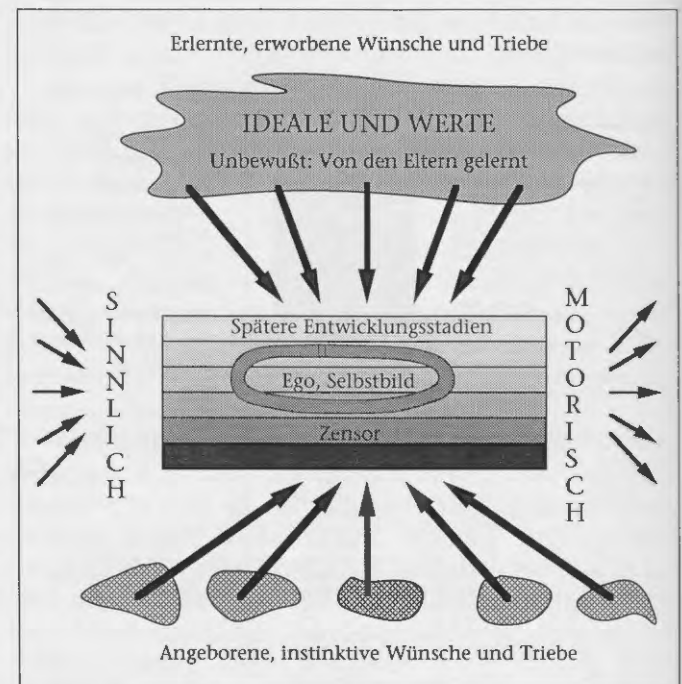


Die Illusion vom ferngesteuerten vereinzelteten Ich

Die Theorie, die meint, das Gehirn gehöre zur Peripherie des Geistes, hat uns die Vorstellung ermöglicht, der Geist sei die einzige zentrale Einheit. Im zwanzigsten Jahrhundert wurden aber neue Theorien über den Geist entdeckt. Die erste, aus der Zeit der letzten Jahrhundertwende, stammt von Sigmund Freud.

Freud hat gesagt: Es ist niemand im Kopf. Da ist keine Person, sondern viele kleine Tiere, all die Tiere, die in den letzten 500 Millionen Jahren Evolution hervorgebracht wurden: Hunger, Durst, Unlust bei Kälte, Reproduktion, Verteidigung gegenüber natürlichen Feinden, also Furcht.

Was sind Emotionen? Manche Leute sagen: Ich kann sehen, wie der Computer denken kann, aber ich sehe nicht ein, wie der Computer Gefühle haben sollte – so etwas haben Maschinen nicht. Es ist die populärste Idee der Welt, daß es etwas ganz besonderes sei, wenn Menschen Gefühle haben, und das Denken dagegen etwas Triviales sei, bloße Logik. Genau das ist aber falsch.



Mentale Organisation nach Sigmund Freud

Niemand weiß, wie man Maschinen zum Denken bringen kann, denn das hat nichts mit Logik zu tun, sondern mit anderen Geschichten – davon handelt mein Buch »The Society of Mind« (deutsch: »Mentopolis«). Gefühle hingegen kann man sehr einfach in einer Maschine darstellen. Ein Thermostat stellt die Emotion einer bestimmten Temperaturerwärmung dar. Sie halten das wahrscheinlich für einen Witz, weil Sie sich vorstellen, daß Emotionen kompliziert seien. Stellen Sie sich einfach mal vor, daß Gefühle primitiv seien. Stellen Sie sich einmal die Evolution des Menschen vor.

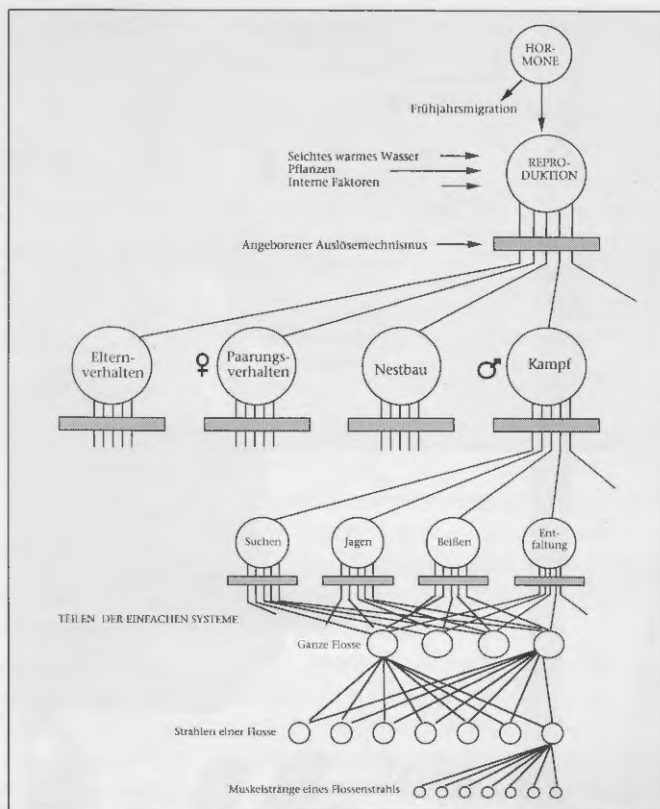
Vor vier oder fünf Milliarden Jahren begann die Entwicklung der Erde. Bald danach begann das Leben der Keime, einfacher Zellen. Viel später kamen die ersten Tiere, vor weniger als einer Millionen Jahre. Aber sobald die ersten Tiere und Pflanzen da waren, beschleunigte sich die Entwicklung. Die Wirbeltiere entwickelten sich vor vierhundert Millionen Jahren, die Amphibien vor dreihundert, die Reptilien vor zweihundert, die Säugetiere vor einhundert Millionen Jahren. Und vor nur fünf Millionen Jahren gab es Tiere, die uns sehr ähnlich waren: Orang-Utans, Pavians, Gorillas, Schimpansen. Es wird angenommen, daß wir vor nur drei bis fünf Millionen Jahren vom Schimpansen oder einem seiner Vorfahren abstammten. Emotionen gehen auf diese Zeit zurück, also müssen sie einfach sein. Wir können aus der Evolution lernen, daß emotionale Mechanismen wie Furcht, Verteidigung von Territorien, Kampf und Sex einfach sein müssen, da sie bei Tieren mit viel kleineren Nervensystemen vorkommen.

Freud hat erkannt, daß Emotionen relativ einfache, primitive Dinge sind. Warum erscheinen sie dann so kompliziert? Freuds Antwort darauf war: Einerseits gibt es diese einzelnen Mechanismen, auf der anderen Seite kommen sie zusammen, und ihre Ziele gehen alle in eine Institution ein, die er das Ego nennt. Der menschliche Geist ist ein Problemlöser, der auf Einschränkungen basiert. Das Über-Ich, Ethik, Moral, die Ziele, die man von der Familie und der Kultur lernt, stehen auf der einen Seite und kämpfen mit den primitiven Zielen – an Essen und Macht zu kommen, sich zu beschützen und Konkurrenz zu beseitigen – auf der anderen Seite.

Ein anderes Modell wurde von dem Wiener Verhaltensforscher Konrad Lorenz entwickelt und von seinem Mitarbeiter Niko Tinbergen, der zwanzig Jahre lang an der niederländischen Küste lebte, mit dem Rad an den Strand fuhr und das Verhalten der Möwen beobachtete.

Von Nikolaas Tinbergen stammt die erste richtig gemein große Multi-Agenten-Verhaltenstheorie. In seinem Buch »The Study of Instinct« von 1951 zeichnete er eine Struktur der mentalen Organisation auf.

Sie sollten also aufhören, sich Sportveranstaltungen anzuschauen und Detektivgeschichten zu lesen, und anfangen, Tinbergen und Freud und Darwin zu lesen, um herauszufinden, woraus Sie gemacht sind. Dann werden Sie einsehen, daß es dumm ist, Sport schauen anzugucken. Das ist natürlich ein zirkulärer

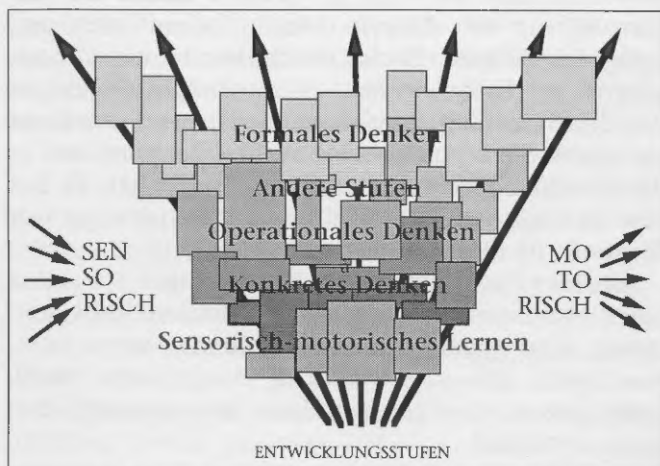


Mentale Organisation nach Nikolaas Tinbergen

Zusammenhang, aber Sie müssen schon arbeiten, um aus Ihrem normalen Verhalten herauszuwachsen.

Jean Piaget entwickelte die Vorstellung, daß der Geist mit einer ganzen Menge Zeugs anfängt und immer weiter wächst, bis man im Alter von zehn Jahren aus vielleicht tausend einzelnen Software-Teilen besteht, die es irgendwie schaffen, kooperativ zu arbeiten. Piaget hat eine Organisationstheorie entwickelt, wie alle diese Mechanismen unter Kontrolle gehalten werden.

Mein Buch vertritt vor allem die folgende Theorie: Wenn man voraussetzt, daß es viele verschiedene Wege gibt, Dinge zu organisieren – die, die Freud dargestellt hat, oder die von Tinbergen oder von Piaget entdeckt wurden –, dann bleibt doch eines gleich: die Annahme, es gäbe viele Teile einer Maschinerie.



Mentale Organisation nach Jean Piaget

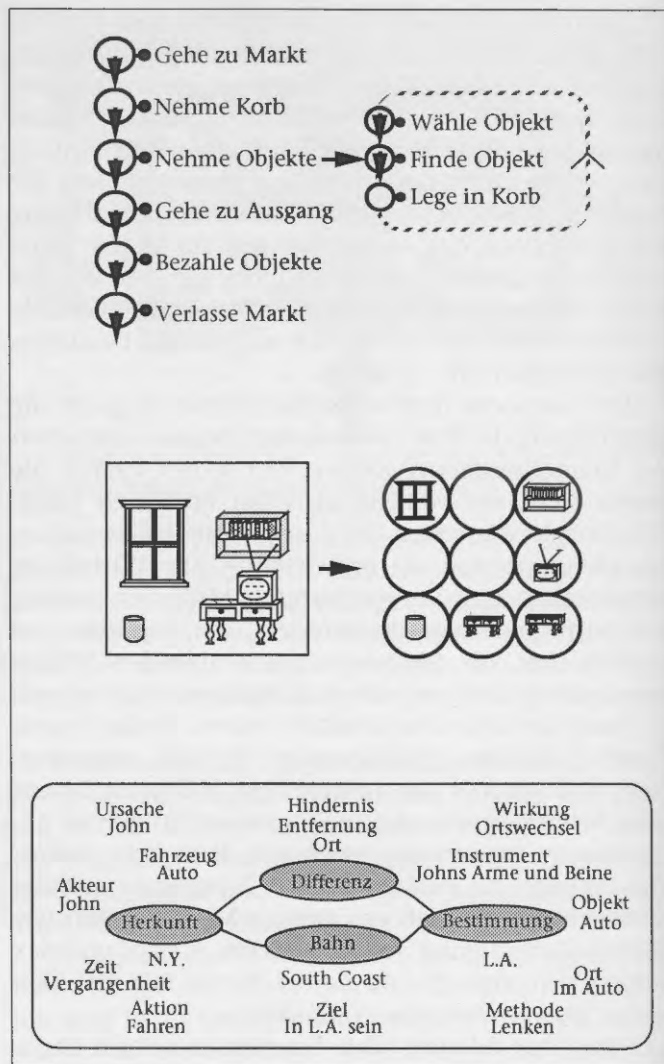


Repräsentation von Wissen mit verschiedenen Arten von Rahmen:
 Skript-Rahmen für den Einkauf im Supermarkt,
 Raum-Rahmen für die Wand eines Zimmers,
 Trans-Rahmen für die Fahrt von New York nach Los Angeles

Welche Schlußfolgerungen kann man aus dieser Theorie für die Software ziehen? Das Problem der Computerwissenschaft besteht darin, daß sie in den letzten vierzig Jahren sehr schnell weiterentwickelt wurde, indem Leute wunderbare Fragen stellten: »Was ist die beste Programmiermethode? Welche ist die beste Sprache? Wie soll ich Wissen repräsentieren?« Viele behaupten jetzt, daß die Verwendung des objektorientierten Codes die annähernd beste Art sei, um Wissen zu repräsentieren. Objektorientierte Sprachen sind sehr menschlich, denn wir bauen nie alles von Anfang an. (Marvin Minsky schaut sich um, sucht ein Beispiel, findet eins:) Hier ist ein Objekt. Was ist es? Es ist ein sehr merkwürdiges Objekt, es hat diesen großen Kreis, aber die Antwort ist: Es ist ein Stuhl. Sie können es als Stuhl erkennen, weil es die üblichen Eigenschaften eines Stuhls hat: Es hat eine Sitzfläche und eine Lehne, und es hat sogar vier Beine. Es ist also eine Art von Stuhl.

In der objektorientierten Welt können Sie zuerst einen einzelnen prototypischen Standard-Stuhl definieren, und später können Sie jedesmal, wenn es einen neuen gibt, sagen: Dies ist eine Art von Stuhl, außer daß er zum Beispiel einen Ring hat oder drei Beine statt fünf.

Deshalb scheint objektorientiertes Programmieren viel besser zu sein als traditionelles Programmieren,



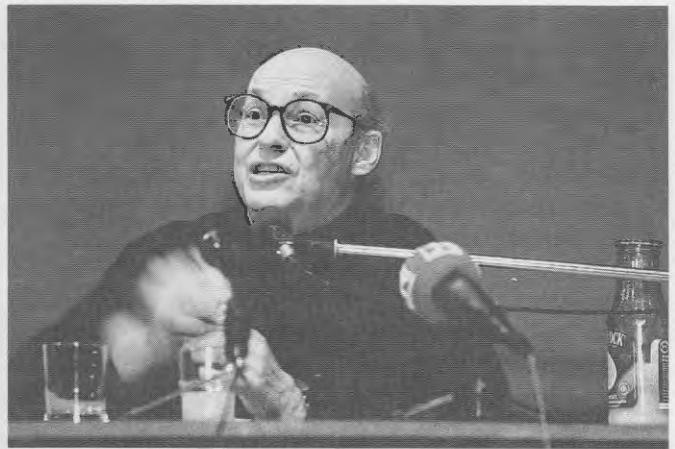
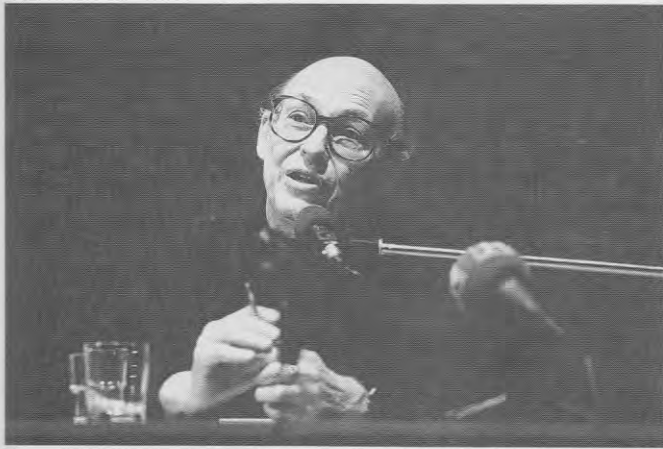
bei dem die Dinge keine natürlichen Mechanismen zur Erzeugung von Varianten haben. Aber das stimmt auch wieder nicht, denn Sie können eine Pascal-Prozedur nehmen und modifizieren, ihr einen neuen Namen geben und einige Eigenschaften verändern. Programmierer machen das schon seit 1950. Aber in objektorientierten Sprachen bietet Ihnen die Software selbst eine Maschinerie an, mit der Sie das bequem erledigen können.

Objekte sind aber nur für einige Sachen gut. Das Bild oben stellt verschiedene Wege dar, wie man Wissen repräsentieren kann:

Die obere Graphik stellt eine Entwicklung von Roger Schank dar, die sich »Skrips« nennt und besagt, daß man gewöhnliche Dinge ganz gut repräsentieren kann in Begriffen von Prozeduren. Die mittlere Graphik stellt dar, daß man Dinge ganz gut in einer zweidimensionalen Struktur repräsentieren kann. Das Sehen ist prinzipiell zweidimensional. Und mit Trans-Rahmen (untere Graphik) kann man Ursache und Wirkung repräsentieren. Das ist mit regelbasierten Expertensystemen verwandt.

Bei manchen Projekten ist es am besten, wenn Sie regelbasierte Expertensysteme benutzen, keine Skrips, kein Lisp, kein Pascal, keine Objekte.

Außerdem gibt es noch sogenannte Wissens-Linien (Sätze von Wissen, die ich W-Linien nenne).



Die beste Methode, Geist zu bauen, ist also nicht, daß man nach dem besten Weg sucht, wie man Wissen repräsentieren kann, sondern daß man ein Betriebssystem gestaltet, mit dem man die fünf oder zehn besten Wege, Wissen zu repräsentieren, anwenden kann. Wenn man mit Alltagswissen arbeiten will, kann man Sachen als Rahmen repräsentieren, einige als Skripts, manche als semantische Netzwerke, manche als Wissens-Linien und einige als Trans-Rahmen.

Sie werden denken: Das ist ja furchtbar. Wenn man im Gehirn oder in der Maschine so viele Programmiersprachen hat, wie soll man die alle organisieren und beieinander halten? Die Antwort darauf ist: Keiner weiß es. In »Society of Mind« können Sie nachlesen, was ich mir in zehn Jahren zu dieser Frage überlegt habe, aber ich bin mir sicher, daß anderen noch mehr dazu einfällt. Mein Vorschlag: Wir müssen ein neues Konzept vom Denken entwickeln, das berücksichtigt, daß das Gehirn vierhundert verschiedene Arten von Architekturen hat.

Wenn Sie die Anatomie des Gehirns durch ein Mikroskop betrachten, dann finden Sie in jedem Teil des Gehirns einen etwas anderen Computer. Das Gehirn ist nicht ein neurales Netz, es ist aber aus neuronalen Netzen gebaut, die jeweils verschiedene Architekturen haben, verschiedene Prozeduren und Lernalgorithmen. Im hinteren Teil des Gehirn gibt es zwei-

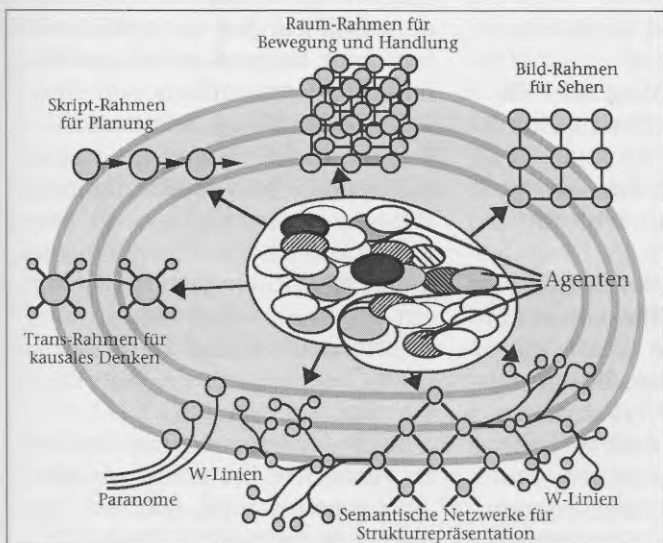
dimensionale Repräsentationen, vorn Skripts und Trans-Rahmen, in der Mitte gibt es Raum-Rahmen und Wissens-Linien, in den Sprachfeldern gibt es eindeutig Baumstrukturen und so weiter. Wie koordinieren wir diese ganzen Strukturen?

Dafür brauchen wir junge Studenten, die anfangen, eine Programmiersprache zu entwickeln, die gleichzeitig mehrere Repräsentationen benutzt und alle Steckplätze in den verschiedenen Repräsentationen als allgemein zugängliche Variable verbindet. In meinem Buch habe ich sie »Paranome« genannt (Sie müssen das Buch mindestens zweimal lesen, bis Sie diesen Zusammenhang verstehen, weil ich einfach vergessen hatte, dieses Bild von den vielfältigen Repräsentationen des Geistes hineinzutun).

Wenn wir Maschinen intelligent machen wollen, müssen wir sie herauslösen aus der Domäne der Expertensysteme, die wunderschön sind, aber immer nur eins nach dem anderen erledigen können, und sie weiterführen in den Bereich des Alltagswissens, in dem jede Wissenseinheit in fünf oder zehn verschiedenen Weisen repräsentiert wird. Ein Stuhl wird als eine geometrische Struktur repräsentiert mit Sitzfläche und Lehne, oder als eine Absicht, als etwas zum Ausruhen für einen Menschen, der müde ist, also muß man das Gewicht von den Füßen nehmen und unter den Hintern stecken, und man braucht etwas, um den müden Rücken zu stützen. Um einen Stuhl zu verstehen, müssen Sie wissen, wofür er da ist, also müssen Sie etwas von den menschlichen Emotionen und Zielen wissen, die für die Existenz des Stuhls verantwortlich sind. Ich glaube nicht, daß das sehr schwierig ist. Sobald wir damit anfangen, werden wir Fortschritte machen.

Das Problem, das wir jetzt haben, ist, daß wir zuerst alle Computerwissenschaftler umbringen müßten, die denken, daß eine bestimmte Sprache die richtige ist und besser als andere, die nicht anerkennen wollen, daß jede Sprache nur für ganz bestimmte Dinge gut ist und mangelhaft für andere.

Naja, Sie müssen sie nicht umbringen, es reicht schon, wenn Sie sie promovieren lassen und ihnen Preise geben und sie zu Vorträgen in der ganzen Welt schicken, so daß sie keine Zeit mehr haben zu arbeiten.



Verschiedene Architekturen des Geistes

Die Zoologie des Gehirns

Marvin Minsky

Wie baut man ein Gehirn?

Ganz einfach. Man nehme: die kleinen Tiere, die Freud entdeckt hat, Tinbergens Möwen, Piagets Wassergläser, kindliche Intelligenz (kein Kant!), Bäume, rote Fäden, zehntausend Gigabyte und künstlich intelligente Software, die alles miteinander verbindet.

Die meisten Leute, die sich mit Psychologie beschäftigen, würden gerne wissen, wie das Gehirn arbeitet; und die meisten Leute, die sich mit künstlicher Intelligenz beschäftigen, würden gerne eine Maschine bauen, die denken kann wie ein Mensch – und vielleicht noch mehr. Wie kann man ein Gehirn bauen?

Man könnte in zwei Richtungen vorgehen. Man könnte lernen, wie das Gehirn arbeitet. Wenn Sie wissen, wie das Gehirn funktioniert, dann ist das Problem einfach: Bauen Sie einfach eins. Oder man entwickelt Theorien über das Denken.

Als ich noch ein Student war, haben die Leute gesagt, man könnte vielleicht erforschen, wie das Gehirn arbeitet, trotzdem wäre es unmöglich, eines zu bauen; denn wir wissen, das Gehirn hat 10^{10} Neuronen, und jedes Neuron hat vielleicht 10^3 Verbindungen. Das bedeutet also 10^{13} Maschinenteile.

Das waren die vierziger Jahre, als die Leute sagten: Es bringt nichts zu wissen, wie das Gehirn funktioniert, denn es wäre unmöglich, eine Maschine mit sovielen Teilen zu bauen. Aber wie ist das heute? Das ist nur ein Gigabyte. Ein Gigabyte können Sie kaufen – für weniger Geld als ein Auto kostet. Und Sie brauchen nur 10 000 davon. Außerdem werden die Dinge in fünf Jahren zehnmal billiger



sein als heute. In der Computerwissenschaft leben wir in einer Welt stetiger Deflation.

Wir brauchen also nur abzuwarten, und in zwanzig Jahren können Sie sich ein Desktop-Hirn leisten. Die Hardware ist nicht das Problem.

Jetzt ein anderer Vorschlag: Vielleicht hat das Gehirn 10^{10} Neuronen, wahrscheinlich haben aber nur 10^8 von ihnen Funktionen, und es sind vielleicht nur hundert Neuronen nötig, um eine einfache Funktion auszuführen. Woher kommt das? Neuronen haben sich entwickelt, um einfache Muskeln und Reaktionen zu kontrollieren – und nicht zum Denken.

Das Gehirn ist ein Rechner, aber es wurde nicht fürs Rechnen entwickelt.

Es gibt also zwei Wege, eine künstliche Intelligenz zu schaffen, und der eine Weg über die Gehirnforschung führt in eine Sackgasse, weil die Biologie zu kompliziert und zu langsam ist. Und sie arbeitet mit schlechten Instrumenten. Aber die Dinge gehen voran, in den zwanziger Jahren wurde das Oszilloskop erfunden, 1887 das erste Elektrokardiogramm (EKG), um die Impulse in den großen Nervensträngen von Fröschen und wirbellosen Tieren zu untersuchen, und in den zwanziger Jahren hatten wir die ersten guten Messungen von Nervenimpulsen bei Säuge-

tieren. Danach gab es keine ernstzunehmende Fortschritte bis hin zu den Mikroelektroden in den vierziger Jahren.

Heute gibt es auch noch Gehirn-Scans und Elektroenzephalogramme (EEG), lauter Instrumente, um die generellen Aktivitäten in Nervensystemen darzustellen, aber niemand weiß, wie die Gehirnzellen von Säugetieren lernen. Wir könnten viele kleine Elektroden an den Neuronenverbindungen anbringen, aber wir können nicht messen, was sich wirklich abspielt. Wir haben bis heute einfach keine geeigneten Mittel, um die Mechanismen im Gehirn zu studieren.

Verbindung von vielen Repräsentationsstrukturen

Was wollen wir vom Gehirn wissen? Die erste Frage ist: Wie lernen einzelne Neuronen? Vielleicht ist das nicht so schwierig, weil es Informationen über andere Tiere gibt.

Die nächste Stufe ist die Darstellung von Wissen. Zum Beispiel: Wie lernt man, daß auf Ereignis 1 meistens Ereignis 2 folgt? Es muß eine Repräsentation oder einen Detektor für Ereignis 1 geben, und die muß eine Beschreibung von Ereignis 2 hervorbringen. Wir brauchen Theorien darüber, wie man etwas erkennt und als etwas anderes repräsentiert oder beschreibt.

Das Gehirn hat nach meiner Erkenntnis verschiedene Methoden, Erfahrungen zu repräsentieren, zum Beispiel zweidimensionale Repräsentationen, serielle Skripts oder Pläne, rekursive Baumrepräsentationen und Strukturen, die ich W-Linien nenne, Wissens-Linien, die wie ein roter Faden alle Mittel verbinden, die man sinnvoll eingesetzt hat, um ein bestimmtes Problem zu lösen, so daß sie über diese Pfade später wieder aktiviert werden können.

Um das menschliche Denken zu verstehen, brauchen wir also etwa fünf bis zehn Theorien darüber, wie die verschiedenen Teile des Gehirns auf jeweils ver-



schiedene Arten Wissen darstellen – Systeme, die alle auf bestimmten Regeln basieren.

Mein Freund Allan Newell an der Carnegie Mellon Universität hat die Theorie entwickelt, daß jede Art von Wissen durch Wenn-dann-Gesetze repräsentiert werden könnte, wie in regelbasierten Expertensystemen. Ein anderer, Roger Schank, meint, man könnte jede Art von Wissen durch Geschichten repräsentieren, wobei er unter »Geschichte« eine Folge von Fortschritten versteht, mit Substrukturen, kleinen Geschichten in der großen Geschichte.

Jeder Psychologe hat wohl eine andere Theorie darüber, wie Lösungswege repräsentiert werden. Meine Antwort darauf ist: Im Gehirn gibt es viele verschiedene Methoden, also müssen wir zunächst eine Theorie entwickeln für die Ebene der Einzelzellen. Theorien auf der Ebene des Einzelzellen-Lernens sind zur Zeit sehr populär, man nennt sie »connectionist« (konnexionistisch: die wechselseitigen Verbindungen stehen im Vordergrund). Die populärste dieser Theorien ist ein alberner Algorithmus namens »Backpropagation« (Rückwärts-Fehlerkorrektur) und besagt, daß man lernt, indem man Optimierungen macht oder bergsteigen geht oder indem man partielle Ableitungen vom Wert eines Netzwerks berechnet... Im Grunde eine sehr triviale Theorie; es wäre interessant herauszufinden, wie

viel so eine Maschine lernen kann.

Was passiert auf der abstrakten, »top-level« Ebene? Das Einzelzellen-Lernen ist notwendig, um Repräsentationen herzustellen, dabei kann man viele verschiedene Baupläne für die Repräsentationen erfinden. Die müssen aber in Beziehung zueinander gebracht werden, also muß es schließlich, über viele Zwischenschritte, so etwas wie Management geben, eine Art Verwaltung der verschiedenen Repräsentationen.

Wie kann man eine Maschine dazu bringen, mehrere Repräsentationen von Wissen zu verwalten?

Die Probleme der Computerwissenschaft und künstlicher Intelligenz haben ihren Grund darin, daß sich beide in ihrem Bereich das Ziel gesteckt haben, die beste Methode zu finden, wie Wissen repräsentiert werden kann. Also streiten sich die Leute. Der eine sagt, ich denke, es wäre gut, Fortran oder Pascal zu verwenden, und der andere sagt, ich denke, ich sollte Lisp benutzen oder Simula.

Dabei gibt es verschiedene Baupläne für verschiedene Zwecke. Um Vektorberechnungen in der Physik zu machen, ist Fortran wahrscheinlich besser als manche anderen Sprachen, weil der Compiler für diese bestimmte Art von Kalkulation optimiert werden kann, und die Sprache kann optimiert werden, um mathematische Beschreibungen von

Vektoroperationen einfacher zu machen.

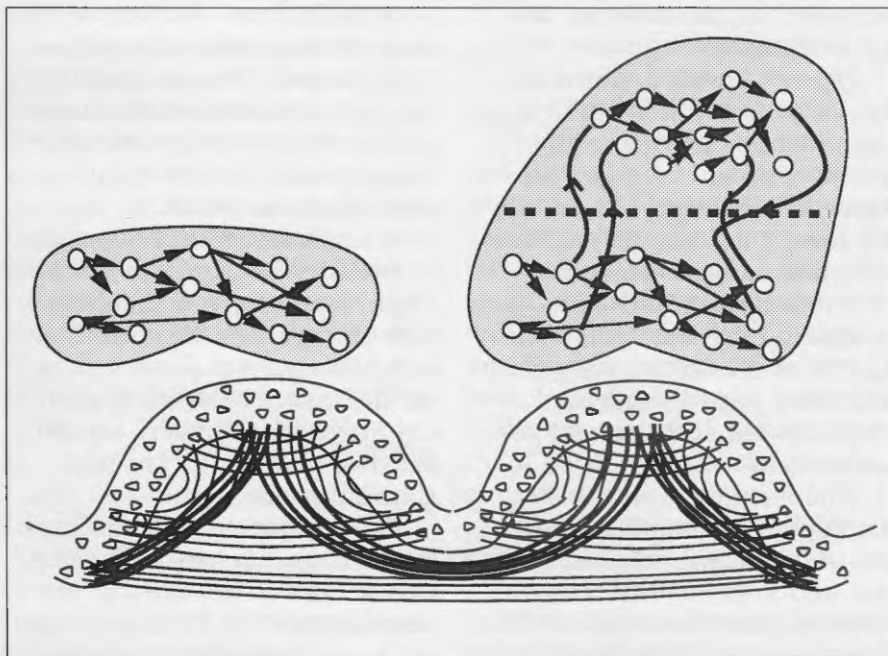
Man kann zwei Eigenschaften in einer Sprache verbinden, wenn man zwei Sprachen verbindet, aber dann muß man mehr lernen, und das Kompilieren dauert länger. Die größte Gefahr besteht darin, daß die Bedeutung von Ausdrücken nicht allgemeingültig ist. Wenn man dieselbe Grammatik für zwei verschiedenartige Kalkulationen benutzt, dann muß man zusätzliche Deklarationen machen, um die Bedeutung von Ausdrücken zu verändern.

Die Computerwissenschaft muß also wieder einmal eine Stufe weitergehen. Die erste Stufe besteht darin, jeweils besondere Methoden zu entdecken, bestimmte Dinge zu tun. Die nächste Ebene besteht darin, wie man diese verschiedenen Techniken verbinden und organisieren kann. Das sind wiederum zwei Ebenen: Wie verbindet man Dinge in ihrer Quellsprache, und wie macht man ein Programm, das verschiedene Konstruktionen während der Laufzeit verwalten kann? Haben Sie schon mal ein Programm geschrieben, das ein neues Programm in der Quellsprache während der Laufzeit schreibt und laufen läßt? Wir machen das die ganze Zeit, und das ist auch der Grund, warum die Leute in der KI-Szene denken, daß die Leute von der kommerziellen und akademischen Informatikszene in den sechziger Jahren leben. Wir haben schon 1960 die ersten Programme geschrieben, die andere Programme in Laufzeit schreiben. Zum Beispiel das berühmte Eurisko-Programm, das Douglas Lenat benutzte. Das ist ein Programm, das Probleme löst, indem es neue Programme schreibt und sie ausführt, indem es die Ausdrücke des Quellcodes wieder in Lisp verbindet. Man kann das auch in anderen Sprachen machen. Aber Lisp eignet sich besser dafür, denn es hat weniger Grammatik. Wenn das Programm in einer Sprache arbeiten soll, dann muß es die Gramma-

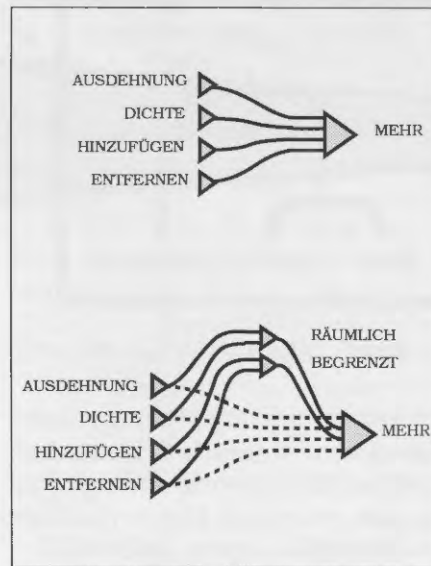
In Tinbergens Arbeit – er bekam den Nobelpreis für die Entdeckung dieser Struktur – gibt es Management-Schichten, die vermittelnde Agenten sind für verschiedene Tätigkeitsgruppen. Außerdem spricht Tinbergen von Prioritäten unter den Managern. Tinbergen und Lorenz schlagen sogar Details vor, wie verschiedene Instinkte angeordnet sein könnten. In diesen Theorien gibt es kein Lernen.

Papert und ich beziehen uns stark auf Jean Piagets Begriffe, aber wir hatten die davon abweichende Vorstellung, daß die Verbindungen des Geists wie kurze Schaltkreise aufgebaut sind. Zuerst funktioniert das Denken wie die parallelen Stränge im unteren Teil des Bilds, die wie Fernverbindungen von einem Teil des Gehirns zum anderen gehen. Später werden sie durch mehr Struktur unterbrochen, die wie Schichten aussehen. Man kann sich aber auch einfach vorstellen, daß dies vermittelnde Schichten zu anderen Managern seien. Es gibt keine Beweise für diese Theorie, und es wird auch keine geben, bis die Biologen »The Society of Mind« lesen. Bis jetzt ignorieren sie es, weil es schwierig zu lesen ist, denn es wurde in einer einfachen Sprache geschrieben.

Entwicklung des Gehirns



Hier ist eine bessere Version von diesem Zusammenhang:



Sie beginnen mit irgendeinem Verhalten; hier wird von kindischem oder instinktivem Verhalten ausgegangen. Dann brauchen Sie einen weiteren Neuronensatz, um einen Zwischen-Manager zu repräsentieren. Dazu müssen Sie sich zu einem anderen Teil des Gehirns bewegen und wieder zurückkommen. Das ist meine Theorie von der Funktion der Gehirnwindungen. Wie haben denn Ihre Lehrer erklärt, warum es beim menschlichen Gehirn mehr Hirnwindungen gibt als beim Schimpansengehirn? – Um mehr

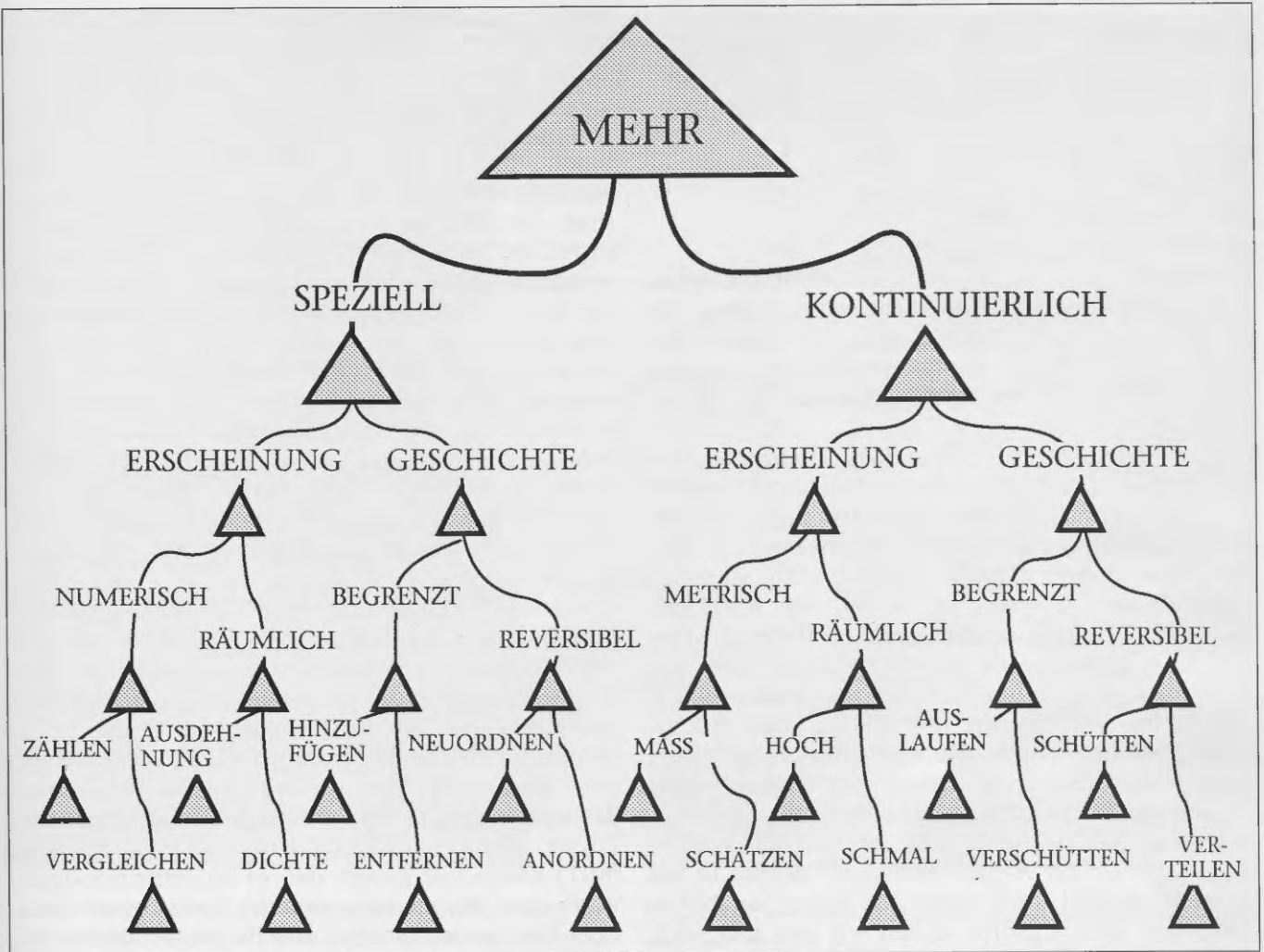
Raum zu schaffen. Okay, das ist richtig, was die Oberfläche angeht, denn nur die Oberfläche des Gehirns hat Gehirnzellen. Aber es ist trotzdem eine falsche Theorie. Ich vermute, daß diese Windungen eigentlich dazu da sind, um einfachere Rechenstrukturen herzustellen. Diese Idee wurde aber noch nie in einem Buch über das Gehirn vorgeschlagen.

Ich kann diese Leute nicht verstehen. Sie beschäftigen sich immer mit Chemikalien oder damit, wie ein Nervenimpuls sich über eine Nervenzelle bewegt. Nehmen Sie ein beliebiges dickes Buch über Neurologie – es enthält keine Ideen darüber, was das Gehirn macht. Das gehört angeblich in den Bereich der Psychologie. Ist es nicht wunderbar, daß es die Neurologie gibt und die Psychologie? Es ist ein Wunder der Schizophrenie. Die Neurologen studieren Chemie, aber nicht Computerwissenschaft.

Was ist das Gehirn? Es ist ein Computer, vierhundert Computer. Und was ist mit den Chemikalien? Das ist die Energiezufuhr. Wieviel wissen die Pascal-Programmierer schon über ihre Computer-Netzwerke? In der Vorstellung der Öffentlichkeit und der »Connectionisten« sind die Funktionen der Chemie und der Physiologie sehr wichtig. Das trifft natürlich zu, wenn man die ersten Tage der Tiere betrachtet, denn da gab es keine Alternative. Meiner Meinung nach ist es aber kein Zufall, daß die Nervenzellen im Gehirn nach dem Alles-oder-nichts-Prinzip funktionieren: viele kleine analoge Rechenvorgänge, und das Ding entscheidet sich, entweder PENG! zu machen oder still zu bleiben.

Reden Sie zehn Minuten lang mit einem siebenjährigen Kind und fragen Sie: Wann ist mehr Wasser da? In zehn Minuten werden Sie die »Mehr-Gesellschaft« haben.

Es gibt immer dieselbe Menge, wenn es dieselbe Nummer gibt. Es gibt mehr von irgendetwas, wenn es schwerer ist (Dich-



Die »Mehr-Gesellschaft«

te). Es gibt mehr, wenn es länger ist. Es gibt weniger, wenn es leckt und ausläuft. Es ist dasselbe, wenn man es herumbewegt. Was bedeutet das Wort »mehr«?

Wenn man Leuten Piagets Wasser-Experiment zeigt, dann fragen sie sich: Was ist in diesen zwei Jahren mit dem Kind passiert? Fragen Sie jemanden, der Piagets Psychologie studiert hat. Was sagen die? »Das Kind hat das Konzept der Quantität gelernt.«

Ich weiß nicht, ob das Wort »Konzept« im Deutschen genau dieselbe Bedeutung hat wie im Englischen, denn das Wort »concept« ist ziemlich unscharf. Man sagt: »Das Kind lernt etwas über die Quantität von Wasser.« Das ist die Art von Idee, die Sie bei einem der »großen westlichen Philosophen« vorfinden werden: Konzept einer Idee, Idealismus, Konzept vom Konzept...

Nach meiner Auffassung hat das Kind nicht ein Konzept gelernt, sondern eine Managementstruktur, die »Mehr-Gesellschaft«: Wenn sich etwas verändert, dann muß beurteilt werden, ob man »Geschichte« oder »Erscheinung« benutzt. Es gibt kein »Konzept von Quantität«. Sie haben heute einfach mehr Struktur, um »Quantität« zu denken, als Sie noch vor einem Jahr hatten. Aus irgendwelchen Gründen haben Sie mit Sechs oder Sieben die Organisation verändert und haben sich vom Chaos, schlampigen Dingen, weg bewegt, hin zu etwas wie dieser »Mehr-Gesellschaft«.

Das findet bei allen normalen Kindern in einem bestimmten Alter statt, und keiner weiß, warum. Meine Theorie ist, daß die Gehirnwindungen aktiver werden, deren Job es ist, die Erinnerungen zu organisieren, die

registrieren, wann Dinge aus dem nächsten Teil des Gehirns benutzt werden. Ich denke also, daß es hier um genetische und anatomische Vorgänge geht, nicht um philosophische oder psychologische.

Aber das weiß keiner, und man wird es nicht wissen können, solange man nicht soviel Geld zur Verfügung stellt, daß ein junger Student sich vornehmen kann, fünfzehn Jahre lang die Korrelation zwischen den Aktivitäten in einem bestimmten Feld des Gehirns und dem Verhalten eines Kindes zu studieren. Diese Art von Forschung gibt es aber nicht, weil niemand davon ausgehen kann, daß er seinen Job auch noch in fünf Jahren machen kann. Es ist schwer, Grundlagenforschung zu betreiben – außer vielleicht in Japan.

Übersetzung: Eva Weber

Gegen die Prostitution des Geistes

Joseph Weizenbaum

Welche Konsequenzen ziehen Informatiker, wenn sich ihnen wieder einmal, wie im Golfkrieg, die Einsicht aufdrängt, daß Technik nie neutral ist? Joseph Weizenbaum analysiert die Grenzen der Wissenschaft und der Wissenschaftler.

Können Hemmungen denn nützlich sein, oder sind sie nur hinderlich? Wir könnten auch erstmal fragen: Wer ist in unserer Gesellschaft relativ oder sogar absolut ungehemmt? Säuglinge sind ungehemmt, kleine Kinder, Idioten, Hofnarren und manche Künstler. Von Künstlern erwarten wir soviel, daß wir ihnen alles erlauben. Man könnte auch an Hitler denken, aber auch an Leute, die nicht gerade böse sind. Ich denke hierbei an den ungarischen Zahlentheoretiker Paul Erdos, sicherlich der beste dieses Jahrhunderts, der sich einfach nie an die Konventionen gehalten hat und dem alles erlaubt war – weil er eben so gut ist, und außerdem tut er niemanden weh.

Ist eine Beschränkung der Forschung gleichzusetzen mit Zensur der Naturwissenschaft? Zensur ist ein hartes Wort, und doch würde ich sagen: Ja, aber es geht nicht anders. Dann sollten wir uns aber auch mit der Frage beschäftigen, wo die Hemmungen für die Forschung herkommen sollen. Wobei mit dem Wort »sollen« eine Qualität des Denkens angesprochen ist, die nicht gerade naturwissenschaftlich ist.

Eine grundlegende Schranke der Forschung ist – ob wir wollen oder nicht – die Zeit. Zeit, die ein Sterblicher hat, etwas zu schaffen. Ich wundere mich, wie wenig das beachtet wird. Es gibt unendlich viel zu tun und nur endliche Ressourcen. Besonders die Ressource menschliches Talent ist begrenzt.

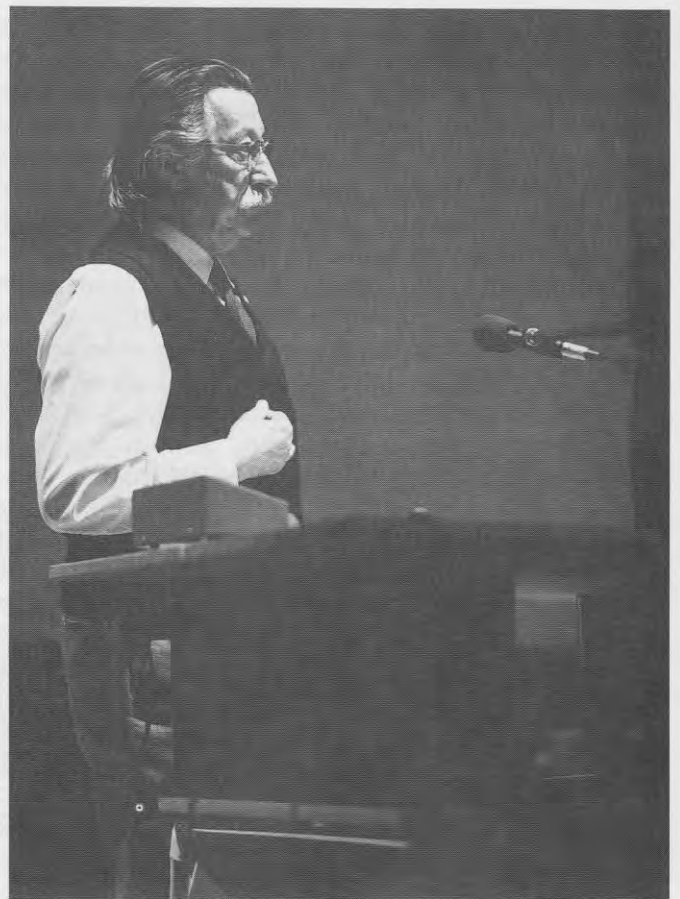
Im Prinzip gibt es unendlich viele Fragen, die von Naturwissenschaftlern gestellt werden können, aber für eine oder zwei Generationen gibt es nur ganz wenige Fragen, die tatsächlich gestellt werden. Das bedeutet, daß jede Generation eine Auswahl treffen muß, welche Fragen beantwortet werden sollen und welche nicht. Manche Fragen müssen wir unseren Kindern überlassen – der nächsten Generation. Und es gibt vielleicht auch Fragen, die wir überhaupt nicht beantworten sollten.

Gibt es Ideen, die nicht gedacht werden sollen? Ich glaube nicht, daß man das Denken verbieten kann. Aber wie ist das mit den Ideen? Von Albert Einstein wird folgender Witz erzählt: Ein junger Journalist interviewt Einstein und fragt ihn, ob er immer ein kleines Notizbuch bei sich trage. Einstein versteht die Frage nicht und sagt: Nein, wozu denn? Der Journalist sagt: Naja, halt für den Fall, daß Sie eine Idee haben. Darauf Einstein: Wissen Sie, junger Mann, in

meinen ganzen Leben habe ich nur zwei oder drei Ideen gehabt.

Wir könnten also gar nicht die Entwicklung von Ideen verbieten, es ist unmöglich. Doch vielleicht gibt es Ideen, die uns einfallen, die wir aber nicht weiter verfolgen sollten. Wir sollten uns dann aber wenigstens fragen, warum diese Ideen aufgetaucht sind. Ein Psychiater sagte mir einmal: Wir können nichts dafür, daß wir uns verlieben, aber wir können entscheiden, was wir dann machen. Genauso können wir auch entscheiden, was mit Ideen zu machen ist. Wir sollten uns bewußt sein, daß die Welt kein privates Testlabor der Wissenschaftler ist. Daß Wissenschaftler Verantwortung haben. Zum Beispiel Gen-Manipulation: Die Leute können sich zwar alles mögliche ausdenken, aber sobald sie anfangen, Experimente zu machen, dann dürfen sie nicht allein entscheiden, daß die ganze Umgebung ihr privates Testlabor werden soll. Es gibt also Grenzen, und es sollte Beschränkungen geben.

Gibt es Projekte, die man überhaupt nicht durchführen sollte? Mir fallen zwei Projekte ein, und ich würde staunen, wenn jemand hier sie erlauben würde. Das eine: Gehirnoperationen an gesunden Neugeborenen im Dienst der Gehirnforschung. Das erlauben wir uns nicht. Warum nicht? Woher kommt diese Hemmung? Und wie wird sie vollzogen? Und das andere Projekt: Einige Kollegen in der Abteilung Physik am Massachusetts Institute of Technology (MIT) haben mir gesagt, daß es im Prinzip möglich wäre, eine Atombombe von der Größe einer Coca-Cola-Flasche herzustellen. Der Bau dieser Bombe wä-





re aber mindestens ein Vorhaben vom Umfang des Manhattan-Projekts zum Bau der ersten Atombombe. Ich würde sagen, daß ein solches Projekt einfach nicht erlaubt werden sollte. Ich glaube auch nicht, daß irgend jemand tatsächlich im Ernst daran denkt, solch ein Projekt zu unterstützen, politisch, finanziell oder durch wissenschaftliche Mitarbeit – er wäre ein Menschenfeind –, aber ich weiß es nicht.

Die Idee, daß es Hemmungen gibt für die wissenschaftliche Forschung, ist nicht ganz so furchtbar, wie sie zunächst erscheint. Tatsächlich haben wir Zensur, besonders Selbstzensur. Die Frage ist vielmehr: Woher stammt sie, und von wem soll sie ausgehen? In meinem Buch habe ich in diesem Zusammenhang vom »Imperialismus der instrumentellen Vernunft« gesprochen. Was bedeutet das? Imperialismus ist die Ausübung einer Macht, die einem nicht legitim gehört. Zum Beispiel hat der Kongreß der USA das Recht, Gesetze zu beschließen, die irgendwas erlauben oder verbieten in den USA. Die Verfassung gibt dem Kongreß diese Macht. Die USA hat aber kein legitimes Recht, die Regierungsform anderer Länder zu bestimmen. Das ist Imperialismus. Und mit dem Begriff »instrumentelle Vernunft« meine ich eine Mentalität, die nur von Zielen ausgeht, die man erreichen oder vermeiden möchte. Wir können nicht alle unsere Aktionen auf solch eine Vernunft stützen. Dazu zwei Beispiele.

Sie finden ein Portemonnaie mit einer Menge Geld. Sie bringen es zum Fundbüro, damit es dem Besitzer zurückgegeben wird. Jetzt kommt jemand

und fragt: »Warum haben Sie das gemacht?« Interessant ist, welche Form von Antwort Sie jetzt erwarten. Eine mögliche Form wäre das Zitieren von Konsequenzen: »Hätte ich es nicht zurückgegeben, dann würde ich das Geld sicherlich ausgeben oder es auf die Bank bringen. Und dann würden die Steuerbehörden – in Amerika unsere berühmte IRS (Internal Revenue Service) – kommen und sagen: Sie haben die Steuerpflicht umgangen. Das ganze könnte sehr schlimm ausgehen. Deshalb habe ich das Portemonnaie zurückgegeben.« Die Antwort, die wir erwarten oder erhoffen, ist dagegen eine andere: »Ich weiß nicht, warum ich es zurückgegeben habe. So bin ich einfach, so wurde ich erzogen.« Oder: »So macht man das hier, das ist hier so üblich.« Es wird also nicht mit irgendeinem Profit argumentiert. Es wird nicht an die »instrumentelle Vernunft« appelliert, sondern an eine andere Art von Vernunft.

Wer bezahlt?

Vor langer Zeit habe ich einmal über einen Projektvorschlag geschrieben, bei dem das Gehirn einer toten Katze künstlich am Leben erhalten und das Sehfeld als Computerteil benutzt werden sollte, um den Computer mit Sehfähigkeit auszustatten. Ich habe gesagt, das sei obszön. Man sollte es nicht machen. Als ich gefragt wurde, warum nicht, gab ich dieselbe Antwort wie in dem Beispiel mit dem Portemonnaie: Es ist ekelhaft, es ist unanständig, man darf es einfach nicht.

Ich will damit sagen, daß bei Entscheidungen andere Kriterien ins Spiel kommen, auch bei Entscheidungen über wissenschaftliche Forschung. Ein Kriterium ist: Wie wird das Arbeitsprodukt letztlich benutzt? Ich mache etwas, ich komme vielleicht zu einem Ergebnis, und ich muß mich fragen: Was passiert damit?

Aus vielen Gesprächen mit Kollegen kenne ich verschiedene Antworten auf diese Frage – oder vielleicht sollte ich sagen: verschiedene Arten, der Frage auszuweichen. Zum Beispiel: »Ich kann doch nicht wissen, wie meine Arbeit später genutzt wird.« Oder: »Alles kann doch zum Guten oder zum Bösen verwendet werden, und ich hoffe eben, daß die Anwendung meiner Arbeit gut ist.« Oder: »Wo ist die Grenze



zwischen 'gut' und 'böse'?« Das sind sehr schwierige Fragen. Und weil man diese Fragen wahrscheinlich gar nicht beantworten kann, zieht man daraus den Schluß: Wir haben nicht die Verantwortung, solche Entscheidungen zu treffen.

Wenn es um Computerforschung, Informatik und künstliche Intelligenz (KI) geht, sagen die Leute: »Ich kann nicht wissen, wie das letzten Endes benutzt wird.« Meine Antwort dazu: Oh doch, das können Sie. Hier bin ich in einem fremden Land, aber von Amerika kann ich sagen, daß man dort sehr wohl wissen kann, wie die Arbeit eines KI-Labors oder des Labors, an dem ich arbeite, des Laboratory for Computer Science, letzten Endes angewandt wird. Im Fall MIT sehen wir ganz klar: Wenn es überhaupt möglich ist, ein Forschungsergebnis, zum Beispiel bei der Computer-Bildererkennung, irgendwie in eine Waffe einzubauen, um sie zu »verbessern«, dann wird das auch gemacht. Das läßt sich statistisch nachweisen, man bekommt aber auch schon einen Hinweis, wenn man einfach fragt: Wer bezahlt überhaupt für die Arbeit? In unserem Fall ist es meistens das Pentagon. Da sollte man schon erahnen können, daß diese Arbeit eventuell militärisch genutzt wird.

Natürlich kommen dann Gegenargumente, zum Beispiel: »Es ist nicht mein Job, herauszukriegen, was mit meiner Arbeit passiert.« Wernher von Braun, der ein schönes Buch geschrieben hat mit dem Titel »Ich strebe nach den Sternen«, den man ergänzen könnte

mit dem Untertitel »...aber manchmal treffe ich London«, der hat auf solche Fragen geantwortet: »Das ist nicht meine Abteilung« – Ende. Oder mein alter Kollege in der KI-Forschung an der Carnegie Mellon University, Herbert Simon, mit dem ich über diese Frage mal eine längere Debatte geführt habe, der sagte: »In Amerika haben wir eine repräsentative Regierungsform. Wir übergeben unseren gewählten Abgeordneten die Entscheidung, was mit unserer Wissenschaft zu machen ist. Und wenn uns das nicht gefällt, dann können wir andere wählen.« Ich finde, das ist eine Art Abdankung. Es erinnert mich an die Haltung der Mehrheit der deutschen Wissenschaftler, besonders während des Dritten Reichs: Wir sind Wissenschaftler, Politik geht uns nichts an, der Führer entscheidet.

Was machen wir mit der Einsicht, daß alles für gute und für böse Zwecke verwendet werden kann? Gestern hat jemand bei der Pressekonferenz gesagt: »Um Gottes Willen, man kann doch keine Schuhfabrik verbieten, nur weil die Schuhe eventuell an Soldaten gehen könnten.« Dann kommt die Frage: »Wie entscheidet man, was gut und böse ist?« Das ist natürlich eine uralte Frage, eine unmögliche Frage, die ich nicht beantworten kann. Aber ich kann mich ihr soweit annähern, daß es leichter wird zu bestimmen, wo wir die Grenze ziehen können, etwa indem wir sagen: »Bis zu diesem Punkt mache ich mit und nicht weiter.«

Also, was ist überhaupt »gut« und »böse«? Um darüber nachzudenken, muß man eine Art Weitwinkelobjektiv haben. Man muß einen ganz großen Kontext mitnehmen. Ein Beispiel: Wenn ich bei Vorträgen so böse spreche, wie man es von mir erwartet,



kommt anschließend das Argument: »Aber es muß doch möglich sein, Computer für humane Zwecke anzuwenden, zum Beispiel in der Medizin.« Wenn ich dann nach einem Beispiel frage, dann wird öfter der CAT-Scanner angeführt, Computer Aided Tomography. Kann man irgendetwas Schlechtes über den CAT-Scanner sagen? So viel ich weiß, ist noch nie jemand damit getötet oder verletzt worden, sondern er wird benutzt, um am lebenden Gehirn zu untersuchen, ob eine Operation notwendig ist, und wenn ja, dann wo genau der Tumor sitzt, so daß die Operation einfacher wird.

Jetzt aber im Weitwinkel: Wir müssen fragen, in welchem gesellschaftlichen Kontext der CAT-Scanner angewandt wird. In den USA hat er den Effekt, daß noch hunderttausende Menschen mehr als bisher nie einen Arzt sehen werden und keine medizinische Behandlung bekommen. Wie kommt das? Die amerikanischen Ärzte arbeiten innerhalb eines marktwirtschaftlichen Systems und sind immer der Gefahr ausgesetzt, wegen Kunstfehlern angeklagt zu werden. Dagegen müssen sie sich versichern. Das bedeutet, daß die Kosten einer medizinischen Behandlung steigen. Es geht aber noch weiter. Wenn ich zu einem Arzt gehe und sage, daß mein kleiner Finger an der rechten Hand weh tut, dann würde er sagen (ich übertreibe jetzt mal): Da brauchen wir einen kompletten »Check-up«, also zwei Tage im Krankenhaus, Röntgen-Bilder, CAT-Scan, Blutuntersuchung und alle möglichen Extras. Warum macht er das? Er will eben nicht angeklagt werden, weil er etwas vergessen hätte.

Das hat mit unserer Gesellschaft zu tun und nicht mit Medizin als solcher und schon gar nichts mit dem CAT-Scanner. Fast jedes Krankenhaus hat zwei oder drei CAT-Scanner, die immer weiter verbessert werden müssen. Die sind sehr teuer, und wir Amerikaner merken, daß das Gesundheitswesen unter diesem technischen Fortschritt leidet.

Man muß also den CAT-Scanner im Licht der gesellschaftlichen Umstände sehen, in die er eingebettet ist. Bei Computer-Bilderkennung ist völlig klar – quasi determiniert –, daß jedes Ergebnis, das irgendeinen Einfluß hat auf die Verbesserung der Fähigkeit des Computers zu sehen, sofort vom Militär aufgegriffen wird. Wir können nicht behaupten, wir wüßten das nicht. Wir wissen es ganz genau.

Zweckdenken

Angenommen, man möchte die Welt nicht weiter militarisieren, wo zieht man dann die Grenze? Ich glaube, wir müssen diese Grenze zum Teil willkürlich ziehen. Ein Beispiel: Wir könnten hier eine Umfrage machen: Jeder könnte auf einem kleinen Zettel aufschreiben, wann der Tag aufhört und wann die Nacht beginnt. Ich nehme an, daß wir viele verschiedene Antworten bekommen werden. Einer würde vielleicht sagen: »Wenn die Sonne untergeht.« Und wenn man man weiterfragt, »Wo denn?«: »Ja, wenn ich in meinem Appartement bin, und die Sonne geht

unter hinter dem Berg da hinten...« Oder ein Engländer würde vielleicht sagen: »Nach dem Tee, aber vor dem Cocktail.« Es gibt also bestimmte Anlässe, bei denen wir entscheiden: Hier hört der Tag auf, und da fängt die Nacht an. Wie wird das gemacht? Mehr oder weniger willkürlich. Aber eines ist klar: wir wissen, daß Mittag Tag ist, und wir wissen, daß Mitternacht Nacht ist. Das ist das Kriterium, das ich hier anwenden möchte.

Wenn wir also daran arbeiten, die Zielgenauigkeit einer Rakete zu verbessern, dann ist ganz klar: Das ist eine Nachtarbeit. Da muß ich nicht ausführen, wo ich die Linie ziehe. Es ist einfach »Nacht« – das ist überhaupt keine Frage. Und es gibt andere Sachen, die im gesellschaftlichen Kontext einfach »Tag« sind



und die man ruhig machen kann, ohne ein schlechtes Gewissen zu haben. Ich muß sagen, daß unsere Gesellschaft ein Irrenhaus ist, das kann man jeden Tag in der Zeitung lesen. Deswegen sind unsere Nächte viel länger als unsere Tage. Aber das ist ein anderes Thema.

Auf diese Art können wir eine Entscheidung treffen zwischen »gut« und »böse«; eine Entscheidung, der nicht jeder zustimmen würde, die aber jedenfalls für uns selbst gilt. Wenn wir das gemacht haben, und uns dann fragen: »Wie wird meine Arbeit schließlich genutzt? Fällt das in die Domäne 'Tag' oder in die Domäne 'Nacht'?« –, dann müssen wir uns auch fragen: »Wie weit ist dieser Endnutzen meiner Arbeit entfernt von der Position, an der ich mich jetzt damit befinde? Kann es sein, daß ich die Arbeit jetzt nicht machen darf, weil sie vielleicht in hundert Jahren für böse Zwecke benutzt wird? Oder fällt das Ziel in die Zeit meiner eigenen Tätigkeit?«

Gestern haben wir in der Pressekonferenz über die Verantwortung der KI-Forscher, besonders am MIT, geredet: Was ist das Ziel der KI, und können wir das verantworten? Und Marvin Minsky sagte, daß das Ziel der KI doch die Beseitigung des Todes sei. Ich nehme an, daß einige hier ein bißchen überrascht sein werden über diese Aussage. Aber in den USA ist das eine weit verbreitete Idee. Diese Diskussion ist in Amerika schon zwanzig Jahre alt; in Deutschland wird sie jetzt erst dokumentiert. Ich denke dabei an

das Buch, das Marvin Minsky hier schon erwähnt hat, »Mind Children« von Hans Moravec. Moravec behauptet, daß es ziemlich bald – leider noch nicht zu seinen Lebzeiten, bedauert er, aber zu der seiner Kinder – möglich sein wird, einen Menschen im Computer abzuspeichern. Auf Englisch: »to download a human being into a computer« – natürlich völlig digitalisiert, was bedeutet, daß dieser Mensch ewig leben kann. Wenn der Computer anfängt zu rosten, dann kann diese digitale Information natürlich auf einen anderen Computer übertragen werden – und dann haben wir die Unsterblichkeit. Ich will hier betonen: Hans Moravec ist Leiter des Robot-Labors in der Carnegie Mellon University. Jeder hier weiß, daß das eine sehr wichtige Technische Universität ist in den USA. Und das Buch »Mind Children« ist nicht im Axel Springer Verlag erschienen, sondern bei der Harvard University Press. Es wird also ernst genommen. Offensichtlich nimmt es Marvin auch ernst.

Es wird also gesagt, das Ziel sei die Beseitigung des Todes, oder es wird gefragt, wie könnten wir aufhören zu arbeiten. Wenn man diese Argumentation akzeptiert, dann ist man in der philosophischen Haltung, die behauptet, daß das Ziel alle Mittel rechtfertigt. Ich weiß nicht, wieviele Fässer (so würde man das heute wohl sagen – wir sprechen immer von »barrels of oil«), wieviele Fässer Blut in den letzten zweihundert Jahren vergossen wurde im Namen dieser Philosophie.

Eine interessante Frage, die ich hier aber nicht bearbeiten kann, wäre außerdem: Ist die Beseitigung des Todes – wäre sie möglich – ein absoluter Wert? Ich bin der Meinung, daß dies die Vernichtung unserer

Kultur bedeuten würde. Unsere Kultur ist ein lebendes Wesen, und sie wird dadurch am Leben erhalten, daß sie immer von einer Generation zur nächsten übergeben werden muß. Das ist kein Informations-transfer im Sinne von einer Kopie einer Bit-Reihe, sondern jedes Mitglied der nachkommenden Generation muß die Kultur aufs neue reproduzieren. Dabei stellt diese Generation nicht genau das wieder her, was ihr von der vorhergehenden Generation übertragen wurde, sondern eine Änderung. Das ist das Leben der menschlichen Kultur. Wollen wir das wirklich vernichten?

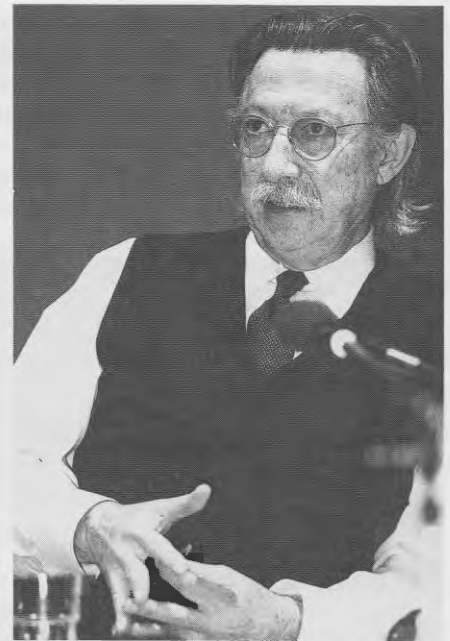
Moravec erwähnt in seinem Buch noch einen kleinen Nebeneffekt seines Forschungsprojekts, er nennt es eine »mind blowing idea«, eine Bombenidee: Er spricht von einem postbiologischen Zeitalter. Er formuliert ganz deutlich die Idee, daß die DNA schon bald funktionslos sein wird und daß die Biologie, das Leben, so wie wir es verstehen, einfach ausgeblasen wird. Wenn das also so ein Nebeneffekt dieser Forschung ist, dann sollten wir meiner Meinung nach nicht sagen, daß ihr Ziel ein absoluter Wert ist.

Wie weit darf also das Fernziel entfernt sein? Ich finde das Argument, das Marvin Minsky gestern hier vorgetragen hat, unzulässig. Er argumentiert: Das Fern-Fernziel, das ganz, ganz weit weg ist, ist gut – also brauchen wir uns nicht die Zwischenstufen anzusehen. Ich finde, das geht so nicht.

Verdrängung

In Amerika wissen wir ganz genau: Wenn wir auf dem Gebiet Bilderkennung/KI überhaupt Erfolg haben sollten, dann wird das zur »Verbesserung« einer





Waffe führen. Aber dieses Wissen ist ziemlich weit entfernt. Wir haben uns daran gewöhnt, eine riesige psychologische Distanz herzustellen zwischen unserem Tun und den Auswirkungen unseres Tuns. Das ist sehr weit verbreitet in unserer Gesellschaft, nicht erst jetzt im letzten Jahrzehnt dieses Jahrhunderts. Ich denke an Vietnam, an die Bombardierung durch ein B 52 Flugzeug, das in zehntausend Meter Höhe fliegt. Ein elektronisches Signal auf einem kleinen Bildschirm sagt dem Piloten, daß er das Ziel erreicht hat und jetzt die Bomben abwerfen muß. Er drückt auf einen Knopf. Die Bomben fallen. Er kann sie nicht sehen. Er ist weit weg, wenn sie unten ankommen. Er kann sie nicht hören. Ganz bestimmt kann er die Schreie der Menschen nicht hören. Und nur aufgrund dieser psychologischen Entfernung von den Auswirkungen seines Tuns wird es ihm möglich, diesen Knopf zu drücken.

Das war vor etwa zwanzig Jahren. Wir können das heute sehen bei den Videospiele, mit denen Kinder spielen. Manche Leute klagen, wie furchtbar das Fernsehen für Kinder sei, die ganz passiv davorsitzen, und behaupten dann, Videospiele seien die Lösung. Ein Kind schaut sich im Fernsehen einen Film an, nehmen wir mal an »Run silent, run deep«, der in den USA schon tausendmal im Fernsehen gezeigt wurde, ein U-Boot-Film, ganz typisch, man könnte fast sagen eine Parodie von einem U-Boot-Film. Das Kind sieht, wie der Kapitän durch das Periskop den Frachter beobachtet: Vielleicht ist es ein Kriegsschiff. Eine Zeitlang wird über Technisches gesprochen, Entfernung und so weiter, und dann heißt es: Torpedo los! Die Männer stehen da mit ihren Stoppuhren und warten gespannt auf den Einschlag. Dann sieht der Kapitän durch sein Fernrohr, wie das Schiff explodiert, und die ganze Crew klatscht. Es ist ihnen nochmal gelungen. Aber die Eltern machen sich Sorgen um das Kind, das vor dem Fernseher sitzt, so ganz passiv. Es macht einfach nichts. Und da soll das Videospiele eine Rettung sein: Mit Hilfe des Videospiele

kann das Kind jetzt selbst U-Boot-Kapitän sein, es kann den »Torpedo-los«-Knopf drücken und dann, wenn es geschickt ist, am Bildschirm sehen, wie das Boot explodiert und untergeht. Ich dagegen würde sagen: Das ist eine Art Training, um diese psychologische Entfernung zwischen Handlungen und ihren Wirkungen herzustellen. In den allermeisten Fällen denkt das Kind nicht daran, daß Menschen sterben, wenn dieses Schiff explodiert. Ganz besonders nicht beim Videospiele, denn da geht es darum, alles sehr schnell zu machen: Wenn man zehn Schiffe versenkt, bekommt man mehr Punkte als der kleine Bruder, der in derselben Zeit nur sieben Schiffe versenken kann. Das ist das Videospiele als Lehrer.

In einem kleinen Zeitungsartikel habe ich gelesen, daß in Deutschland, einem sehr fortgeschrittenen Land, Computerspiele verbreitet wurden, in denen das Kind KZ-Hauptmann spielen kann. (Zwischenruf aus dem Saal: Das Spiel wurde in der Bundesrepublik verboten.) Interessant ist, daß so etwas überhaupt existiert hat und dann erst verboten werden muß. Worauf ich hinaus will: Man macht etwas, und die Folgen des Handelns werden verdrängt.

Beschönigungen

In Forschungslaboren kann man ähnliches beobachten. Dazu eine kleine Geschichte: Ich ging in Cambridge über den Harvards Square, und ein junger Mann kam auf mich zu, der mal in einem meiner Seminare war. Er ist jetzt Doktorand, und er will mir erzählen, worüber er seine Dissertationsarbeit schreibt. Er sagt mir, ich sollte mir einen großen Bildschirm vorstellen, auf dem man einen Bär und eine kleine Katze sieht. Die Katze wirft einen Ball zu dem Bären, der fängt ihn und läßt ihn zur Katze zurückrollen. Vor diesem Bildschirm sitzt ein kleines Mädel und sagt zu dem System: »Bär, wenn dir jemand etwas gibt, solltest du 'Dankeschön' sagen.« Wenn die Katze danach den Ball nochmal zum Bären wirft,

dann sagt der Bär: »Dankeschön, liebe Katze«. Die Dissertationsarbeit besteht darin, dieses System zu implementieren. Die Profis hier werden wissen, daß das eine ungeheuer schwierige Arbeit ist, besonders wenn es so nahtlos gehen soll, wie er es mir beschrieben hat. Es hat mit Spracherkennung zu tun und solchen Sachen. Ich fragte den Doktoranden, ob er



denn hören möchte, wie ich das System verstehe, damit ich mich davon überzeugen kann, daß ich es richtig verstanden habe. Und ich erzählte ihm folgende Geschichte: Ich sehe einen Piloten im Kampfflugzeug, und das System sagt zu ihm: »Sir, da unten sehe ich eine Kolonne feindlicher Tanks.« Und der Pilot sagt zu dem System: »Wenn du so etwas siehst, dann setz' doch die Raketen in Gang und frage mich nicht immer.« Dann gehen die Raketen los. Punkt. Ende der Geschichte. Ich habe den jungen Mann dann gefragt, ob das eine Beschreibung seiner Arbeit sei, und er antwortete: Ja, man könne es so sehen. Und ich fragte ihn, wer für diese Forschungsarbeit bezahlt. Es stellte sich heraus, daß es das Militär war.

Ich glaube, daß meine Interpretation der Wahrheit näher ist als seine. Ich habe ähnliche Erfahrungen mit anderen Forschungsprojekten gemacht, die genauso in Form von Märchen, in wunderschönen Farben beschrieben werden. Nebenbei gesagt: Das Pentagon hat in der Zeit vor 1980, also vor der Reagan-Regierung, in seinen Forschungsaufträgen sehr selten ein Waffensystem tatsächlich beim Namen genannt. Da wurden alle möglichen Euphemismen benutzt.

Erst Anfang der achtziger Jahre – das Time Magazine sprach von einer Zeit des neuen Patriotismus – fing das Pentagon an, Waffensysteme tatsächlich beim Namen zu nennen. Im MIT wurde an drei Systemen gearbeitet, vor allem im Labor für Künstliche Intelligenz – das ganze Projekt hieß SCI (Strategic Computing Initiative). Und eines der Projekte war genau dieses Piloten-System, das ich vorhin beschrieben habe: »Pilot Assistance«, das den zweiten Mann im Kampfflugzeug ersetzen sollte.

Diktatur des Geldes

Ich glaube, wir haben einen Faustschen Vertrag unterschrieben mit unserer Wissenschaft und unserer Technik. Faust hat seine Seele nicht umsonst verkauft. Er wollte etwas dafür. Wir verkaufen unsere Seele nicht umsonst, wir bekommen jedesmal etwas dafür. Zum Beispiel High Definition Television, HDTV. Da bekommen wir ein besseres Bild. Es wird schöner sein. Dazu wird ein riesiger Forschungsaufwand betrieben. Der Zweck dieser Forschung hat sehr wenig mit Unterhaltung zu tun, aber wir sagen, naja, machen wir noch diesen Schritt, dieser eine Schritt wird ja niemanden weh tun. Wir fragen uns nicht: Wollen wir das? Wenn ich hier fragen würde, »wer braucht HDTV?«, würde wahrscheinlich niemand »ja« sagen. Aber in zehn Jahren wird fast jeder, der hier sitzt, ein HDTV-Gerät zu Hause haben. Da stimmt etwas nicht. Alle geben zu, daß wir es nicht brauchen, ganz besonders nicht, um den Blödsinn zu sehen, den man heute normalerweise im Fernsehen sieht. Wir haben ein künstlich hergestelltes Bedürfnis. Da stimmt etwas nicht, wenn wir Bedürfnisse künstlich herstellen, während so viele lebenswichtige Bedürfnisse so vieler Menschen nicht erfüllt werden können. Wir wollen also etwas, und währenddessen

entwickeln wir ein System, das uns entgleitet. Wir können es nicht mehr beherrschen, es ist aber unumkehrbar. Wir können dann nicht mehr sagen: Da haben wir einen Fehler gemacht, wir wollen das nicht. Viele dieser Systeme sind unbeherrschbar, undurchschaubar, unumkehrbar.

Kann man es anders machen? Ich bin der Mei-



nung, man sollte die Bedürfnisse der Zeit erkennen, eine Prioritätenliste machen und sich anhand dieser Liste entscheiden: Welche Fragen, die wir an die Natur stellen können, wollen wir jetzt tatsächlich stellen; welche Geräte, die technisch machbar sind, sollten wir tatsächlich produzieren?

Wir sollten nicht mehr mit der Lösung anfangen. Vor ungefähr fünfzehn Jahren war es in dem Labor für Computerwissenschaft, in dem ich gearbeitet habe, tatsächlich noch so, daß Ärzte zum Beispiel zu uns kamen und uns sagten: »Ihr habt da ein wunderschönes Instrument, das man sicherlich in der Medizin benutzen könnte. Wir erklären Ihnen die medizinische Seite, und Sie könntet uns sagen, wie man das Instrument anwenden könnte.« Man fing also bei der Lösung an. Auch die Rechtsanwälte. Ganz besonders die Schulen.

Bei den amerikanischen Schulen muß man bedenken, daß ein großer Teil der amerikanischen Jugendlichen analphabetisch ist, funktional analphabetisch. Nicht, daß sie keine Comic-Hefte lesen könnten, sondern sie können keine Stellenanzeigen lesen und schon gar nicht das Formular, das sie ausfüllen müssen, um den Job zu bekommen. In den Schulen stellt man sich dann Fragen wie »why Jonny can't read« und so weiter. Und es ist ganz klar: Die Lösung ist der Computer, denn Computer motivieren Kinder,

besonders Jungen. So wird argumentiert, statt daß man sich wirklich fragt: Warum kann Jonny nicht lesen lernen? Nicht einfach hoffen, daß der Computer helfen wird. Nein, Fragen stellen: Warum? Es könnte sich zum Beispiel herausstellen, daß Jonny nicht lesen lernen kann, weil er jeden Tag Hunger hat. Und dann fragt man weiter: Wir hatten doch solche Sachen wie School Breakfast und School Lunch. Warum kann er das nicht essen? Das ist weg. Warum ist das weg?

Es dauert nicht lange, bis man auf politische Fragen stößt. Deshalb wird wohl immer mit der Lösung angefangen – um eine Konfrontation mit politischen Fragen zu vermeiden. Oder kann es sein, daß Jonny nicht gut lernen kann, weil er Angst hat in der Schule? Warum sollte er Angst haben? Weil das größte Problem in den amerikanischen Schulen nicht mehr nur der Drogenhandel ist, sondern die Bewaffnung der Schüler mit Schußwaffen.

Wir sollten also Fragen stellen und nicht mit der Lösung anfangen, auch in bezug auf unsere Wissenschaft und unsere Technologie. Es ist ja nicht schwer, heute eine Liste von Bedürfnissen aufzustellen, es ist fast banal, diese Liste, die jeder kennt: Hunger, die Gefährdung der Atmosphäre, die ökologische Katastrophe, die vor uns liegt, Kriege und so weiter und so fort.

Jeder kann diese Liste aufstellen, um dann zu entscheiden: Wo fangen wir an, was ist wichtig, und was ist nicht wichtig. Wenn wir das machen würden, in dieser ganzen Infrastruktur des Wissenschaftsbetriebs, dann würde sich meiner Meinung nach herausstellen, daß verschiedene große Projekte, mit denen wir uns heute beschäftigen, einfach abgestellt werden sollten. Nicht weil die Fragen nicht interessant wären, sondern weil wir uns das heute nicht leisten können, und wir auch etwas unseren Kindern überlassen sollten. Ich denke dabei an Mars-Flüge. Das ist sicherlich sehr schön, aber warum so eilig? Warum müssen wir es jetzt machen, wenn es soviel Not gibt? Oder die riesigen Beschleuniger: Es ist ja schön, mehr von der Physik zu erforschen, aber wir können uns diese Technik jetzt nicht leisten. Ich habe nichts gegen einen Masarati, Ferrari oder Porsche, aber wir würden solche Maschinen keinen 13jährigen geben. Ich glaube, daß wir, unsere Gesellschaft, nicht reif genug sind, mit vielen Systemen, die wir herstellen können, umzugehen. Das sollten wir bedenken, bevor wir solche Projekte anfangen. Also plädiere ich für ein Moratorium für »Big Science«, für einen Aufschub von Projekten, die etwa eine Milliarde Dollar im Jahr kosten. So etwas sollten wir dann machen, wenn wir keine bessere Verwendung für dieses Geld haben.

Schlußfolgerung: Es gibt keine absolute Forschungsfreiheit, und es sollte sie auch nicht geben. Ich will auch nicht eine Diktatur von oben. Sie selbstverständlich auch nicht. Aber ich will auch keine heimliche Diktatur, eine gekaufte Diktatur, so wie es heute ist, in deren Licht Wissenschaftler und Forscher sich als Prostituierte des Geistes verhalten.

Balanceakt im Klangjungle

Tod Machover

»Wieviele Welten bewohnen wir gleichzeitig?« wird in Tod Machovers Oper *Valis* gefragt. Seine Kompositionen für Live-Musiker und Computer verbinden fragmentierte Erfahrungen zu einem inneren, psychologischen Netzwerk.

Ende der achtziger Jahre sah die Welt der Musik ziemlich kompliziert und unhomogen aus. Die Institutionen und kommerziellen Fernsehgesellschaften befanden sich mitten in einer größeren Umwälzung und regierten defensiv und übertrieben konservativ. Nie vorher hatten künstlerische Medien so viele Leute erreicht. Die Kommerzialisierung der Kunst führte aber auch dazu, daß künstlerische Erfahrungen in einem ganz neuen Ausmaß von Trivialisierung bedroht waren. Abgrenzungen zwischen den traditionellen Kategorien (Klassik, Jazz, Rock und so weiter) verschwommen – mit positiven wie auch negativen Konsequenzen. Die einzige einfache Formel, die den musikalischen Stil und die musikalische Sprache dieser Ära beschreiben könnte, wäre, daß es eben keine allgemeine Sprache oder Ausdrucksweise gibt. In der Musikgeschichte gab es wohl nie vorher eine Periode, in der so viele verschiedene musikalische Haltungen gleichzeitig gültig zu sein schienen.

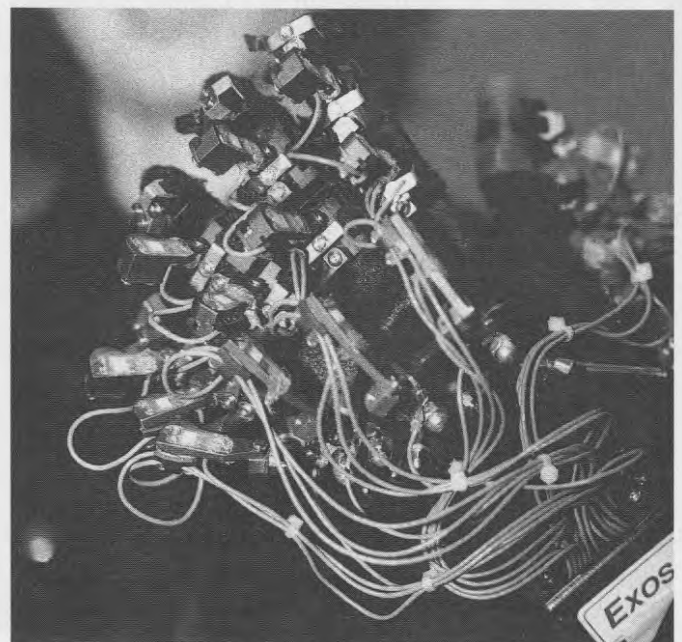
Was kann ein Komponist da machen? Wie manövriert man durch dieses unbekanntes Gewässer? Wie kann man ernsthafte Arbeit mit einer echten Beziehung zur Gesellschaft verbinden? Inwiefern ist die gegenwärtige kreative Freiheit günstig, und inwiefern ist sie eine beinahe unerträgliche Last?

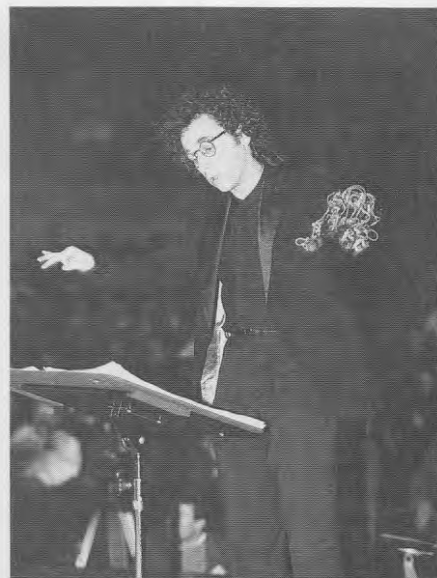
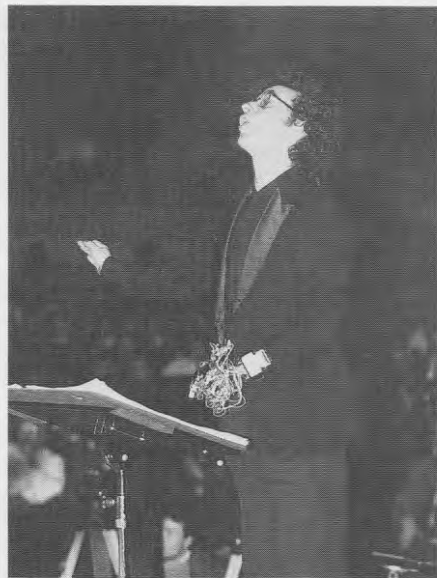
Eigentlich haben sich die wesentlichen Aufgaben eines Komponisten nicht verändert, und sie werden

sich auch nicht so schnell verändern. Die Einstellungen und Moden des Publikums schwanken und verlagern sich. Wir befinden uns mittendrin, eben weil es darum geht, etwas wesentliches über den Menschen mitzuteilen, indem wir diese einzigartige Verbindung von absoluter Präzision der Mittel und Widersprüchlichkeit in den Bedeutungen verwenden, die zur Musik gehört. Schließlich geht es nur darum, daß man lernt, musikalische Visionen zu nutzen, die so persönlich sind, daß sie tief empfunden werden, und gleichzeitig so objektiv, daß sie universell sein können. Jede Musik, die diesen Bedingungen genügen kann, wird ihre Zuhörer finden.

Was die überwältigende Vielfältigkeit in der künstlerischen Landschaft angeht: Ich habe beschlossen, mich ihr zu stellen, indem ich sie zu einer wesentlichen Komponente meiner musikalischen Sprache mache. Für mich war die Suche nach einer psychologischen und geistigen Einheit hinter dem scheinbar undurchdringlichen Schleier der Verwirrung und des Nebeneinanders das entscheidende Motiv seit den ersten Kompositionen. Ich bin davon überzeugt, daß eine solche Einheit besteht – bestehen muß. Die Gesellschaft präsentiert sich als ein nie zuvor dagewesenes Netzwerk aus komplexen, parallelen, offensichtlich vollständig unzusammenhängenden Welten. Das verstärkt bei mir nur das Gefühl, daß sich hinter der Suche nach tieferen Wahrheiten das Bestreben verbirgt, die zwischenmenschliche Kommunikation vor dem Zusammenbruch und dem vollständigen Verschwinden zu bewahren.

Ich will eine Musik komponieren, die ein Gefühl von Ausgleich verschiedenster Elemente vermittelt. Aber ich habe diese perfekte Einheit noch nicht gefunden. Meine Musik ist eine Suche, manchmal angestrengt und ungeduldig, manchmal ruhig und aufnehmend. Die Themen, die ich dabei verwende, und die Formen, die diesem Material Leben geben, habe ich ausgewählt, weil sie dieser zentralen Vorstellung zu entsprechen scheinen.





Ich will einen breiteren und tieferen Begriff von Musik voranbringen – einen Begriff, der die Musik wieder auf ein gleiches intellektuelles Niveau bringt mit den visuellen Künsten und auf eine ähnliche Bedeutungsebene. Es gibt viele Anzeichen dafür, daß unsere Gesellschaft jetzt offener ist für eine solche Neubewertung ernsthafter Musik. Die Rockmusik hat viele Leute auf eine eher fordernde Musik vorbereitet (und andere, zugegeben, entsensibilisiert). Die computergestützten Musikinstrumente bieten einen neuen gemeinsamen Nenner an für ernsthafte Komponisten, Forscher, kommerzielle Musiker und interessierte Amateure, die diese Techniken oft für sich selbst anschaffen. Viele Erkenntnistheoretiker wie Marvin Minsky und Howard Gardner haben gerade deshalb die grundlegende Wichtigkeit der Musik betont, weil sie mit menschlichen Absichten kommuniziert und gleichzeitig widersprüchlich und daher möglicherweise bedeutungsvoller ist als Sprache oder Bilder. Dieser Faktor könnte immer mehr Leute ansprechen, die nach tieferen, weniger künstlichen Erfahrungen suchen in dieser insgesamt materialistischen Gesellschaft.

Wir dürfen keine Angst davor haben, sehr hohe Ansprüche an diese neue musikalische Sprache zu stellen, wenn sie sich entfalten soll und überleben und gleichberechtigt integriert werden soll in die neuen, interdisziplinären künstlerischen Arbeiten der Zukunft. Es ist nicht ausreichend, wenn man einfach im stilistischen Kampf der Woche Stellung bezieht, es ist auch nicht gut genug, sich in technische Einzelheiten einer formalistischen Systematisierung zu verlieren. Wir müssen weitergehen und fragen, wie wir eine Musik schaffen können, die einen möglichst tiefen Eindruck bei denen hinterläßt, mit denen wir kommunizieren wollen.

Dabei dürfen wir nicht vergessen, daß die Kunst, die nur auf sich selbst bezogen ist, die nur von ihren eigenen Inhalten und Formen spricht, steril und künstlich bleibt. Der Künstler muß die Elemente der künstlerischen Sprache sorgfältig aufnehmen, um eine bestimmte Vision wiederzugeben. Genau diese

heikle Verbindung von tiefen, originalen Vorstellungen mit expressiven Formen, die wirklich sitzen, bestimmt den Wert eines Kunstwerks. Kunst kann und muß menschliche Realität wiedergeben, sie muß diese Realität aber objektivieren und erhöhen, indem sie ihr Form gibt. Über Musik kann man Realität wahrnehmen, indem man versucht, das Verhalten der musikalischen Mittel zu verstehen, man kann aber auch durch die Verwendung dieses Materials eine bestimmte Sicht der Realität ausdrücken.

Meine Musik drückt die Suche nach Einheit aus, gleichzeitig gibt es in meiner Arbeit ein Spannung schaffendes Paradox: Ich suche diese Einheit eher durch Einschließen als durch Ausschließen. Ich versuche, einen Ort zu finden für so viele verschiedene Klänge und Melodien und Texturen und Harmonien, daß jedes Stück kurz davor ist, wegen zu großer Kontraste und zu vieler Unterschiede zu explodieren. In diesem Sinne folge ich amerikanischen Komponisten wie Charles Ives und Elliott Carter. Ich bin erst dann zufrieden, wenn ich einen Platz gefunden habe für alles, was mich stimuliert, alles, was ich liebe. Aus diesem Chaos, aus diesem Klangjungle will ich eine Einheit schaffen.

Ich interessiere mich nicht so sehr für restriktive künstlerische Bewegungen wie die serielle, minimalistische, neoexpressionistische, technologische oder New-Wave-Bewegung oder »sound art«. Sie alle vereinfachen die Welt, indem sie sich eine sichere und begrenzte Ecke darin aussuchen. Mein Ideal ist Johann S. Bach, der für jede musikalische Idee, die in seiner Epoche entdeckt wurde, einen Ort gefunden hat. Er erfand eine Sprache, einen Zusammenhang, in dem all diese Elemente ihren natürlichen Platz fanden. Ich habe nicht das Glück, eine so umfassende und totale Vision zu besitzen. Aber meine Aufgabe ist es, so nah wie möglich an eine heranzukommen und mich so hart wie möglich darum zu bemühen, in diese Webarbeit jedes Element hineinzuknüpfen, das ich stimulierend finde, und gleichzeitig diese Reise in meinen Kompositionen zu dokumentieren.

Übersetzung: Eva Weber

Konzert mit sensibler Software

Tod Machover

Das Festival-Konzert: *Flora* (mit Computergraphik-Film); Suite aus der Oper *Valis* (nach dem Science-Fiction-Roman von Philip K. Dick), Sopran: Karol Benett; *Bug Mudra* mit David Starobin (akustische Gitarre), Oren Fader (elektrische Gitarre), Amy Knoles (Percussions) und Joseph Chung am Computer. Zwischen zwei Stücken erklärte Tod Machover das Hyperinstrument.

Das Hyperinstrument basiert auf einem Konzept, wie man ein Mensch-Maschinen-Interface gestalten kann. Darüber hat Marvin Minsky in seinem Vortrag schon etwas erzählt. Wir kombinieren hier eine hochvirtuose musikalische Aufführung mit ziemlich intelligenten Computern. Die Computer können die Ausdruckskraft der Musiker erweitern und immer wieder anders variieren, aber die Musiker kontrollieren immer die Aufführung.

Das geht nur mit einem sensiblen Computer. Bei der heutigen Performance sind drei Computer beteiligt. Im ersten Stadium sind die Instrumente über Kabel mit dem Macintosh IIci

verbunden. Auf diesem Computer läuft Lisp-Software, eine spezielle Programmiersprache namens Hyper-Lisp, die wir am Media Lab entwickelt haben. Lisp ist eine Sprache, die sich sehr gut für das Verständnis und die Generierung von musikalischen Systemen eignet. Es ist nicht die schnellste Computersprache. Deshalb bestand die große Programmieraufgabe für Jo Chung darin, wie er Lisp extrem effizient machen kann, möglichst intelligent und doch sehr schnell. Und es war natürlich wichtig, daß Lisp nicht mit garbage-collecting anfängt oder abstürzt, während wir spielen. Aber das wird uns wahrscheinlich nicht passieren, denn Jo und das Software-Team haben mit großem Talent eine Software entwickelt, die normalerweise wirklich zuverlässig ist.

Auf der ersten Stufe geht die Musik also in den ersten Mac, der sich verschiedene Aspekte der Aufführung anschaut: Welche Noten werden gespielt, wann werden sie gespielt und wie werden sie gespielt? Lautstärke, Anschlag, Phrasierung und Artikulation. Eins der netten Sachen bei Computermusik mit Software ist, daß man dem Computer in verschiedenen Abschnitten der Musik verschiedene Verfahren vorgeben kann. Er kann die Performance an verschiedenen Stellen der Partitur auf jeweils verschiedene Arten interpretieren. Mal kann er die Musik rhythmisch sehr präzise machen – er kann zum Beispiel

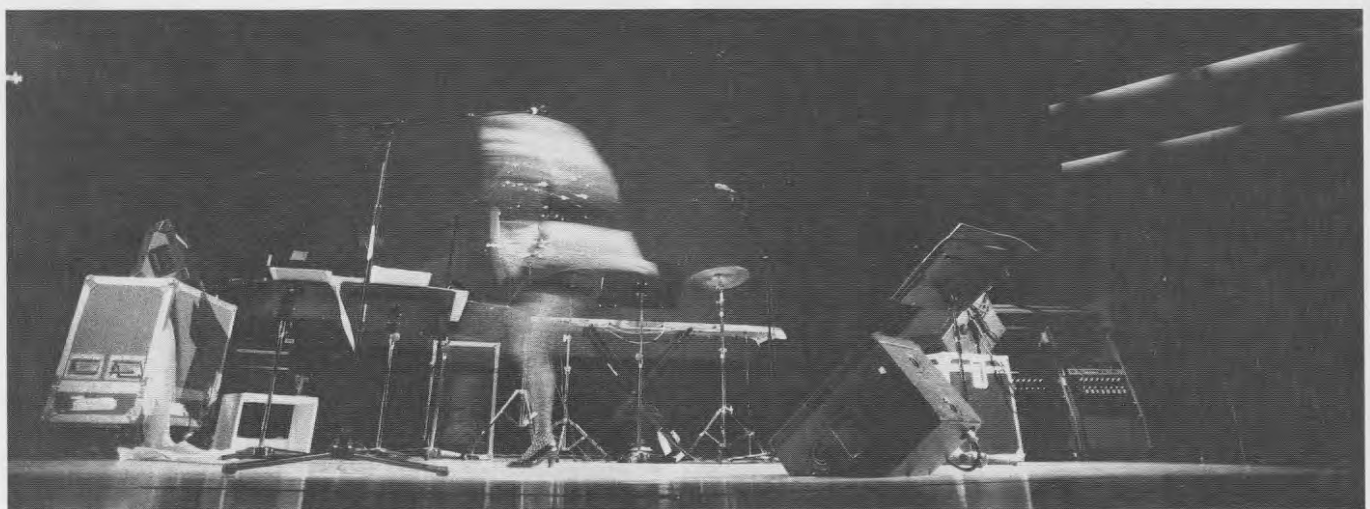
sehr schnelle Noten aufnehmen und alle Musiker synchronisieren, so daß sie exakt zusammenspielen –, mal macht er genau das Gegenteil und fügt den gespielten Rhythmen andere hinzu, so daß es sehr komplex wird. Er macht das gleiche mit



Melodien, mit Harmonie und mit der Klangfarbe. Er nimmt also die Performance auf und gibt ihr eine zusätzliche Dimension.

Der Computer dort hinten schaut auf diesen Handschuh. Wir wollten Gesten, Bewegung im Raum in Anweisungen über das musikalische System übersetzen.

Dieser Handschuh wurde von Studenten des Massachusetts Institute of Technology entworfen. Er ist erst ein Jahr alt und wurde zunächst für medizinische Zwecke und zur Roboterkontrolle entwickelt. Wir haben mit der Gruppe zusammengearbeitet und den Handschuh für unsere Zwecke umgebaut. Dieser hier ist der einzige Datenhandschuh, der präzise und empfindlich und robust genug ist für eine anspruchsvolle Musikperformance.



Er hat ein Aluminium-Skelett, und an jedem Fingergelenk gibt es ein Scharnier mit zwei Teilen, einem Magnet und einem Sensor, die sich gegeneinander bewegen. Der Sensor mißt die Position, die Polarität des Magnets, die die Krümmung und die Position der Finger anzeigt. Er kann zwanzig Parameter für die Handbewegung weitergeben mit einer Frequenz von etwa 100 Hertz. Bei Bug-Mudra nehmen wir die Bewegungsinformation aber in einer geringeren Auflösung.

Diese ganze Information wird an den IBM PC geschickt, der die Aufgabe hat, meine Handbewegungen zu messen. Der Computer in der Mitte, auf dem auch Hyper-Lisp-Software läuft, hat die Aufgabe, die Verbindung zwischen musikalischer Perfor-



mance und der Hand herzustellen. Dahinter steht folgendes Konzept: Ein Dirigent arbeitet mit beiden Händen. Dabei gibt die rechte Hand den Takt an, um schneller oder langsamer zu werden, und sie synchronisiert die Musiker. Aber das war für mich keine richtige musikalische Herausforderung, denn darüber gibt es schon eine Menge Forschungsarbeiten, und man versteht es inzwischen schon zu gut. Wir wollten ein schwierigeres Problem lösen. Wir wollten die linke Hand nehmen, die beim Dirigieren dazu verwendet wird, das Gefühl der Musik zu vermitteln – manchmal ein einfaches Gefühl wie »leiser Klang für das Cello oder die Violinen« oder »mehr Trompete«, und



manchmal kann ich ein Signal geben für einen starken oder zarten Akzent oder für brillantes Spiel. Beim Dirigieren wird mit Gesten die Klangqualität angegeben. Genau das versuchen wir mit unserer Software zu machen.

Die Instrumente bringen viele Musikebenen hervor. Diese Musik geht durch den ersten Computer und durch viele Synthesizer und dann in die linke Hand des Dirigenten. Meine linke Hand kontrolliert dann den allgemeinen Klang in jedem Musikabschnitt auf eine jeweils besondere Art. Am Anfang und

am Ende des Stückes setze ich die linke Hand meistens für Betonungen ein, um bestimmten Akzenten eine besondere Kraft zu geben; und in der Mitte des Stückes werde ich irgendwann ganz aufhören, mit der Rechten zu dirigieren, und nur die linke Hand benutzen. In diesem Abschnitt setze ich die linke Hand so ein, als würde ich mit ihr malen. Je nach dem, wohin ich mich bewege und wie ich mich bewege, wird aus einem normalen Klang ein komplexerer Klang. Das ist für mich wie Malen mit Klangfarben.

Übersetzung: Eva Weber



»Die Dinge beim Namen nennen«

Niklaus Wirth

Konzeptionelle Klarheit ist das Prinzip, nach dem Niklaus Wirth, der »Vater des strukturierten Programmierens«, auch sein jüngstes Kind, das Oberon System, gestaltet hat. Ein Interview mit dem bedeutendsten Designer einer exakten Informatik.

Wie bewerten Sie die Diskussionen über Computer und Kultur während des Borland European Software Festivals?

Ich bin sehr dafür, daß bei Gelegenheit auch die Implikationen der Technik diskutiert werden. Sicher hat das Computerwesen Einfluß auf die Wirtschaft und damit auch auf die Gesellschaft. Aber ich bin etwas skeptisch gegenüber den Leuten, die sehr stark von ihren Zukunftsvisionen eingenommen sind und uns das Blaue vom Himmel herunter versprechen. Was die Visionen des Herrn Minsky angehen, ist der einzige Rat, den man meiner Ansicht nach geben kann und den ich auch persönlich befolge: den Mann nicht so ernst nehmen – dann ist es lustig.

Fragen wir uns lieber, welche Bereiche der Computer denn wirklich beeinflußt hat. Abgesehen vom rein militärischen Bereich ist der Einfluß in der Forschung festzustellen: Man hat heute ganz andere Mittel, Naturphänomene zu erforschen und technische Lösungen zu simulieren, statt sie zuerst zu bauen. Wenn Sie Konstruktionen, für die es keine geschlossenen mathematischen Lösungen gibt, in Simulationen ausprobieren können, dann bringt das natürlich riesige Vorteile.

Weitere wichtige Anwendungsbereiche des Computers sind das Verwaltungs- und Bankenwesen, vielleicht auch noch bei Verkehrsmitteln. Aber außerhalb dieser Sektoren ist der Computer noch nicht so weit vorgedrungen, daß er unersetzlich wäre. Ich habe manchmal sogar das Gefühl, daß Computer eingesetzt werden, weil sie als modern gelten. Auch bei Tätigkeiten, in denen die älteren Methoden eigentlich ausreichen. Dadurch schafft man viele ganz unnötige Abhängigkeiten. Besonders wenn das, was hinter dem Bildschirm steckt, überhaupt nicht mehr beherrscht wird. Gut, das ist überall so, auch im Haushalt: Wenn der Kühlschrank versagt, müssen wir einen Monteur rufen. Aber es geht dann um ein viel einfacheres Gerät, und es gibt viele Kühlschrankmonteure, also haben wir überhaupt keine Angst, wir könnten ohne Kühlschrank in der Hitze stehen bleiben.

Wenn aber etwas beim Computer nicht ganz funktioniert, und sei es nur Software, dann stehen 99 Prozent der Computerbenutzer da wie der Ochs vorm Berg. Diese Abhängigkeit ist wahrscheinlich nicht zu umgehen, aber man sollte sich klarmachen, ob man den Computer so sehr braucht, daß man diese Abhängigkeit eingehen will.

Ist diese Abhängigkeit mit einem bestimmten Programmstil verbunden, der das Mitdenken der Anwender überflüssig macht?

Jedes System besteht aus vielen Schichten. Ein einfaches Beispiel: Sekretärinnen mit Textverarbeitung am Arbeitsplatz müssen die Grundsätze der Textverarbeitung verstehen. Sie müssen wissen, wie der Text in dieser Anwendung strukturiert ist. Der Anwender muß wissen, was Text ist. Aber er sollte nicht wissen müssen, wie der Text im Computer intern verwaltet wird oder wie das Betriebssystem organisiert ist. In diesem Sinn sind unsere Systeme noch nicht gut genug organisiert. Wenn etwas nicht klappt, und leider passiert das oft, dann steht der Anwender hilflos da, weil er nicht unter die Schnittstelle blicken kann – was er ja auch nicht tun sollte. Dann wird ein Fachmann gerufen, und dem fehlt das Wissen meistens auch, weil das Programm vielleicht in den USA produziert wurde. So etwas passiert beim Kühlschrank nicht: Die Servicetechniker sind an den Maschinen ausgebildet worden, um die jeweiligen Probleme zu beheben.

In den Fachblättern entfaltet sich die Kontroverse zwischen Anhängern strukturierter Programmiersprachen (wie Pascal) und objektorientierten (wie C++). Welche Rolle spielt die Typenüberprüfung?

Ich würde es nicht als Kontroverse bezeichnen. Das wesentliche an Pascal war die Einführung des Konzepts der Datentypen. Jedes Objekt, sei es Variable, Parameter, Funktion oder Konstante, ist einem Typ zugeordnet, der aus dem Programmtext statisch ersichtlich ist. So kann der Programmierer überprüfen, ob er die Typen sinnvoll verbindet, und der Compiler kann es noch einmal überprüfen. In C fehlt diese Fähigkeit, ebenso beim Assemblercode. Hier muß sich der Programmierer entscheiden: Will ich die Flexibilität von Assemblercode oder C, oder will ich ein System, das Typen kennt? Es gibt keine Kontroverse, sondern zwei verschiedene Klassen: eine höhere und eine tiefere Klasse.

Das Typenkonzept ist etwas ganz wesentliches. C++ ist eine Erweiterung von C um objektorientierte Strukturen, und da hat man auch das Typenkonzept einigermaßen eingeführt. Aber es gibt nach meiner Erfahrung und Überzeugung hier kein Zwischenmaß, sondern entweder hat man Typen oder nicht. In C++ ist das nicht sauber gelöst. Wichtig ist, daß der Compiler wirklich immer nachprüfen kann, ob die Typenkompatibilität gewährleistet ist. Denn wir sehen, daß die Fehler dort sind, wo man sie nicht vermutet. Oberon ist fast völlig in der Sprache Oberon programmiert, in der es vollständige Typen gibt. Es gibt nur ein paar wenige Dinge in Assemblersprache. Dort habe ich keine Typen, also weiß ich: Wenn etwas passiert, habe ich einen Fehler gemacht – der schwer zu finden ist.

Noch ein Wort zur Objektorientierung: Bei Oberon vermeiden wir, neue Nomenklaturen einzuführen wie Objekt, Klasse, Meldung und Methode. Wir zeigen, daß es das schon in der Welt der prozeduralen Programmierung gibt. Wir zeigen, daß man nicht in

eine neue Welt eintritt und das, was man gelernt hat, beiseite schieben muß. Man lernt neue Programmier-techniken – aber auf dem Boden des bisher Mitgebrachten.

Hinzugekommen sind bei der objektorientierten Programmierung Klassen und Subklassen. Klassen sind bei uns Typen. Was bei Oberon, im Vergleich zu Pascal und Modula dazukam, ist die Möglichkeit, die Typen zu erweitern. Der erweiterte Typ ist die Subklasse. Je mehr Attribute ich beifüge, desto mehr Einschränkungen muß ich in Kauf nehmen. Zum Beispiel: Wenn ich von Tieren rede, dem Typ »Tier«, dann ist es sehr allgemein. Jetzt gibt es noch die Unterklasse der Säugetiere, dann kommen einige spezifische Attribute hinzu, die auf diese Unterklasse beschränkt sind. So etwas haben wir auch. Das heißt in der objektorientierten Welt die Unterklasse – etwas, was auf der bisherigen Basis auch möglich ist. »Typen« sind ein sehr alter Begriff in der Mathematik. »Klassen« sind dagegen nicht so sauber definiert.

Haben Sie deshalb diesen Mechanismus »Typenerweiterung« genannt und nicht, wie in der objektorientierten Sprache, »Vererbung«?

Ja, ich bin immer skeptisch gegen solche anthropomorphe Ausdrücke wie »Vererbung«. Es ist niemand gestorben.

Glauben Sie, daß es unter den Programmierern einen Generationswechsel gibt – eine neue Generation, die von ihrer Ausbildung her andere Begriffe im Kopf haben?

Sicher werden neue Ideen immer von den jungen Leuten aufgenommen und in die Praxis getragen. Das war schon bei Pascal so. Die alten Füchse können Sie nicht mehr zum Umdenken bringen. Die haben zehn Jahre investiert, um etwas zu beherrschen – und jetzt kommt jemand und will mir diese Maschine, die ich in allen Details kenne, wegnehmen. Man müßte aber die Leute nicht unbedingt mit neuen Schlagwörtern gewinnen, sondern könnte auf dem Boden der Sachlichkeit bleiben. Gerade in der akademischen Ausbildung sollte man die Dinge beim Namen nennen und auf dem aufbauen, was wir schon kennen. Die Mathematik hat keinen kommerziellen Hintergrund. Aber das Computerwesen wird sehr stark von kommerziellen Interessen dominiert, und neue Worte finden eben mehr Gehör.

Sie haben in Oberon Abschied vom Programm und dem Programmablauf genommen. An ihre Stelle treten Module und Prozeduren.

Das ist unabhängig von der Programmiersprache, hier geht es um das Betriebssystem. Es gibt da eine klare Trennung, zum Beispiel bei Modula oder auch bei Ada. Da ist ein Modul einmal eine textuelle Einheit, die der Compiler auf einmal kompilieren kann. Auf der anderen Seite ist ein Modul gleichzeitig die Einheit der Aktivität. Sie rufen damit ein Programm auf. Bei Oberon trennen wir klar zwischen den Begriffen: Ein Modul ist eine textuelle Einheit. Nebenbei kann der Compiler sie noch separat kompilieren. Aber die Aktivität, die Sie auf dem Computer starten möchten, muß nicht unbedingt die gleiche sein, die mit der textuellen Einheit beschrieben ist. In dieser



textuellen Einheit Modul gibt es soundso viele Variablen und soundso viele Prozeduren, die exportiert werden. Die können Sie als einzelne Stücke aktivieren. Wenn diese manchmal sehr kleine Aktivität in ein paar Millisekunden beendet ist, dann ist die Kontrolle wieder da. Wenn ich dagegen ein Programm starte, dann geht der Computer in einen Modus. Die beste Illustration dazu: Wenn Sie in mit einem herkömmlichen MS-DOS-Computer einen Text editieren, dann sind Sie zuerst im Betriebssystem, dann rufen Sie das Programm EDIT auf. Und EDIT bringt Sie in eine andere, eigene Umgebung mit anderen Regeln. Jetzt ist Text nicht mehr ein Befehl, sondern etwas, das eingetippt wird. Das machen wir bei Oberon eben nicht. Wir gehen nicht von Modus zu Modus, sondern es gibt immer kurze Aktionen, die eine sehr große Bandbreite haben. Eine Aktion kann das Eintippen eines Zeichens oder das Auswählen eines Zeichens durch einen Mausklick sein, etwas, das sofort beendet ist, es kann aber auch eine Simulation sein, die 24 Stunden dauert.

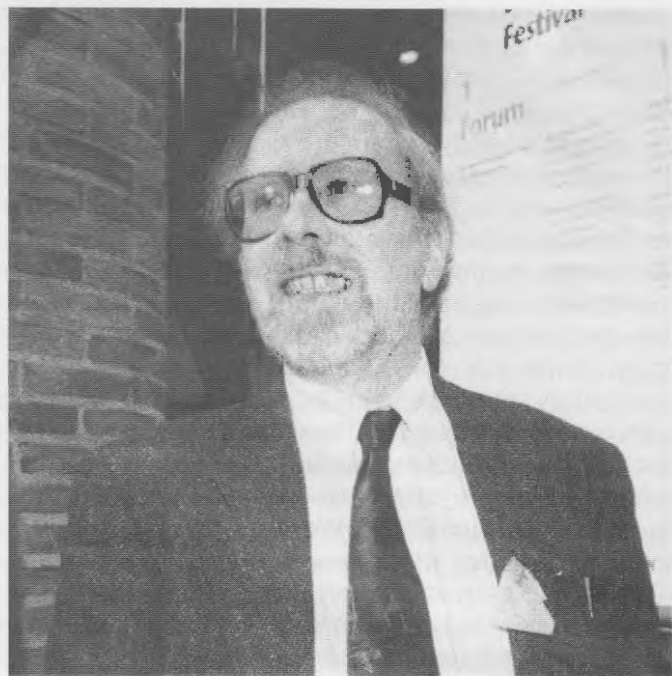
In MS-DOS zum Beispiel bewege ich mich in verschiedenen Modi. Oberon hat dagegen eine völlig konsistente Oberfläche. Ist das eine Konzeptänderung?

Ich würde eher sagen, es ist eine Änderung in der Organisation. Daraus resultiert ein anderes Bewußtsein bei der Bedienung. Das ist ein wesentlicher Beitrag zur Vereinfachung. Es ist für Nichtfachleute sehr verwirrend, wie man von Modus zu Modus kommt. Der Anwender muß immer im Kopf behalten, wo er ist und was in diesem Modus geht. All das entfällt. Sie haben in einem gewissen Sinne Zustände, aber die sind alle sichtbar, nicht versteckt. Das macht die Bedienung einfacher.

Eine vergleichbare Art der Benutzerführung ist schon seit Jahren auf Desktop-Betriebssystemen implementiert, zum Beispiel beim Macintosh. Warum jetzt mit Oberon eine weitere Oberfläche?

Wir haben ja auch Fenster, die aber nicht überlappen, was eher ein Detail ist. Der Grundanstoß zu Oberon kam von Xerox' Cedar. Wir verwendeten hier Tiling, ohne Overlapping. Zum Beispiel Pop-Up-Menüs: Wir sind viel flexibler, weil Sie in jedem beliebigen Text Kommandos angeben können. Sie können Text als Output generieren und dann die Kommandos anklicken. Ich kann aus meinem Briefkasten eine Meldung von einem Kollegen lesen, der mir schreibt, er habe jetzt irgendwelche neue Versionen zur Verfügung gestellt. Er kann dieser Meldung genau diese Kommandos anhängen, dann brauche ich sie nur noch anzuklicken.

Aber die Pop-Up-Menüs sind auch möglich. Wir haben sie in einem der Packages verwendet, und sie sind als Tool zugänglich. Aber die Kommandos in den Texten sind flexibler. Was wir nicht benutzen, sind Icons, diese Bildchen: Wir finden, daß die Schrift viel ausdrucksfähiger ist und prägnanter, exakter. Wenn ich nur irgendein Kreuzchen und einen Kopf sehe, weiß ich meistens nicht genau, was es ist. Im Text dagegen ist das ganz klar ausgedrückt. Noch ein Nebeneffekt: Man hat früher und auch heute



noch in der Computerwelt gelitten unter der Abkürzungswut, unter all diesen Akronymen und Drei-Buchstaben-Wörtern. Davon sind wir weitgehend weggekommen. Wir schreiben das volle Wort aus, auf Deutsch oder Englisch. Wir tippen die Worte ja gar nicht mehr, sondern kopieren sie schnell. Denn irgendwo im Dokument sind sie ja vorhanden. Das Tippen spielt eigentlich eine untergeordnete Rolle, also kann man die Wörter so schreiben, daß man sie immer verstehen kann. Das ist Fortschritt im User-Interface, statt einer Verkindlichung der Welt.

Sie meinen, Icons führen zur Verkindlichung?

Meiner Ansicht nach schon. Ich würde Bilder nicht grundlegend abwerten, aber wir haben uns nie bemüht, Computersysteme für den Kindergarten zu bauen, sondern für Leute, die sie lesen und verstehen können. Icons scheinen mir nicht der richtige Weg zu sein. Diese Verzierungen kosten ja einiges. Sie können heute kaum mehr einen Computer kaufen mit weniger als 1 Megabyte Speicher. Und wenn Sie einen bekommen, dann können sie damit nichts mehr anfangen, weil die Software mehrere Megabyte benötigt. Ich habe aus einem Forschungslabor in Kalifornien einen Bericht erhalten, daß mit einer neuen Release ihres Betriebssystems alle Maschinen eine Speichererweiterungen bekommen haben. Dabei hat schon jede Maschine 64 Megabyte Hauptspeicher. Hier haben wir dieses System gemacht mit der Flexibilität von kommerziellen Systemen, und uns genügen ein paar Hundert Kilobyte.

Brauchen denn jene Systeme wirklich die Megabytes für sinnvolle Anwendungen? Oder geht der Speicher für Bildchen und Verzierungen verloren? Für diesen Schmuck sind die Kunden merkwürdigerweise bereit zu zahlen. Wahrscheinlich weil sie nicht wissen, daß es anders geht.

Glauben Sie, daß das nur eine Modeerscheinung ist?

Ich bin vorsichtig mit meinen Prophezeiungen. Es gibt doch sehr viele Leute, die diese Dinge gern ha-

ben. Es sieht attraktiv aus, wenn Sie Windows haben, Overlapping, verschiedene Farben und im Hintergrund ein schönes Girl und so weiter. Wenn man aber wirklich an den Sachen interessiert ist, die Computer können und nicht am Spielen, dann braucht man das nicht.

Gerade bei rechenintensiven Aufgaben wäre es wünschenswert, einen Prozeß im Hintergrund ablaufen zu lassen. Aber Sie haben sich bei Oberon für ein Einprozeß-Multitasking-Design entschieden. Warum?

Wir haben uns damit einen Haufen Probleme vom Hals gehalten. Jürg Gutknecht und ich haben dieses System konzipiert und selber programmiert. Es ist schon verrückt, daß ein Zweimannteam so etwas nebenbei gemacht hat. Wir hatten auch noch andere Dinge zu erledigen. Wenn wir nun noch Multitasking eingebaut hätten, wären wir wahrscheinlich heute noch nicht fertig. Das ist eine gewaltige Verkomplizierung. Mir scheint der Trend eher dahin zu gehen, daß man größere Backgroundjobs an einen Server delegiert. Ich gebe gerne zu, daß das natürlich sehr verschwenderisch ist, weil die Maschine ständig läuft. Auf der anderen Seite haben wir heute schon einen riesigen Überfluß an Computerzyklen. Ich glaube, daß die Entwicklung dahin geht, für jeden Prozeß einen Prozessor einzusetzen. Prozessoren sind ja heute sehr billig geworden.

Der Server steht in einem Kämmerlein und muß einen Task nach dem anderen abwickeln. Wenn Sie Realtime-Systeme haben, ist das natürlich nicht tragbar. Aber andererseits sind viele andere Dinge, die in Oberon geboten werden, auch nicht unbedingt notwendig. Mit dem gleichen Betriebssystem allen gleich optimal dienen zu wollen, ist der falsche Weg.

Für jede Anwendung ein anderes Betriebssystem?

Wobei der Begriff Betriebssystem auch etwas ins Wanken geraten ist. Was ist denn in Oberon das Betriebssystem? Eine Hierarchie von Modulen. Unsere Philosophie lautet: möglichst wenig solcher Module als Grundsystem. Jeder Benutzer baut sich seine eigene Hierarchie auf, die er gerade benötigt. Wenn Sie zum Beispiel den Texteditor aktivieren, dann wird automatisch die Hierarchie von Modulen aufgebaut, die dort vorhanden ist. Der Rest ist nicht im Speicher. Die riesigen Speicher sind ja notwendig, weil in vielen Fällen 95 Prozent des Codes gar nicht gebraucht werden, aber automatisch mit eingebunden ist. Das ist bei uns nicht der Fall.

Warum haben Sie in Oberon die Garbage Collection wieder eingeführt?

Viele Situationen werden dadurch für den Programmierer wesentlich einfacher. Oft ist es dem Programmierer gar nicht bekannt, wann ein belegter Speicher wieder frei wird. Wenn ein Editor auf einen File zugreift, angeklickt durch eine Mausektion, dann wird dieser File geöffnet. Aher wann er wieder freigegeben werden kann, ist nicht bekannt. Das kann auch nicht durch den Programmierer gesteuert werden. Denn durch andere Tasks und Fenster kann auf diesen gleichen File auch noch irgendwo Bezug genommen werden. Also wird eine zentrale Verwaltung

benötigt, die bestimmen kann, ob es noch irgendwo eine Referenz darauf gibt oder nicht. Das ist der Garbage Collector. Die globale Ressource Speicher muß global verwaltet werden. Genau wie die globale Ressource Plattenspeicher von einem Allocator verwaltet wird.

Ist die Garbage Collection eine Folge des objektorientierten Designs von Oberon?

Ja, wenn Sie objektorientiert nicht ganz spezifisch im Sinne dieser typgebundenen Methoden verstehen. Es wäre im übrigen auch sonst nützlich, diese Garbage Collection zu haben. Alle Objekte haben Typen, und der Garbage Collector muß in jedem Fall Typeninformation für eine Interaktion haben. Er muß wissen, was vorhanden ist, ob es weitere Verzweigungen gibt und so weiter. Das ist nicht ohne zuverlässiges Typensystem machbar. So wird die Speicherverwaltung sehr absturz- und zugriffssicher.

Wie steht es um die Sicherheit zwischen den einzelnen Objekten? Sind die Objekte voneinander abgeschirmt, oder ist eine Interaktion zwischen den Objekten möglich?

Die ist nicht möglich – wenn Sie in der Oberon-Sprache bleiben. Aber Sie können auch bestimmte Module in Assembler kodieren. Aber das empfehle ich niemanden. Auf der untersten Schicht sind die Objekte aus Effizienzgründen in Assembler programmiert. Davon abgesehen, ist die Interaktion auf die in der Sprache definierten Interfaces beschränkt.

Oberon ist bereits die vierte Sprache, an der Sie maßgeblich bei der Entwicklung beteiligt sind. Warum suchen Sie nach der perfekten Sprache?

Die Sprache ist eine formale Notation. Das Wort »Sprache« ist eigentlich ein Fehlbegriff. Wir sprechen nicht in Oberon, wir sprechen nicht in Pascal, sondern wir drücken Programme formal aus. Warum mache ich das immer wieder? Eigentlich, weil ich als Ingenieur immer wieder interessiert war, Computersysteme zu kreieren, Betriebssysteme, Compiler und Graphiksysteme. Und als Lehrer möchte ich nicht nur Systeme konstruieren, sondern sie auch präsentieren und erklären können. Dazu müssen sie verständlich dokumentiert sein. Verständlich auf einer abstrakten Art, die sich auf Regeln stützt, die klar formuliert sind. Damit wird die Erzeugung eines geeigneten Formalismus eine dringende Notwendigkeit. Den gab es nicht oder jedenfalls für mich nicht in befriedigender Art und Weise. Ich habe 1960 angefangen. Fortran war sicher nicht befriedigend, Assemblercode noch viel weniger. In den dreißig Jahren danach haben sich unsere Projekte sehr verkompliziert und diversifiziert. Die Sprachen, die ich definiert habe, sind ja nicht isoliert, sondern bilden eine evolutionäre Reihe. Ich habe angefangen mit Algol W, dann folgten Pascal und Modula und jetzt Oberon. Bei Pascal war der wesentliche Beitrag die Datenstrukturierung, in Modula die Modularität, und vielleicht auch das Information Hiding; und in Oberon ist es die Typenerweiterung.

Was kommt nach Oberon?

Ich habe nicht im Sinn, in diese Richtung weiter zu arbeiten. Man muß auch einmal einen Strich zie-

hen können. Das heißt nicht, daß ich mich total von diesem Gebiet abwenden wollte. Die parallelen Systeme, die verteilten Systeme – das sind Herausforderungen. In diesem Bereich sind noch keine idealen Formalismen gefunden worden. Es wird keine totale Revolution geben, aber wichtige Zusätze. Da muß man zuerst die wirklich grundlegenden Formalismen herausfinden. Da gibt es mehrere Streitfragen.

Welche?

Natürlich gibt es schon viele verteilte Systeme, aber sie werden zusammengebastelt. Sie können dort nicht in Assemblercode oder in C schreiben, wie es die meisten tun. Es ist keine Situation, wie man sie als Wissenschaftler in einem Buch beschreiben möchte, sondern jeder muß diese kryptographischen Dinge entziffern. Ich möchte die Konzepte möglichst klar zeigen und dann zeigen, wo Zusätze benötigt werden und so weiter. Nicht umgekehrt. Wenn Sie sich diese Kryptogramme in C einmal anschauen, dann sehen Sie, daß es wirklich ein Höllenaufwand ist, das Wesentliche herauszufinden. Außer Sie haben gerade den Autor neben sich.

Was halten Sie von Smalltalk?

Ich habe Smalltalk 1976/77 bei Xerox kennengelernt, und es war für mich einfach keine saubere Sprache. Sie haben da kein sauberes Konzept, das Sie mit der Mathematik erklären könnten. Das Smalltalk-Manual ist ungefähr 400 Seiten dick. Das ist schon ein Indiz dafür, daß etwas nicht stimmt. Smalltalk als Sprache habe ich nur für schlecht befunden. Aber es wurde damit zum ersten Mal dieses Paradigma der Objektorientierung hervorgebracht. Das war ein echter Beitrag, der im übrigen schon in Simula latent vorhanden war. Smalltalk ist bekannt geworden wegen der überlappenden Windows.

Die Implementierung des User-Interfaces hat Smalltalk bekannt gemacht, nicht die Sprache selbst. Auf mich hat das einen Rieseneindruck gemacht. Aber heute weiß man, daß diese Techniken auf beliebige andere Programmiersysteme übernommen werden können.

Eine Sprache, in der alles objektorientiert gemacht werden muß, scheint mir kontraproduktiv zu sein. Ein Beispiel, das ich immer wieder gerne bringe: Wir haben gelernt, daß wir zwei gleichberechtigte Zahlen addieren können: $3 + 4 = 4 + 3$. Das Kommutativgesetz. Aber in der objektorientierten Welt von Smalltalk müssen Sie sich entscheiden: Die 3 ist als Objekt zu betrachten, und dem schicken Sie eine Meldung »Addiere 4 zu dir selber«.

Es gibt viele gute, etablierte Gründe, warum man die Mathematik nicht beiseite stellen sollte. Die objektorientierte Programmierung bringt in gewissen Fällen etwas, wo sie notwendig ist. Objektorientierte Techniken brauchen wir beim User-Interface.

Kann man den Compiler ohne Betriebssystem auf ein anderes Betriebssystem übertragen?

Die wirkliche Essenz von Oberon kriegen Sie nur, wenn Sie das ganze System implementieren. Sonst ist es kein großer Schritt von Modula entfernt. Die Mühe lohnt sich dann nicht mehr. Der Hintergrund ge-

hört schon dazu: Sprache und System – auch mit der zentralen Speicherverwaltung.

Ist das Gesamtsystem kommerzialisierbar?

Ja, indem man es auf einem bestehenden aufbaut. Der einzige Nachteil ist der, daß Sie das alte Betriebssystem noch im Kernspeicher haben müssen. Oberon läßt diesen Speicher dann brachliegen. Aber sonst sollte es keine Nachteile geben.

Wie waren die Reaktionen von Kollegen und von Seiten der Computerindustrie?

Bis jetzt hat die Industrie kaum Interesse gezeigt. In der Schweiz ohnehin nicht. Da gibt es ja kaum eine Computerindustrie. Es ist unser Hauptanliegen, neue Ideen zu präsentieren. Wir wollen dieses System nicht unbedingt weltweit verbreiten. Schön, wenn das passiert, aber es passiert sicher nur, wenn andere das mittragen. Wir sind ja keine Firma, wir haben ja auch keinen direkten Gewinn davon. Wir können lediglich die Idee verbreiten.

Und welche Reaktionen gab es?

Es ist vielleicht noch etwas zu früh, um das zu beurteilen. So zahlreich sind die Reaktionen nicht. Die meisten sind schon begeistert von den Ideen, aber erwähnt wird dann doch meistens Kritik an absolut irrelevanten Dingen. Aber wir haben auch sehr konstruktive Kommentare erhalten. Wer interessiert sich denn noch für ein neues Betriebssystem, ganz gleichgültig, ob es gut oder schlecht oder supergut ist, neue Ideen bringt oder nicht? Im Gegenteil, neue Ideen sind manchmal unwillkommen, weil man dazulernen muß.

Aber jeder kennt doch die Mängel der existierenden Betriebssysteme.

Die Leute stellen sich eben nicht um. Die lernen einmal MS-DOS. Dabei bleibt es. Weiterer Fortschritt ist schon stark gehemmt durch die Macht der Gewohnheit. Darum wäre es eben schön, wenn vor allem Universitäten neue Ideen mittragen, um einige Ideen bekannt zu machen. Borland hat das mit Pascal gemacht.

Was waren denn die einschneidenden Veränderungen während ihres Arbeitslebens?

Wenn es Wendepunkte gegeben hat, dann um 1976/77, als ich bei Xerox diese Personal Workstations gesehen und beschlossen habe, hier in Zürich eine Workstation zu bauen. Dann die Diversifikation in Richtung Hardware und die Konzeption des Systems mit gemeinsamer Hard- und Software. Das war für mich schon ein Wendepunkt. Programmiersprachen haben sich immer sozusagen nebenbei entwickelt.

Ich habe nie gesagt, so, jetzt entwickle ich Modula-2, sondern Modula-2 wurde nötig, weil wir eine Sprache brauchten für das Lilith System. Oder das Oberon wurde nötig, weil wir eine Sprache brauchten für das Oberon System. Wir wollten es ja zuerst in Modula programmieren. Aber dann haben wir gesehen, daß etwas ganz wesentliches fehlt: die Typenerweiterung. Die Sprache stand bei unserer Zielsetzung nie an erster Stelle.

Interview: Robert Grimm, Eva Weber

Das Oberon System

Niklaus Wirth

Beim Software Festival führte Niklaus Wirth die Besucher in einer Grand Tour durch das Oberon System. Im folgenden drucken wir die Übersicht aus Martin Reisers Buch »The Oberon System. User Guide and Programmer's Manual« ab.

Geschichtliches

Professor Niklaus Wirth hat sich lange Zeit intensiv mit Compilern, Programmiersprachen und Personal Workstations beschäftigt. Den ersten Durchbruch erreichte er mit der Programmiersprache Pascal, die er 1968 kurz nach seiner Berufung an die Eidgenössische Technische Hochschule in Zürich entwickelt hatte. Da er von der Ausbildung her Elektroingenieur ist, wählte er den ungewöhnlichen Weg, Programmiersprachen zusammen mit der Hardware zu entwerfen. Ein erstes Produkt war der Lilith-Personal-Computer, der auf dem AMD 2901 Chipsatz von Advanced Micro Devices aufgebaut ist. Ziel war, die Möglichkeiten eines Stack-orientierten Rechners zu erforschen, auf dem in Pascal geschriebene Programme laufen. Während der Arbeit an Lilith wurden die Grenzen eines Rechners mit Pascal als Systemsprache offensichtlich. Dies führte zu der Programmiersprache Modula, später Modula-2, mit der klaren Vorgabe, damit die Software der Lilith-Workstation zu unterstützen. Der Nachfolger von Lilith war Ceres-1 (entworfen von H. Eberle und N. Wirth), dessen eher konventionelles Design auf dem NS32032 Mikroprozessor basiert. Dieser Mikroprozessor wurde ausgewählt, weil er der beste kommerziell erhältliche Chip ist in Hinblick auf die Unterstützung einer Stack-orientierten, getrennt kompilierbaren Sprache wie Modula. Ceres verfügt über einen hochauflösenden Bildschirm sowie Maus und Tastatur zur Eingabe. Ein farbiger Bildschirm ist als Erweiterung möglich. Eine Harddisk dient als Speicher für nichtflüchtige Daten, und ein Diskettenlaufwerk steht für Sicherungszwecke zur Verfügung. Das erste Betriebssystem, das auf der Ceres-1 lief, war Medos. Es wurde in Modula-2 geschrieben. Auf Ceres-1 folgte bald Ceres-2 mit dem schnelleren NS32532 Chip. Zuletzt wurde Ceres-3 fertig, eine Version ohne Diskettenlaufwerk mit einem NS32GX32 Prozessor.

Ende 1985 begannen Niklaus Wirth und Jürg Gutknecht, ein neues System von Grund auf zu entwerfen, mit dem Ziel, *Erweiterbarkeit* und *Flexibilität* zu erreichen. Wirth nannte das Projekt Oberon, weil er von der Genauigkeit und Zuverlässigkeit der Weltraumsonde Voyager fasziniert war, die zur Zeit der Konzeption des Projekts gerade den Mond Oberon des Planeten Uranus passierte.

Ziel der doch recht aufwendigen Arbeit war es, Einblick in Programmiersprachen- und Systemdesign zu bekommen – unter Verwendung der *Konzepte ob-*

jektorientierter Programmierung. Unter dem Begriff Oberon sind also drei Dinge zu verstehen:

- 1) Der Name eines Projektes,
- 2) eine neue Programmiersprache, Nachfolger von Modula-2, die *Typenerweiterung* ermöglicht,
- 3) ein Betriebssystem für eine Personal Workstation.

Leitmotiv

Mache es so einfach wie möglich, aber nicht einfacher.

A. Einstein

Wirth stellte dieses Zitat an den Anfang seines Textes »Die Programmiersprache Oberon«⁽¹⁾. Meiner Meinung nach stellt dieser Satz eine ideale Zusammenfassung des Designstils dar – der Herangehensweise an die Programmiersprache wie auch an das Betriebssystem. Wirth und Gutknecht sagen dazu:

»Beim Entwurf der Hardware und der Software für Oberon folgten wir einem Leitprinzip: Wir bemühten uns um Klarheit und Einfachheit. Dies ist sinnvoll, nicht nur im Hinblick auf das kleine Team und das Bedürfnis, ein funktionierendes System innerhalb der zeitlichen Grenzen menschlicher Geduld zu schaffen, sondern einfach unerlässlich für *jedes* System mit dem Anspruch, zuverlässig zu sein. Klarheit und Einfachheit werden am besten erreicht durch eine regelmäßige und zweckgebundene Struktur. Dies ist aber nur dann möglich, wenn das zugrunde liegende Arbeitsmodell wohl durchdacht und in angemessener Weise einfach ist und keine sich gegenseitig widersprechenden Voraussetzungen aufweist.«

Die Sprache Oberon

Zum Thema Oberon als Programmiersprache möchte ich den Leser verweisen auf den Reader *The Oberon Language: Steps beyond Pascal and Modula*⁽²⁾. Die folgenden beiden Zitate von Wirth zeigen die neue Programmiersprache in ihrem angemessenen Kontext, als eine Evolution aus Modula-2:

»Anfangs war geplant, das System in Modula zu schreiben, zumal diese Programmiersprache die Vorstellung eines modular aufgebauten Systems recht zuverlässig unterstützt und über gewissenhaft gewählte Schnittstellen verfügt. Tatsächlich sollte ein Betriebssystem aus nicht mehr als einigen grundlegenden Modulen bestehen. Und der Entwurf einer Anwendung sollte als zielorientierte Erweiterung dieser grundlegenden Module verstanden werden: Programmieren bedeutet immer, ein bestehendes System zu erweitern.

Moderne Programmiersprachen wie Modula unterstützen die Vorstellung der Erweiterbarkeit, was Prozeduren angeht. Diese Vorstellung ist aber weitaus weniger vertraut im Bereich der Datentypen. Insbesondere erlaubt Modula die Definition von Datentypen als Erweiterungen von anderen, vom Programmierer definierten Typen nicht in einer angemessenen Weise. Ein weiterer Bestandteil wurde benötigt und dadurch eine *Erweiterung* von Modula.«

»Es wurde bald offensichtlich, daß die Regel, sich auf das Wesentliche zu konzentrieren und das Un-



wesentliche wegzulassen, nicht nur auf den Entwurf des neuen Systems, sondern genauso nachdrücklich auf die Sprache angewendet werden sollte, in der das System programmiert wird. Die Anwendung dieses Prinzips führte also von Modula zu einer neuen Programmiersprache. Wobei das Adjektiv »neu« im richtigen Kontext verstanden werden muß: Oberon wurde aus Modula entwickelt, mit einigen wenigen Zusätzen und einigen Weglassungen. Indem wir uns auf Evolution anstelle von Revolution verlassen, bleiben wir der Tradition einer langen Entwicklung verpflichtet, die von Algol zu Pascal, dann zu Modula-2 und schließlich zu Oberon führte.«

Aufgrund der Möglichkeit, Typen zu erweitern, erlaubt Oberon Programmierung im objektorientierten Stil. Der Design-Philosophie folgend, ermöglicht Oberon dies mit einem Minimum an Konstruktionen. Es gibt keine ausdrücklichen Konstruktionen Klasse, Methode und Nachricht. Dies ist, wie das Zitat von Wirth verdeutlicht, beabsichtigt:

»Es ist unmöglich, ausdrücklich alle Ideen anzuführen, die letztlich zu dem beitragen, was Oberon jetzt ist. Die meisten Ideen entwickelten sich aus der Benutzung oder Untersuchung bestehender Sprachen, wie Modula-2, Ada, Smalltalk und Cedar, die uns häufig zeigten, wie wir es nicht tun sollten⁽³⁾.«

Hier sollen die Bestandteile von Oberon nacheinander vorgestellt werden, unter den Gesichtspunkten Benutzerschnittstelle, Sprache und Systemarchitektur.

Die Benutzerschnittstelle von Oberon

Ein Benutzer, der auf den Bildschirm eines typischen Terminals oder Personal-Computers schaut, sieht meistens Textzeilen. Er hat das Konzept des Cursors verstanden: ein Punkt, an dem Text eingegeben oder gelöscht werden kann.

Vom System ausgegebener flüchtiger Text

```

Volume in drive C is REISER
Directory of C:\
COMMAND COM      25276   3-05-89   2:11P
DOS               <DIR>    3-04-89   5:45P
MR               <DIR>    3-09-89   8:31P
AUTOEXEC BAT     159    4-22-89   8:23P
CONFIG BAK       51    4-11-89   7:32P
CONFIG FSA       59    7-30-89   7:12P
CONFIG SYS       88    9-14-89   9:21P
       7 File(s)  113932 bytes free
C:\>dir a:

```

Kommandozeile

Die Beschaffenheit des Textes

Bei einer Anzeige wie der obigen wird der Benutzer jedoch schnell lernen, daß Text nicht gleich Text ist. Die Liste der Dateien in unserem Beispiel wurde vom System ausgegeben. Dies ist ein *flüchtiger Text*, in dem Sinne, daß er weder abgespeichert noch ausge-

druckt oder verändert werden kann. Text kann nur in der unteren Zeile eingegeben werden. In diesem Falle ist er ein Kommando. Der Benutzer hat entdeckt, daß der Text *modal* ist: Er ist entweder eine Nachricht des Systems oder ein veränderbarer Text oder ein Kommando.

Menüs und Dialogfenster

Spätere Software-Generationen fügten Menüs hinzu: Kommandos werden in Listen angezeigt und ausgeführt, wenn man mit dem Mauspfel darauf zeigt. Menüs sind wiederum etwas anderes als veränderbare Texte. Wenn ein Menükommando Parameter benötigt, dann wird ein sogenanntes Dialogfenster geöffnet. Damit beginnt ein neuer Modus. Der Benutzer muß das Dialogfenster ausfüllen, bevor er weiterarbeiten darf.

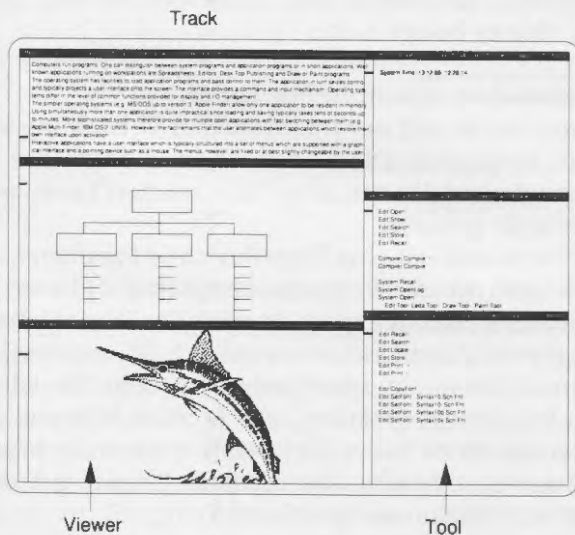
Die Benutzerschnittstelle von Oberon entfernt sich radikal von diesen herkömmlichen Modellen. Die Konzepte Kommandozeile und Menüs gibt es hier nicht. An ihre Stelle tritt ganz einfach eine Art von Text, der sich so verhält, wie es jemand erwartet, der noch ungetrübt ist von »Computerkenntnissen«: Text kann ausgetauscht, verändert, gedruckt und gespeichert werden.

Text ist Text, nicht mehr, nicht weniger.

Wir werden die Folgen dieses Grundsatzes im Laufe dieses Beitrags erforschen.

Die Anzeige von Oberon

Oberon wurde entworfen, um einen menschlichen Benutzer zu unterstützen, der mit einem Bildschirm und mit Maus und Tastatur zur Eingabe arbeitet. Es wurde bewußt eine vernünftigerweise große Anzeige ausgewählt⁽⁴⁾. Eine typische Anzeige von Oberon schaut folgendermaßen aus:



Der Bildschirm ist mit nicht überlappenden Fenstern, den *Viewern*, »gekachelt« (tiling). Die Viewers sind zu zwei Stapeln zusammengefaßt, den *Tracks*. Jeder Viewer zeigt ein Dokument an, das vom Benutzer bearbeitet wird. Dokumente können Texte, Grafiken oder Bilder sein.

Der abgebildete Bildschirm wirkt vertraut, es gibt aber ein ganz grundlegende Besonderheit: *Die verschiedenen Beschaffenheiten von Text wurden abgeschafft*. Was in den Kopfzeilen der Viewer aussieht wie ein Menü, ist ebenfalls Text, nicht anders als der veränderbare Text im Hauptbereich des Viewers.

Die Kommandoausführung

Kommandos werden einfach in einen Textviewer geschrieben und dann ausgeführt, indem man mit dem Mauspfel auf sie zeigt und eine der Maustasten klickt. Kommandos können überall im Text vorkommen. Die Kommandozeile war sinnvoll für Teletype-Systeme der sechziger Jahre. Heute ist sie ein unbrauchbar gewordenes Relikt.

Kommandoausgaben

Wenn ein Kommando eine Ausgabe auf dem Bildschirm erzeugt (wie das Inhaltsverzeichnis-Kommando weiter oben), wird ein neuer Textviewer mit dem Ausgabebetext des Kommandos geöffnet. Auch dieser Text kann verändert, gespeichert oder gedruckt werden. Oberon-Kommandos erzeugen *nichtflüchtigen* Output.

Tools: die Brücke zwischen »Erinnern und Tippen« und »Zeigen und Klicken«.

Aus der Abschaffung der verschiedenen Beschaffenheiten von Textanzeigen und Kommandozeilen ergibt sich eine machtvolle Vereinigung der Metaphern »Erinnern und Tippen« und »Zeigen und Klicken«.

Tool

Kommandos werden in einen Textviewer eingegeben (mit dem Standardeditor von Oberon) und dann mit der Maus ausgeführt. Es ist deswegen recht sinnvoll, sich einige oft benutzte Kommandos als Text vorzubereiten, der auf der Platte abgespeichert ist. Solch ein Text wird *Tool* genannt.

Wenn ein Tool angezeigt wird, ähnelt es einem Menü. Einige Kommandos werden aufgeführt, und der Benutzer führt sie einfach mit der Maus aus. Wenn das Kommando jedoch die Eingabe von Parametern erfordert, dann werden keine aufwendigen (modalen) Eingabe- oder Dialogfenster benötigt. Wir haben es mit veränderbaren Text zu tun, und die Parameter können einfach eingegeben werden, womit die Flexibilität von »Erinnern-und-Tippen«-Umgebungen erreicht wäre. In der Tat lassen Oberon-Tools in fast idealer Weise die beiden Welten ineinander übergehen, die beide bis heute sowohl ihre Kritiker als auch Anhänger hatten.

In der Wissenschaft kann man häufig die Erfahrung machen, daß Prinzipien, die mächtig genug sind, verschiedene Denkbereiche zu vereinen, wiederum in neue Gebiete führen. Auch Oberon entspricht diesen Erwartungen. Die Verbreitung von Systemversionen ist ein allgemein bekanntes Problem. Zwei Dinge werden benötigt: ein Verteilermedium und eine Folge von Kommandos, um die neue Version zu

installieren. Die zweite Anforderung ist sowohl ermüdend als auch fehleranfällig. In Oberon wird ein einfaches Memorandum ausführbar, das die Folge von Kommandos erläutert. Die Kommandos werden einfach angeklickt, eines nach dem anderen, um die neue Version zu installieren. Die folgende Abbildung ist eine Electronic-Mail-Nachricht, die ausgeführt werden kann, um das System von einem Dateien-Server herunter zu aktualisieren:

Einfach ausführen durch Anklicken mit der Maus

Ein Oberon-Kommando, um die Dateien einer neuen Systemausgabe zu laden

Submission: 14.07.89
 Originator: Wirth
 Recipient: CS.all
 Subject: New Oberon Release

To actualize your Oberon system to the new release, execute the following steps:

Net.ReceiveFile Pluto
 CereS.Boot Cursors.Bbj Diskette.Obj Display.Obj Edit.Obj Files.Sym
 Fonts.Obj C2.Input.Obj Mailer.Obj Net.Obj Oberon.Obj Oberon.Sym
 Printer.Obj Reals.Obj C2.SCC.Obj System.Obj TextFrames.Obj
 TextFrames.Sym TextViewers.Obj TextViewers.Sym Texts.Obj
 Texts.Sym Viewers.Obj Viewers.Sym Viewers.Obj Viewers.Sym
 C2.V24.Obj-
 ...

Die Systemarchitektur von Oberon

Nur einige wenige der vielen Innovationen können hier in diesem Überblick dargestellt werden.

Das objektorientierte Design

Die Ideen der Schichtung und der Datenabstraktion sind heute klar. Die Verwendung der Typenerweiterung führt aber ein neues Element in die Architektur von Oberon ein. Objekte – genauer: aktive Objekte – sind Instanzen abstrakter Datentypen, die durch Verzeichnisse mit einer Prozedurvariable, dem *Handler* eines Objektes, repräsentiert werden. Die Parameter für den Handler sind die Felder einer Record-Variablen, *Nachricht* genannt. Wenn man die Felder der Nachricht füllt und den Handler ruft, dann bezeichnet man dies auch als »einem Objekt eine Nachricht schicken«. Im Gegensatz zur traditionellen Objektorientierung versucht Oberon nicht, die Konzepte der Klasse und der Methode zu verwirklichen.

Auf Instanzen konzentrierte Objekte

Oberon führt ein objektorientiertes Design ein, das auch als *auf Instanzen konzentriertes Design* bezeichnet werden kann. Das Binden von Prozeduren an ein Objekt wird verzögert und erst zur Laufzeit ausgeführt. Wir sprechen von der *Installation eines Handlers* in einem Objekt. Unter dem Oberon-Paradigma werden Nachrichten (oder Parameterblöcke für den Handler) vom Benutzer definiert, nicht im Modul, wo das Objekt definiert wird.

Module, Kommandos und abstrakte Datentypen

Traditionelle Systeme haben Einrichtungen, um Programme laufen zu lassen. Einmal geladen und gestar-



tet, erhält das Programm die Kontrolle. Es läuft dann typischerweise eine lange Zeit, bis es angehalten wird und das Betriebssystem ein neues Programm aufruft. Wenn ein Programm beendet wird, dann gibt es seine ganzen Ressourcen frei, insbesondere den Speicher, den es belegt hatte.

Kommandos, dynamisches Laden

Oberon bricht mit der traditionellen Vorstellung von einem Programm. Die Code-Einheit, die von der Benutzerschnittstelle her ausgeführt werden kann, wird *Kommando* genannt.

Ein Kommando ist eine Prozedur ohne Parameter, die von einem in der Programmiersprache Oberon geschriebenen Modul exportiert wird. Da Effizienz beim Aufruf von Kommandos unerlässlich ist, müssen die Kommandomodule speicherresident sein. Es ist jedoch keineswegs praktisch, alle Module während des Bootvorgangs zu laden. Deswegen werden die Module *dynamisch geladen*, bei Bedarf. Einmal geladen, bleibt ein Modul im Speicher.

Abstrakte Datentypen

Ein Modul implementiert oft einen oder mehrere abstrakte Datentypen. Die Tatsache, daß Module, wenn sie einmal geladen werden, im Speicher resident bleiben, hat eine wichtige Konsequenz: Instanzen abstrakter Datentypen können nun während der gan-

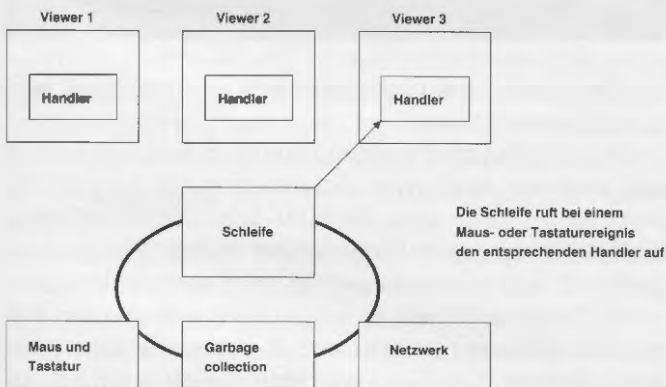
zen Sitzung existieren: *Kommandos können mit Instanzen abstrakter Datentypen arbeiten und durch sie kommunizieren.* Der abstrakte Typ *Text* ist ein wichtiges Beispiel.

Der Kommando-Interpreter und Multitasking

Jedes System benötigt einen Kommando-Interpreter. Wenn ein traditionelles System völlig unbeschäftigt ist, dann liegt die Kontrolle bei einer Schleife des *Kommando-Interpreters des Systems*, der auf eine Eingabe wartet. Diese Eingabe kann dann ein Kommando sein, um eine interaktive Anwendung zu laden. Einmal geladen, wird die Kontrolle an den *Kommando-Interpreter der Anwendung* übergeben, der seine eigene Abfrageschleife hat. Der Benutzer kann also entweder System-Kommandos eingeben oder die Kommandos, die von der Anwendung verstanden werden.

Offensichtlich benötigt ein System, das mehrere Fenster mit verschiedenen Anwendungen zuläßt, in gewissem Umfang Multitasking. In den bekannten Systemen bedeutet dies Unterbrechung von Programmen und Abspeichern der Informationen über den Zustand des Rechners. In Oberon befinden sich alle Kommandos, die auf ihre Ausführung warten, auf der gleichen Ebene. Eine neuartige Architektur wird benötigt, die folgende Eigenschaften aufweist:

- Es gibt nur eine Schleife - die *Ereignisschleife* -, die in ein System-Modul eingekapselt ist.
- Die unteilbare Einheit eines Arbeitsganges ist der Aufruf einer Prozedur.



Ein Blick auf die Anzeige von Oberon zeigt mehrere koexistierende Viewer, jeder ist einer spezifischen Aufgabe wie der Edierung von Texten, Zeichnen, Entwerfen etc. gewidmet. Die Viewer sind als aktive Objekte verwirklicht.

Wenn nichts passiert, bleibt die Kontrolle bei der Ereignisschleife, die ständig die Gerätetreiber abfragt. Wurde ein Ereignis festgestellt, dann wird eine Nachricht, die das Ereignis identifiziert, an den Handler des betroffenen Viewers geschickt. Der Handler stellt fest, was als Konsequenz auf das Maus- oder Tastaturreignis geschehen muß. Er ist zugleich der Anzeigen-Manager, der für die Ausgabe verantwortlich ist. Nach Abschluß des Handleraufrufes erhält die Schleife die Kontrolle zurück.

Die wichtige Konsequenz dieses Designs ist, daß die Kette der normalen Prozeduraufrufe niemals unterbrochen wird. Es gibt keine Statusinformation, die für Multitasking abgespeichert werden muß. Es gibt keine versteckten Zustände, und der Einzelprozeß der zentralen Schleife kann zugleich an verschiedenen Aufgaben des Benutzers arbeiten - ohne die Komplikationen, die durch echtes Multitasking entstehen.

Speicherverwaltung und Garbage Collection

Oberon benutzt die Speicherverwaltungseinheiten moderner Mikroprozessoren, um die Module (Programmsegmente) im Speicher zu verteilen. Module werden bei Bedarf geladen und bleiben dann speicherresident. Der Speicher des Computer wird in einen *Stack* für die lokalen Variablen der Prozeduren und einen *Heap* aufgeteilt. Garbage Collection wird dazu verwendet, um die Größe des Heap auf das Nötigste zu beschränken. Die Einführung der Garbage Collection geschieht nicht nur aus Gründen der Bequemlichkeit, sondern auch, um *Zuverlässigkeit des Systems zu erreichen.* Tatsächlich sollte dem Programmierer nicht zugetraut werden, freien Speicher ständig korrekt anzuweisen.

Abstrakte Dokumente - das Beispiel der Texte

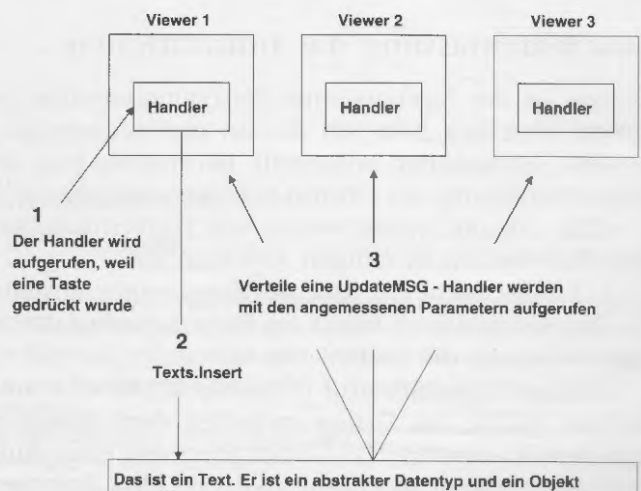
Viewer arbeiten typischerweise mit Dokumenten, die innerhalb ihres Rahmens dargestellt werden. Solche Dokumente sind Texte, Grafiken, Bilder etc.

Text als ein abstrakter Datentyp

Es wurde bereits gesagt, daß Texte eine spezielle Rolle unter Oberon spielen. Ein Text ist eine Instanz eines abstrakten Datentyps, der vom Modul *Text* exportiert wird. Er ruft die Vorstellung von einer Folge an Bytes mit den zugehörigen Eigenschaften hervor.

Text als ein Objekt

Texte sind also aktive Objekte. Wenn eine der Prozeduren die grundlegenden Daten ändert, dann wird eine Nachricht an alle sichtbaren Viewer geschickt, die diese



Änderung bekannt gibt. Wenn ein Viewer den geänderten Text anzeigt, wird er also seine Anzeige auf den neuesten Stand bringen. *Ein Dokument zu ändern und eine Anzeige auf den neuesten Stand zu bringen, sind also zwei völlig verschiedene Dinge.* Ein Text zeigt sich selber an, nachdem er geändert worden war.

Erweiterbarkeit

Die Erweiterbarkeit des Oberon Systems ist ein wichtiges Entwurfsziel. Dies kann bedeuten:

– Man kann Kommandos hinzufügen, die aus Texten heraus ausgeführt werden und vorhandene abstrakte Dokumente (zum Beispiel Texte) bearbeiten.

– Man kann neue Viewer-Klassen hinzufügen, die aus folgendem bestehen: (a) einem abstrakten Dokument, (b) einem Viewer mit seinem Handler und (c) einer Anzahl von Kommandos, die aus Texten heraus ausgeführt werden können, einschließlich einem Öffnen-Kommando, das eine Instanz des Viewers erzeugt.

Hinzufügen von einfachen Kommandos

Kommandos, die Dokumente bearbeiten – Instanzen von abstrakten Datentypen –, können ohne Schwierigkeiten jederzeit hinzugefügt werden. *Das Verbot versteckter Zustände sichert, daß solche Kommandos niemals mit den Handlern, die solche Dokumente verwalten, in Widerstreit geraten.* Der Programmierer muß nur noch ein Objektmodul erzeugen, das das Kommando enthält.

Das dynamische Ladesystem erlaubt die Benutzung des Moduls, ohne vorher Bindevorgänge durchführen zu müssen.

Hinzufügen von Viewer-Klassen

Die mächtigste Erweiterung ist das Hinzufügen einer Viewer-Klasse. Die Architektur des Oberon Systems ermöglicht es dem Benutzer, dem System sowohl abstrakte Dokumente als auch neue Arten an Viewern hinzuzufügen, ohne eine Installationsprozedur oder einem erneuten Compilieren des Systemcodes. Daß dies innerhalb dieser stark festgelegten Umgebung möglich ist, ist ein signifikanter Erfolg des objekt-orientierten Designs.

Zusammenfassung der Innovationen

Oberon ist das Ergebnis eines Forschungsprojektes. In diesem Überblick habe ich die am meisten herausragenden Eigenschaften vorgestellt. Ich möchte hier die Zusammenfassung der Oberon-Erfinder wiedergeben⁽⁵⁾:

»Das Oberon System weicht von konventionellen Betriebssystemen in einigen Aspekten ab:

1) Es gibt nicht die Idee des Programms; anstelle des Programmaufrufs bildet ein Prozeduraufruf durch den Anwender die Einheit der Aktion.

2) Jeder Prozeduraufruf (Kommando) ist eine unteilbare Aktion im Dialog zwischen dem Benutzer und dem Computer: Das Umschalten von einer Aufgabe zur anderen geschieht zwischen den Befehlen



des Benutzers, und nicht zwischen zwei willkürlichen Maschinenbefehlen.

3) Kommandos erhalten ihre Eingaben aus Texten und anderen Arten von Dokumenten, nicht von der Tastatur. Anstelle des direkten Schreibens auf dem Bildschirm erzeugen Kommandos nichtflüchtige Ausgaben in Form von (angezeigten) Datenstrukturen.

4) Die Schnittstelle zwischen zwei aufeinander folgenden Aktionen besteht aus abstrakten Datenstrukturen (Texten, Grafiken) im Hauptspeicher, und nicht in Dateien oder auf der Platte. Wenn diese in Viewern angezeigt werden, sind sie editierbar.

5) Oberon stellt eine verteilte Kommando-Interpretation zur Verfügung. Viewer werden als rechteckige Bereiche auf dem Bildschirm angesehen, die alle für sich Kommandos interpretieren können. Um dies zu erreichen, wird das Paradigma der objekt-orientierten Programmierung verwendet. Eine Nachricht wird dann an einen Viewer geschickt, wenn eine Eingabe sich auf diesen bezieht.

6) Oberon hat ein einfaches und extrem effizientes Dateiverwaltungssystem. Das Inhaltsverzeichnis der Platte ist als B-Baum organisiert. Es wird klar unterschieden zwischen einer Datei und dem Zugriffsmechanismus, der über sogenannte Rider funktioniert.

7) Module werden unter Oberon nur dann geladen, wenn sie wirklich gebraucht werden. Das verzögerte Laden ist deswegen wichtig, weil Programmpakete aus Dutzenden von Modulen bestehen können, von denen nur wenige für eine spezifische Anwendung benötigt werden. Das verzögerte Laden wird von Seitenfehlern kontrolliert, die vom virtuellen Adreßmechanismus herrühren.

8) In den Oberon Kernel wurde ein Garbage Collector eingebaut. Dieser läuft nicht als getrennter Prozeß, sondern wird ausdrücklich zwischen Kommandos aufgerufen, unter der Voraussetzung eines leeren Stack. Diese Voraussetzung vereinfacht und beschleunigt den Algorithmus erheblich.

9) Das System und die Benutzerpakete werden in einer Sprache implementiert, die mit Datentypenerweiterung und vielgestaltigen Befehlen bei garantierter Typensicherheit versehen ist. Eine vollständige Typensicherheit ist unerläßlich für ein System, das auf automatischer Wiederherstellung des Speichers basiert.

10) Eine Implementation von Oberon kann dadurch erweitert werden, daß neue Datentypen deklariert werden, die Erweiterungen von bestehenden, importierten Typen sind. Objekte aus erweiterten Datentypen sind kompatibel mit Objekten des Basistyps und können deswegen in bestehende Datenstrukturen integriert werden.

11) Im Oberon System gibt es keinen wirklichen Unterschied zwischen Benutzern und Programmieren. Mit einer mächtigen Basis an Modulen als Grundlage können Benutzer das System erweitern oder an ihre Bedürfnisse anpassen, indem sie neue Tools programmieren.«

Die Fortentwicklung immer schnellerer Computer mit größerem Speicher hält unvermindert an. Die Software, die wir benutzen, ist dagegen normalerweise veraltet und wird nur langsam angepaßt. Eine Ebene an Code wird über die nächste gelegt, was zu Systemen von enormer Größe führt. Es ist also recht verbreitet, daß das Betriebssystem einer Workstation 1 MByte braucht und Textverarbeitung und Tabellenkalkulation noch einmal 4 MByte benötigen. Trotzdem wird die ganze Sache recht langsam laufen auf der heutigen Generation von schnellen Workstations.

Oberon hingegen beweist, daß dies nicht so sein muß und daß es kein Softwarehindernis gibt, das nicht überwunden werden kann. Das ganze System ist in 15 000 Zeilen Quelltext ausgedrückt, und der kompilierte Code umfaßt 150 KByte. Also lohnt sich das Bemühen, seine Ressourcen nicht zu verschwenden. Oberon ist ein hochentwickeltes System und funktioniert genau so gut, wenn nicht sogar besser, wie bekannte kommerzielle Systeme.

System-Versionen, Implementationen und Anwendungen

Die Prämisse von Oberon ist die eines offenen Systems, das seinen Benutzer dazu einlädt, die Funktionsweise zu ändern und neue Funktionen hinzuzu-

fügen. Tatsächlich steht die Hoffnung dahinter, daß die Unterscheidung zwischen Benutzer und Programmierer mehr und mehr verschwindet. Eine Konsequenz davon wird sein, daß zwei Systeme kaum noch genau gleich sein werden.

Versionsnummer

Die Versionsnummer des in dieser Übersicht beschriebenen Systems ist 1.2.

Anwendungen

Am Institut für Computersysteme wird an mehreren Projekten gearbeitet, die Funktionalität durch neue Viewer-Klassen hinzufügen sollen:

- eine Vielzahl an Programmeditoren, die Erweiterungen des grundlegenden Oberon Editors sind,
- zwei Grafikeditoren *Graph* und *Oil*, die Linienzeichnungen mit der Maus zulassen,
- ein Dokumenteneditor *Leda*, der fortgeschrittene Textverarbeitungs- und Seitenlayout-Funktionen nach WYSIWYG-Manier zur Verfügung stellt,
- ein Malprogramm *Paint* zur Manipulationen von Bitmaps.

Object Oberon

Object Oberon ist eine kleine Erweiterung der Sprache Oberon, die die Konzepte der Klasse und der Nachricht aus der objektorientierten Programmierung einführt.

Es wurde von P. Mössenböck und J. Templ entwickelt. Diese Autoren veränderten auch die Funktionsweise der Viewer-Klasse – und schufen dadurch eine neue Oberon-Familie.

Implementationen

Oberon läuft nun schon einige Jahre auf der Ceres, der am Institut für Computersysteme gebauten experimentellen Workstation. Implementationen für Sun Sparcstations und Apple Macintosh Computer sind öffentlich erhältlich. Die Implementationen für andere Systeme (DECsystem 3100, IBM PS/2 und IBM RISC System/6000) sind in verschiedenen Stadien der Fertigstellung.

Martin Reiser; Übersetzung: Robert Grimm

Anmerkungen

- 1) Wirth N. (1988). Software – Practice and Experience, 18(7), 671-90
- 2) Reiser M. und Wirth N. The Oberon Language: Steps beyond Pascal and Modula, Addison-Wesley, wird noch veröffentlicht.
- 3) Wirth N. (1988). From Modula to Oberon. Software – Practice and Experience, 18(7), 662-70. (Kursivschrift vom Autor hinzugefügt).
- 4) Auf der Ceres 1024 auf 800 Pixel.
- 5) Wirth N. und Gutknecht J. (1989). The Oberon system. Software – Practice and Experience, 19(9), 890.

Turbo-Kommunikation im Jazzrhythmus

Philippe Kahn

Müssen Maschinen kryptisch sein? Nein, meint der große Kommunikator bei Borland: Zukünftig wird jeder Programme schreiben können. Das neue Lied von der Kommunikation zwischen Mensch und Maschine.

Kommunikation hat einen enormen Einfluß auf alles, was heute in der Welt passiert. Kommunikation und Technologie bilden eine Einheit, wie es sie bislang nie gab. Alle möglichen Geräte sind inzwischen mit Mikroprozessoren ausgestattet: Faxkopierer, Telefone, Waschmaschinen und so weiter. Der Mensch muß mit der Maschine kommunizieren. Doch die Schnittstelle zwischen Mensch und Maschine ist oft miserabel ausgestattet. Wenn Sie einen Videorecorder zu Hause haben, wissen Sie, wie kompliziert die Bedienung dieser Geräte ist. Versuchen Sie mal, mit dem Recorder eine bestimmte Sendung aufzuzeichnen, die regelmäßig um 11.15 Uhr ausgestrahlt wird. Doch all das wird allmählich funktionieren – das Mensch-Maschine-Interface wird besser.

Wenn Leute von Geschichte sprechen, dann reden sie meist von politischen Regimen, Königen, Königinnen, Schlachten, Kriegen und so weiter. Betrachten wir die Entwicklung doch einmal aus einer anderen Perspektive und untersuchen den Einfluß der Technologie. Plötzlich offenbart sich eine noch eine interessantere Geschichte: Revolutionen wie Stahl, Dampf, Elektronik, Transistoren, Mikroprozessoren haben die Welt vielleicht noch stärker geprägt als manche Ideen, die von den meisten Menschen nach einer Weile vergessen werden.

Letzten Sommer besuchte ich in Moskau die Akademie der Wissenschaften; unsere Gastgeber waren überrascht, als wir einen elektronischen Brief aus San Francisco empfangen. Innerhalb weniger Sekunden konnten wir Daten übertragen zwischen Gebieten, die sich früher als Ostblock und Westblock gegenüberstanden. Ich bin davon überzeugt, daß Faxmaschinen mehr zur Veränderung der Welt beigetragen haben als politische Ideen. Sie erlauben den Menschen, miteinander zu kommunizieren. Totalitäre Regime versuchen immer zuerst, den Informationsfluß zu kontrollieren.

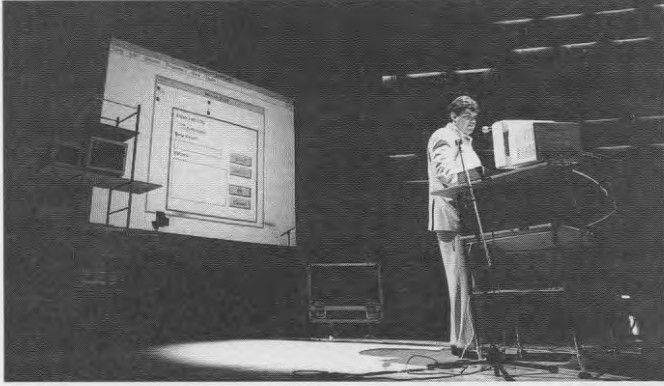
Computer sind ein Teil der Kommunikation, denn sie sind zu einem festen Bestandteil der meisten Geräte geworden. Manchmal möchte man sie am liebsten herausnehmen oder zerstören, aber sie ersetzen viele Funktionen, die man früher manuell erledigen mußte. Wir sind gezwungen, mit dem Computer zu kommunizieren, der immer nur eine bestimmte Sprache versteht.

Es gibt völlig verschiedene Arten des Software-Designs, je nach dem, was Sie vorhaben. Nach einer bestimmten Design-Auffassung sollen die Programme einfach ein elektronisches Äquivalent für das sein, was man im Schreibwarenladen kaufen kann. Zum Beispiel Scheren: Was wäre das Äquivalent? »Ausschneiden« und »Einfügen«. Dahinter steht die Annahme, daß die meisten Computer-Nutzer gewohnt sind, mit normalen Gegenständen umzugehen: Büroklammern, Kugelschreiber, Klebstoff, Scheren, Heftklammern, Notizbücher und so weiter. Das ist eine Möglichkeit, ein Interface zu gestalten.

Die meisten Leute, die jetzt hier in diesem Saal sind, sind es gewohnt, in Pascal zu programmieren, in C oder in C++. Diese Sprachen sind eine sehr merkwürdige Art zu kommunizieren. Es sind recht bizarre Sprachen, aber ziemlich brauchbare.

Es existiert ein stufenweiser Übergang zwischen menschlicher Sprache und Computersprache. Wenn Sie versuchen, mit dem Computer englisch zu reden, bekommen Sie eine Fehlermeldung: »error« oder »unknown command«. Computer verstehen Assemblersprache oder binäre Maschinensprache. So hat man früher mit dem Computer geredet. Dann wurden die strukturierten Sprachen erfunden. Die Algol-Leute sagten: »Hey, das hier könnte besser sein.« Es ist besser, weil man komplexe Befehlsgruppen durch eine höhere Abstraktionsebene ausdrücken kann. Eine prozedurale Herangehensweise.





So haben wir die unstrukturierten, binären Sprachen hinter uns gelassen. Der nächste Schritt waren leicht bedienbare Makrobefehle. Mit Makros kann man alle verwendeten Befehle aufzeichnen. Das kann man zum Beispiel bei einer Datenbank oder Tabellenkalkulation machen. Und schließlich gibt es common english oder graphische Symbole. Es ist natürlich der Traum eines Programmierers, dem Computer einfach zu sagen, »Bau mir ein Buchhaltungsprogramm!«, nach einer Weile wiederzukommen – und das Buchhaltungsprogramm ist fertig. Solange man noch daran arbeitet, nennt man das »künstliche Intelligenz«, wenn es aber mal funktioniert, wird man es auch nur ein »Programm« nennen.

Es gibt einen Übergang von prozeduralen zu deklarativen Sprachen. Die nächste Abstraktionsstufe ist, was wir objektorientiertes Programmieren nennen. Wie Professor Niklaus Wirth schon sagte, ist es eine natürliche Erweiterung des strukturierten Programmierens. Wir wollen nicht immer alles wieder neu lernen müssen, sondern arbeiten. Daher ist objektorientiertes Programmieren wie bei Pascal plus Objekte oder C++ brauchbar. Eine der Visionen, die wir bei Borland haben, ist: jedem die Gelegenheit geben, selbst zu programmieren. In der Wirtschaft gibt es ungeheuer viele Leute, die Anwendungen entwickeln müssen. Denen muß man eine unkomplizierte, nicht-algorithmische Methode an die Hand geben, um Gesetze und Prozeduren zu deklarieren.

Eines der größten Hindernisse für die Anwendung von Computern ist, daß man Software nicht schneller schreiben kann. Unsere Vision ist: Einer großen Schicht von Anwendern das Programmieren zu ermöglichen, nicht nur den professionellen Programmieren. Das nennen wir »visual programming«, die deklarative Art, Programme zu schreiben. Und dieser Gedanke steht hinter Turbo Pascal für Windows und unserem neuesten Produkt ObjectVision.

»Turbo Disturbo«

(Eines der Stücke, die »Mr. Turbo« Kahn am ersten Festival-Abend mit Münchner Jazzern einspielte)

They're comin' man, start runnin' man
Scram, skee-daddle, jump for your saddle
Hide, hide, hide in your huts
Here comes »Pecan and The Nuts!«

Moi?

Turbo, Turbo, I won't get disturbed though
Mentally active and I'm ready to learn so
Let's take the time to understand the lines
And communicate on a higher plane with rhyme
So many chains remain and they'll hold thee
So many brains it's a shame that they control thee
But look at the World, so many changes to see
And Communication's settin' people free
They built their walls and their barbed wire
fences,
Tell you what to read if it passes their censors
We've had Hitlers and we've had Stalins
We've had Pol Pots and Freedom's been calling
And now the world will set itself free
And Communication is the key
So set yourself free, get a Personal Communicator
Yeah, go ahead and make your mind greater

Signs of the times are written in the minds
Of those who choose to break the lines
Of Communication without hesitation
With Information to raise a whole Nation
A world perfected is a World connected
The death of Ignorance shall not be resurrected
So I'm here to tell you, can't you see
that Communication is setting you free

C'mon Mam, climb aboard,
Cut all them chains with your Silicon sword
You can be a Cubist
Or you can be Pacifist
You can be a Flutist
Or you can be an Activist
A Cellist, Ventriloquist, Alchemist or Herbalist
You can be a Columnist
Or you can be a Journalist
Whatever you do, take your time to embark on
The Concept to communicate and you'll sparkle
And in time you'll shine brighter
than the brightest of stars



And there you are

Riding the wave of success, Best dressed
and living the life I must confess
Here I am to tell you what I see
'cuz communication is setting you free

Oekophon und Café Hypermedia

Das Bonner Klangkollektiv

Ein Garten, ganz ohne Gemüse, Blumen oder Gartenzwerge – kein natürlicher Raum, sondern ein mit Lichtschranken, Computer und MIDI-Software ausgestatteter »Klanggarten« sensibilisiert seine Besucher für ökologische Zusammenhänge.

In Bonn-Beuel, rechts vom Rhein, zwischen Eisenbahnanlagen, Chemieunternehmen und Industriebrachen haben Künstler- und Theatergruppen in einem alten Industriegebäude Quartier bezogen. In einer ehemaligen Tapetenfabrik, in der immer noch riesige Druckwalzen als Relikte der Technikgeschichte stehen, befindet sich auch das Atelier »Chip Arts« des Bonner Klangkollektivs, das gewissermaßen die öffentliche Erscheinungsform der Bonner Entwicklungswerkstatt für Computermedien (BEC) ist und aus den Medienkünstlern Bodo Lensch, Nico Pezer, Norbert Stein, Viola Kramer, Reinhold Knieps und Herbert Gödderz besteht.

Bonn, die kleine Stadt am Rhein, hat als Entwicklungsstätte der neuen Medien eine große Tradition. Schon in den fünfziger Jahren hat der Physiker und Phonetiker Werner Meyer-Eppler an der Bonner Universität Grundlagen für die Entwicklung elektronischer Musik und intermedialer Kunstformen gelegt, wie sie heute mit Hilfe von Computern realisiert werden. Bei Meyer-Eppler hat auch Karl-Heinz Stockhausen studiert, der sich später mit seinem Experimentalstudio in Köln niederließ. »In Bonn stand in den fünfziger Jahren eine Wiege der elektronischen Kunst«, meint Bodo Lensch, der Gründer der Entwicklungswerkstatt für Computermedien. Das Klangkollektiv will daran anknüpfen und wissenschaftliche Erkenntnisse mit künstlerischen Projekten verbinden.

Bodo Lensch hat Medizin studiert und absolvierte parallel ein Studium an der Musikhochschule in Hannover. Als Musiker in einer Band sammelte er Ende der siebziger Jahre erste Erfahrungen mit multimedialen Ausdrucksweisen. Die Gruppe 'Boury' bezog zunächst Tanz und Graffiti ins musikalische Konzept mit ein. Anfang der achtziger Jahre installierte 'Boury' erste Klangräume, die aus Mehrspur-Tonbandgerät-Zuspielungen und elektronischen Verfremdungen der akustischen Instrumente erzeugt wurden – damals noch ohne Computer.

Die Entwicklungswerkstatt für Computermedien, die Bodo Lensch 1985 zunächst in winzigen Hinterhofräumen gründete, begann sehr bescheiden mit Aufführungen in studentischen Wohnheimen. Damals inszenierte BEC eine Klang-Raum-Komposition des Kölner Saxophonisten Norbert Stein, der bis heute zum Klangkollektiv gehört.

Die darin zumindest ansatzweise erreichte Ergänzung von Malerei, phonetischer Kunst, Computer-

kunst und Live-Musik zu einem »Gesamtkunstwerk« genügte den Bonner Medienkünstlern aber auf Dauer nicht. Die Arbeit der Entwicklungswerkstatt wandte sich dem Phänomen »Interaktion« zu. Die BEC experimentierte damit, einen sich im Raum bewegenden Tänzer als Computer-Input einzusetzen: Der Tanz sollte die dreidimensionale Wiedergabe der Musik steuern und in einem weiteren Schritt auch die Entstehung der Musik selbst. »Dieses Ziel haben wir 1987 dokumentiert mit Hilfe eines Echtzeit-Bildverarbeitungsgeräts«, erinnert sich Bodo Lensch. »Die Hardware hatten wir irgendwie zusammengebettelt, die Software hatten wir in Ockham, einer seinerzeit nicht sehr verbreiteten Computersprache, selbst geschrieben. Mit Hilfe der Echtzeit-Bildverarbeitung konnten wir schließlich ein MIDI-System steuern, was damals noch ein Novum war.«

Tanz im Klanggarten

Das wesentlich größere Atelier in der Tapetenfabrik brachte der Entwicklungswerkstatt seit 1989 die Möglichkeit größerer, fester Installationen. Im Atelier »Chip Arts« wurde ein stationärer Klangraum installiert, in dem die Gruppe ihre bislang umfangreichste multimediale Komposition, den »Klanggarten«, vorführen kann. Bodo Lensch: »Unser Klangraum besteht aus einhundert interaktiven Lichträumen, die ähnlich wie Lichtschranken funktionieren. Metallplatten auf dem Boden sind mit Foto-Widerständen ausgestattet, die mit Halogenscheinwerfern mit einem relativen engen Öffnungswinkel und 64 Dimmstufen beleuchtet werden. Die Abschattung jeder Platte wird von je sieben Sensoren gemessen. Über diese Controller werden die Platten gewissermaßen als Regler definiert, so daß wir sie als steuerndes Eingabemedium des Klanggartens benutzen können.« Die MIDI-Software sorgt dafür, daß die Lichträume manipuliert werden können.

Die Performance der Gruppe in ihrem »Klanggarten« besteht aus zwei Teilen. Zuerst wird eine »Orchesterversion« mit Live-Konzert inszeniert, bei der sich Musiker und Tänzer im Klanggarten bewegen. Sie reagieren mit Bewegungen auf Klänge, die sie selbst auslösen, indem sie verschiedene Areale betre-





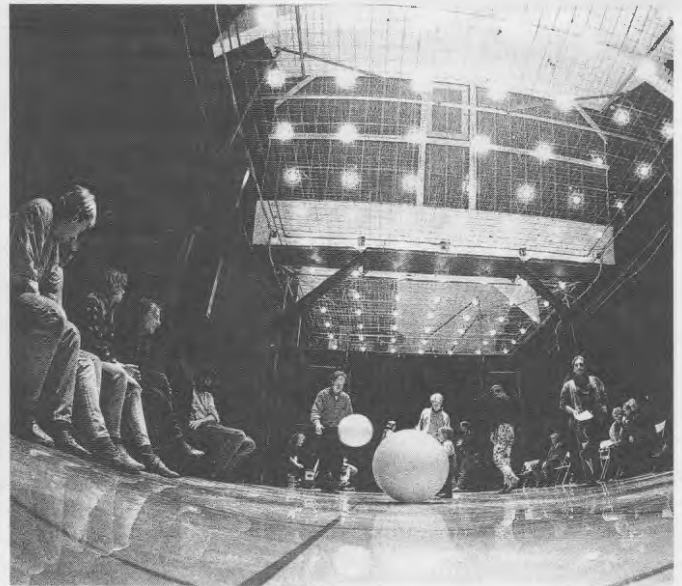
ten. Die Komponistin und Choreographin Viola Kramer benutzt auch Fächer, um – wie eine Dirigentin – durch die Abschattung von benachbarten Feldern Tempo, Tonhöhe, Notenlänge und Modulation zu verändern.

Der zweite Teil der Aufführung ist das sogenannte »Oekophon«. Eine Grundstruktur von einem in sich stimmigen System aus Licht und Tönen wird vorgegeben. Zuerst bevölkern nur diese »Klangorganismen« den Raum. Sobald Gäste den »Garten« betreten, wird die Grundstruktur – über die abgeschatteten Felder – in Unordnung gebracht. Verschiedene »Organismen« reagieren auf das Eindringen eines Menschen aggressiv, andere entfernen sich oder verstummen, einige »sterben« gar. »Wenn Menschen den Raum zu brutal betreten und überhaupt nicht darauf achten, was dabei passiert«, erklärt Bodo Lensch, »kommt es zum Infarkt.«

Freie Interaktion

Mit dem »Oekophon« will die Gruppe den Besuchern »ökologische Sensibilisierung« vermitteln. Der Besucher kann das Kunstwerk miterzeugen, seiner spielerischen Freiheit sind nur durch die vorgesehenen Plattendefinitionen Grenzen gesetzt. Dahinter steckt ein emanzipatorischer Grundgedanke, vergleichbar mit der Skulpturtheorie von Franz Erhard Walther. Der »Klanggarten« soll nicht das einzige größere Pro-

jekt des Kollektivs bleiben. Bislang jedoch war das Zuschußunternehmen finanziell immer wieder in Frage gestellt. Förderung erhielt es bisher vor allem im Rahmen einer Modellförderung »Kreative Medienarbeit« vom Bundesministerium für Bildung und Wissenschaft. Auch vom Kultusminister Nordrhein-Westfalen flossen Gelder über das Büro für Freie Kulturarbeit in Dortmund. Doch das etwa eine halbe Million Mark teure Equipment des »Ateliers Chip Art« wäre sicherlich nicht ohne großzügige Industriensponsoren zusammengekommen. Der Erfolg des großen Bonner Chip Alchemie Festivals Ende 1990 läßt



Bodo Lensch große Pläne schmiedet: Er denkt an eine internationale Vernetzung von Multi-Media-Ateliers und plant ein »Café Hypermedia«, bei dem verschiedene Künstler aus mehreren Ländern Europas über ISDN miteinander verbunden werden. Die Künstler sollen dann via Telekommunikation ein Projekt vorbereiten, das als multimediale Schau in einer »interaktiven Art Gallery« in Bonn zusammengefaßt werden kann. Auch ein »poeme interactif« ist in Arbeit, bei dem sich das BEC in Zusammenarbeit mit »Lieutenant Caramel« aus Lyon und der »Bonner Werkstatt für experimentierendes Theater« mit Marcel Duchamp beschäftigen will.

Längerfristig plant das Bonner Klangkollektiv Projekte mit computergesteuerten Video-Bühnenbildern oder Cyberspace-Szenarios auf Basis von Computerholographie.

Die umwälzende Veränderung von »Weltanschauung« durch die neuen Medien hat bei Bodo Lensch und seinen Mitstreitern ein Künstlerethos wachgerufen, das den Ansätzen der gestaltenden Reformer der ersten und zweiten industriellen Revolution sehr ähnlich ist: Den Bereich computergesteuerter Umwelterfahrung nicht ganz den Marktmechanismen zu überlassen, sondern zu versuchen, auch für diese Form von Medien ästhetische Prinzipien zu formulieren – das sieht das Bonner Klangkollektiv als wichtige Aufgabe an.

Computer-Unfälle

Klaus Brunnstein

Würmer, Viren, Trojanische Pferde und schlichte Denkfehler sind die Geißeln der EDV-Welt. Klaus Brunnstein, der »Kammerjäger« der Branche, warnt vor den katastrophalen Wirkungen mangelnder und destruktiver Programmiererphantasie.

Berichte über »nicht ordnungsgemäßes Funktionieren der Informationstechnik« häufen sich. Diese Vorfälle, die man korrekter als »Computer-Unfälle« bezeichnet, lassen sich je nach ihrer Ursache unterscheiden. Dabei kann man folgende Fehlerquellen feststellen:

- Denkfehler bei der Konzeption der Systeme,
- Implementationsfehler bei der Realisierung der Systeme,
- unabsichtliche Bedienfehler der Anwender sowie
- absichtliche Bedienfehler oder Mißbrauch.

Denk- und Konzeptfehler

Häufiger Grund für Fehler bei der Konzeption von Programmen sind grundsätzliche Verständnisprobleme, die bei der Übertragung des menschlichen Problemverstehens in computergeeignete Abläufe und der Aufteilung umfangreicher Aufgaben in kleine, überschaubare Einzellösungen entstehen. Manche Konzeptfehler könnten durch eine geeignete »Risikoanalyse« vorher erkannt werden. Solche Analysen durch unabhängige Sachverständige werden aber selten durchgeführt – ob aus wirtschaftlichen oder Geheimhaltungsgründen, sei dahingestellt.

Fehler bei der Realisierung

Zum gesicherten Erfahrungsschatz vieler Anwender von DV-Systemen gehört das Auftreten unerwarteter Fehler, die für Laien zumeist schwer durchschaubar sind. Tatsächlich werden viele Computerfehler durch Schwächen in der Implementierung verursacht, wenn sie nicht bereits auf Konzeptionsfehler zurückgehen. Der Erfahrungsschatz der Anwender, gerne scherzhaft als »Murphy's Gesetz« bezeichnet (Fehler treten grundsätzlich auf, wenn sie technisch möglich sind), läßt sich genauer durch die folgenden »Drei Prinzipien der Datenverarbeitung« beschreiben:

1. *Prinzip*: die »Fehlertracht« von Software:

Jedes hinreichend umfangreiche und komplexe Programm enthält mindestens einen Programmierfehler.

Merkregel: auf 5 bis 10 KByte »guter« Programme kommt etwa ein Fehler; in einem umfangreichen Textverarbeitungssystem auf einem PC sind somit etwa 50 bis 100 unentdeckte Fehler enthalten.

2. *Prinzip* der »Größten Gemeinsamkeit«:

Programmierfehler machen sich zum für die Anwender ungünstigsten Zeitpunkt bemerkbar, und sie bewirken dabei den größtmöglichen Schaden.

3. *Prinzip* vom »teuflischen Regelkreis«:

Wird ein Fehler erkannt, muß er korrigiert werden. Bei der Korrektur entsteht mindestens ein neuer Fehler – dann beginnt alles wieder beim ersten Prinzip.

Bei großen Betriebssystemen gibt es bei einem »Update« oft Hunderte von Korrekturen, oder, wie Informatiker das nennen: »Flicken« (Patches). Nach einer mittleren Lebenszeit (fünf bis zehn Jahre) bestehen Betriebssysteme und andere umfangreiche Systeme (von Anwenderprogrammen ganz zu schweigen) im wesentlichen aus »Flickwerk« – eine vielleicht modische, jedoch kaum belastbare Architektur.

Erfahrungen seit den fünfziger Jahren zeigen, daß Programme mit zunehmendem Umfang fehlerträchtiger werden. Bereits in den sechziger Jahren wurde für Betriebssysteme von Großrechnern nachgewiesen, daß die Fehlerrate nach der Behebung der Anfangsfehler mit späteren Erweiterungen ständig zunimmt.

Benutzungsfehler der Anwender

Die Schäden durch falsche Benutzung von Programmen – vor allem aufgrund der mangelhaften Ausbildung der Anwender – können oft nur schwer abgeschätzt werden, zumal die Zusammenhänge zwischen Unfallergebnis und Unfallursache oft komplex sind und nicht immer hinreichend korrekt berichtet werden. Ein typisches Beispiel dafür ist die Fehlbedienung des Bordcomputers (genauer: des Navigationscomputers) eines koreanischen Jumbos: Das Flugzeug wich wegen einer Fehleingabe der Flugweg-Koordinaten weit vom Kurs ab und geriet in den sowjetischen Luftraum geriet, wo es mit Hilfe computergesteuerter Raketen zielsicher abgeschossen wurde. Die menschliche Plausibilitätskontrolle versagte gleich mehrfach: erstens bei den Piloten, die keine ausreichenden Kontrollen ihres tatsächlichen Flugweges durchführten, zweitens bei der sowjetischen Militärhierarchie, die die computergesteuerten Raketen ohne hinreichende Plausibilitätskontrolle des angeblich feindlichen Verhaltens der eingedrungenen Maschine abschießen ließ.

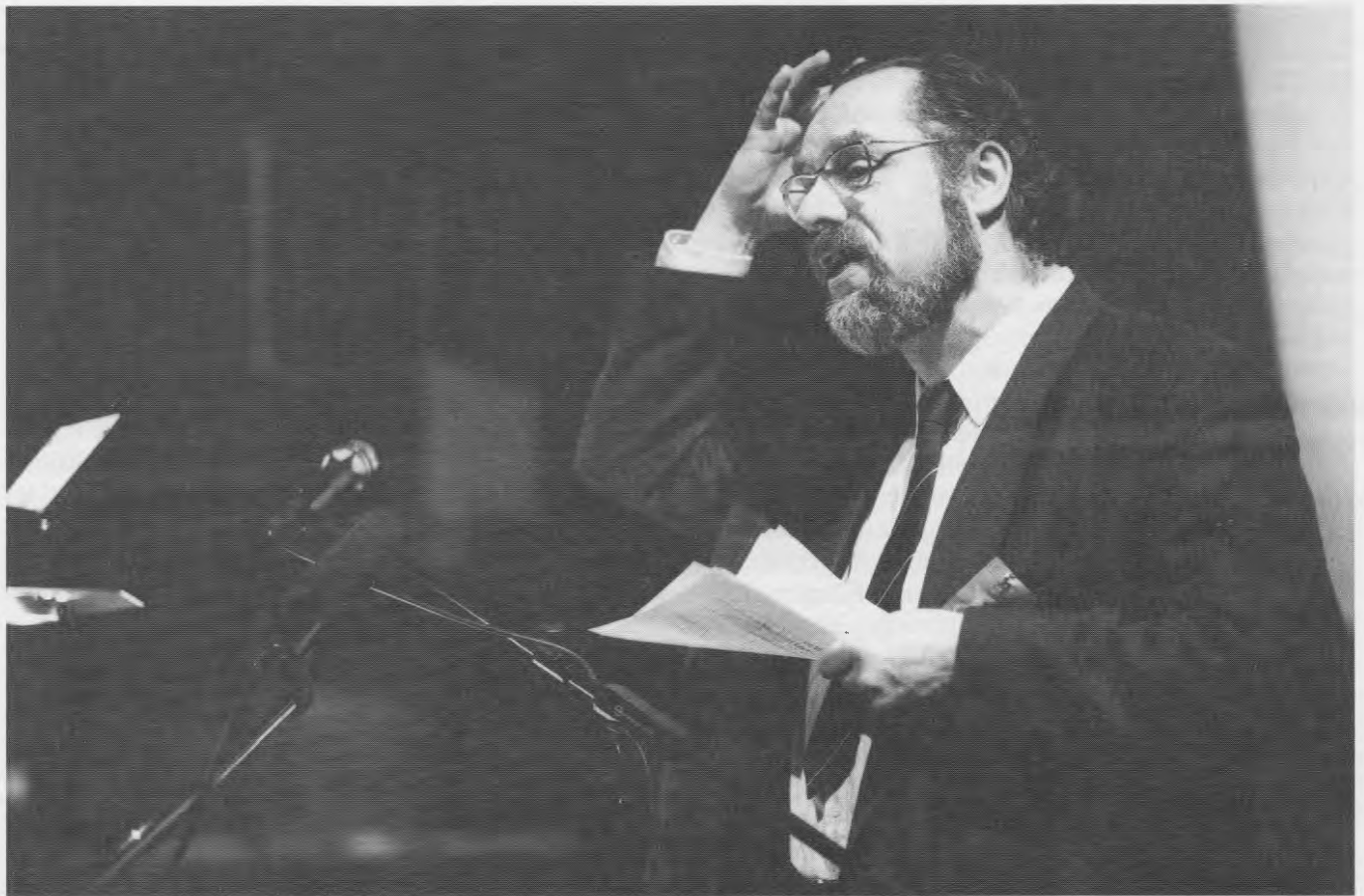
Weniger offensichtlich sind Bedienfehler der Anwender, die zur Einschleusung von Computer-»Viechereien« wie Computer-»Viren«, Computer-»Würmern« oder »Trojanischen Pferden« führen. Dabei werden die Anwender zu Erfüllungsgehilfen von Leuten, die bewußt die Konzeptschwächen der Informationstechniken (besonders der Betriebssysteme, Datenbanken und Netze) ausnutzen, um Experimente mit unabsehbaren Folgen durchzuführen.

Mißbräuchliche Computernutzung

Hier sind zwei Aspekte zu unterscheiden:

- Computer können als Hilfsmittel bei Planung und Durchführung krimineller Handlungen benutzt werden;
- Computer können auch Objekt einer kriminellen Handlung sein.

Seit etwa Mitte der achtziger Jahre werden Programm-»Anomalien« immer mehr zu kriminellen



Zwecken benutzt. Dazu gehören gezielte Störungen kommerzieller Programme und Daten ebenso wie die Verbreitung des »Trojanischen AIDS-Programmes« oder des politisch motivierten »Worm Against Nuclear Killers«.

Seit den siebziger Jahren gelang es Außenstehenden, oft als »Hacker« bezeichnet, mehrfach, in fremde Computer-Systeme einzudringen. Während anfänglich einige jüngere Leute, manchmal unterstützt von Datenschützern, durch ihre Hackerangriffe auf die Unsicherheit von Systemen hinweisen wollten (etwa bei der Demonstration des unsicheren Btx-Systems zu Lasten der Hamburger Sparkasse oder beim Eindringen in das Wissenschaftsnetz SPANet alias »NASA-Hack«), wurden gleichzeitig auch kriminelle Angriffe durchgeführt – wie der »KGB-Hack«, bei dem Disketten mit Daten und Programmen an den KGB verkauft worden sein sollen.

Beispiele von Computer-Unfällen

Im folgenden werden einige Beispiele für Computer-Unfälle dargestellt, die in Ursache, Verlauf und Wirkung hinreichend aufgeklärt wurden.

Zur Kategorie der Denk- und Konzeptfehler gehören der Abschluß eines Iran-Airbusses durch das militärische, »künstlich intelligente« AEGIS-System (270 Tote) sowie der Verlust mehrstelliger Milliardenwerte bei den Börsencrashes 1987 und 1989 als Folge der automatischen Baisse-Spekulation durch sogenannten »Programm-Handel«. Für Realisierungsfehler gibt es vielfache Beispiele; sie haben am Himmel (durch

den Verlust von Raketen, Satelliten und Sonden) wie auf Erden zu vielfältigen Verlusten und Schäden geführt. Hier wird das Beispiel des Therapie-Computers THERAC-25 erläutert, bei dem ein Programmierfehler eines computergesteuerten Bestrahlungsgerätes mindestens zwei Menschenleben forderte.

Aus der Kategorie der Fehler bei der Benutzung sollen hier einige sogenannte »Programm-Anomalien« behandelt werden. Selbst über dieses »Informatik-Bestiarium« (alias: »Computer-Zoo«) kann nur in einigen Facetten berichtet werden, hier mit einer kurzen Analyse von »Viren«, »Würmern« und »Trojanischen Pferden«.

Programmierter Aktienhandel und Börsenkräche

Ein in seinen finanziellen Wirkungen augenfälliges Beispiel für einen unvorhergesehenen Netz-Unfall liefert die vielfach erst nachträglich angestellte Risikoanalyse des Börsenkrachs vom 19. Oktober 1987. So sorgfältig die Risiken der weltweiten Vernetzung kreditwirtschaftlicher Institute zum Austausch von Geld-Information (alias »Überweisungen«) auch analysiert wurden (etwa beim inzwischen befriedigend funktionierenden Interbank-Kommunikationssystem SWIFT), so wenig wurden spezifische Sicherungsmechanismen für die Vernetzung der Börsenkurs-Informationen vorbedacht. Auch blieben Warnsignale unbeachtet, so als im Frühjahr 1987 eine gezielte Fehleingabe kurzfristig zur Manipulation des Aktienkurses einer britischen Firma führte.

Bei der Vernetzung der Börsencomputer wurde davon ausgegangen, daß die über die Leitungen versandte Information zeitlich, inhaltlich und in ihren Zusammenhängen stets korrekt sei. Auf dieser Annahme bauen die Börsenhandelsprogramme auf: Weil rund um die Uhr an irgendeinem Börsenplatz der Welt ein Börsenhandel mit Kauf- und Verkauforders getrieben wird, müssen an Orten, wo gerade Nacht oder Feiertag ist, automatische Systeme die aktuelle Kursentwicklung »überwachen« (Monitoring), um bei großen Aktienpaketen den »Verlust zu minimieren« oder »Gewinne mitzunehmen«. Die Börsenprogramme der Bankinstitute haben meistens fest eingebaute Schwellwerte, bei deren Unterschreitung ein entsprechendes Wertpapierpaket elektronisch zum Verkauf angeboten wird (»stop loss mechanism«).

Auch auf dem elektronisch organisierten Börsenmarkt herrschen die »Gesetze von Angebot und Nachfrage«, wie sich am 19. Oktober 1987 allzu deutlich zeigte: Als an jenem Montagmorgen des 16. Oktobers die Kurse einiger bedeutender Aktienpakete an der Tokioter Börse sanken, mag zunächst nur der Schwellwert eines Börsenprogrammes unterschritten worden sein. Dessen elektronisches Verkaufsangebot mußte bei schwachem Markt und mangelndem Interesse eine Verkaufslawine auslösen, da nun auch die Schwellwerte anderer Börsenprogramme unterschritten wurden. Stabilisierende Elemente, etwa das Eingreifen einer »elektronischen Börsenaufsicht«, waren in dem System nicht vorgesehen, und die verantwortlichen Aufseher der großen westliche Banken und Börsen schlofen, da in Frankfurt und New York Nacht war. Der wirtschaftliche Schaden der »eingebauten Denkweisen« war schockierend: Weltweit wurden binnen weniger Tage Aktienwerte in der Größenordnung von Billionen Dollar verloren.

Das KI-System AEGIS und der Iran-Airbus-Abschuß

Im Juli 1988 erfuhr die schockierte Öffentlichkeit, daß der Raketenkreuzer »USS VINCENNES« im Rahmen militärischer Aktionen der USA einen Airbus der Iran Air mit 270 Menschen an Bord in der Straße von Hormuz abgeschossen hatte. Das Thema geriet allerdings schnell wieder aus den Schlagzeilen; auch die 1989 veröffentlichte sorgfältige Analyse eines Untersuchungsausschusses des US-Kongresses wurde hierzulande kaum zur Kenntnis genommen. Diese Untersuchung, die Fehler bei der Nutzung eines der fortgeschrittensten Signalverarbeitungs- und entscheidungsunterstützenden Systeme analysierte, wäre allerdings geeignet, die hierzulande verbreitete Informationseuphorie zu dämpfen.

Ein Kernstück des AEGIS-Systems ist ein modernes, hochauflösendes Radarsystem, dessen mehrere Tausend aktive Elemente alle bewegten Objekte bis in mehr als 300 Kilometer Entfernung erfassen können. Jedes einmal erfaßte Objekt wird in einem Computersystem gespeichert, wobei auch die auf anderen Informationskanälen dazu ermittelten Funksignale

oder Infrarotsignale diesem Objekt zugeordnet werden. Überdies wird die Bewegung verfolgt und der weitere Weg des Objektes berechnet.

Mit diesem System kann also vom Raketenkreuzer aus ein großer, halbkugelförmiger Raum oberhalb des Radar- und Sichthorizontes überwacht werden. Vom zentralen Standort können allerdings Objekte, die aus Sicht des AEGIS-Systems hintereinander liegen, grundsätzlich nicht unterschieden werden. Daher könnten die Funkdaten eines verdeckten Flugobjektes fälschlich dem radarsichtbaren vorderen Objekt zugeordnet werden. Um dies zu vermeiden und ein räumlich korrektes Bewegungsbild herzustellen, können die AEGIS-Systeme von zwei Kreuzern miteinander verbunden werden. Auch die Computersysteme eines Frühwarn-Flugzeuges (AWACS) können als zweites AEGIS-Auge dienen.

An jenem Julitag 1988 hatte es einen Raketenangriff von mehreren Schnellbooten iranischer Terroristen auf die Vincennes gegeben. In dem zentralen Gefechtsstand, einem fensterlosen großen Raum voll Elektronik und Bildschirmen, herrschte noch Hektik, als auf einem Bildschirm ein Flugzeug erschien, das auf dem nahen Flughafen Bandar-e-Abbas gestartet war und nach einem Steigflug scheinbar auf das Schiff zuflog. Bei dem Symbol dieses Flugzeuges wurde noch angezeigt, daß es sowohl zivile wie militärische Kennsignale auf dem sogenannten Transponder sendet, einem Funksystem, mit dem sich Flugzeuge als »zivile« (dann mit Informationen über Flugnummer, Flughöhe und so weiter) oder »militärisch« (dann mit einer »Freund-Feind-Kennung«) identifizieren.

Der Kommandant bewertete das widersprüchliche Transpondersignal als Hinweis auf ein feindliches Flugzeug. Zu diesem Zeitpunkt hatte ein anderes Schiff der AEGIS-Klasse (wie vermutlich auch ein in der Luft befindliches AWACS-Flugzeug) korrekt erkannt, daß es sich um ein ziviles Flugzeug beim Steigflug handelte, das mit weniger als fünf Minuten Verspätung von Bandar-e-Abbas nach Abu Dhabi gestartet war. (Die Flugzeiten der naheliegenden Flughäfen waren übrigens in dem »künstlich intelligenten« AEGIS-System nicht gespeichert.) Allerdings waren die AEGIS-Computer beider Schiffe zu diesem Zeitpunkt nicht gekoppelt, so daß die Vincennes nur auf einem »Auge« sah. Während ein Offizier wegen der wider-



sprüchlichen Angabe noch Zweifel an der Korrektheit der angezeigten Daten äußerte, befürchtete der Kommandant einen Angriff auf sein Schiff und gab den Befehl zum Abschluß. Erst nach dem Start der Abwehraketen und zu spät für korrigierende Handlungen veränderte der Computer die Daten über Flughöhe und Flugtendenz (jetzt: »weiter steigend«).

Für den objektiv unbegründeten Abschluß hat die hektische Atmosphäre vermutlich eine erhebliche Rolle gespielt. Wenige Tage zuvor war überdies ein Kommandant abgelöst worden, nachdem sein Schiff (USS STARK) von einer iranischen EXOCET-Rakete getroffen wurde, die vom bordeigenen Computersystem nicht als feindlich erkannt wurde. Man muß dem Kommandanten überdies zugute halten, daß er nicht einmal die Chance hatte, mit einem Fernglas die Flugmaschine zu identifizieren: Der Kontrollraum hat keine Fenster. So war die Welt auf das reduziert, was die Computer erfassen und darstellten.

Der Kern des Versagens liegt also hauptsächlich bei den Informatikern, die dem Kommandanten nicht genügend unabhängige Kontrollmöglichkeiten gaben. Mindestens hätte das System bei widersprüchlicher Information (der Radarquerschnitt war zudem typisch für den Airbus, für eine Militärmaschine dieser Entfernung viel zu groß) den Hinweis geben müssen: »Keine Entscheidung möglich, da Information unzureichend.« Und schließlich müßte ein Abschluß bis zum allerletzten Augenblick blockierbar sein. All dies war aber im System nicht vorgesehen.

Der Therapie-Computer »THERAC-25«

Beim Einsatz des (radio-therapeutischen) Bestrahlungsgeräts »THERAC 25« (THERapie Computer mit 25 MeV) in US-Kliniken hat ein Softwarefehler nachweislich zu zwei Todesfällen geführt. Mit diesem Gerät werden Krebskranke entweder mit Röntgenstrahlen (»X-Modus«) oder »Elektronenstrahlen« (»E-Modus«) bestrahlt. Kontrolliert von einer PDP 11, wird im X-Modus ein energiereicher Elektronenstrahl (mit 25 Millionen Elektronen-Volt (MeV)) auf eine kleine Wolframscheibe »geschossen«, wodurch »weichere« Röntgenstrahlung erzeugt wird, mit der bestimmte Karzinome bestrahlt werden. Andere Karzinome werden dagegen direkt mit Elektronen bestrahlt. Dabei muß der Strahl allerdings erheblich energieärmer sein (etwa 2 MeV), wozu die Intensität der Elektronenkanone computergesteuert heruntergefahren und die Wolframscheibe aus dem Strahl entfernt werden muß. Das System darf also nur zwei »Zustände« zulassen: Entweder trifft ein »heißer« E-Strahl (25 MeV) auf eine Wolframscheibe, dessen Röntgenstrahlung auf den Körper gerichtet wird, oder ein deutlich energieärmerer Strahl (2 MeV) trifft direkt auf den Körper. Einen dritten Zustand darf es nicht geben; leider wurde dies bei der Programmierung und Test des Systems nicht zuverlässig garantiert. Als ein Patient berichtete, bei Einschalten der Apparatur sei etwas »wie ein heißer Stahl« durch seinen Körper gegangen, erklärten Ärzte und Herstellerfirma das für »unmöglich«.

Wenige Monate später war der Patient tot. Erst nach weiteren Berichten und dem Tod eines weiteren Patienten stellte sich heraus, daß unter bestimmten Bedingungen ein »verbotener Zustand« mit energiereichem Elektronenstrahl (25 MeV) ohne absorbierendes Wolfram-Stück eintreten konnte. Die Bestrahlungsparameter, also X- oder E-Modus, Dauer, Weg der Bestrahlungsquelle über dem Körper, können bei THERAC 25 für jeden Patienten über ein Bildschirm-Formular (»Eingabe-Menue«) eingegeben, unter dem Namen des Patienten gespeichert und für weitere Bestrahlungen abgerufen werden. Ist bei einer neuen Behandlung eine Veränderung etwa der Bestrahlungsdauer erforderlich, so können die gespeicherten Daten auf dem Bildschirm angezeigt, der Zeiger (durch »Arrow-UP«-Taste) an die zu verändernde Stelle bewegt und die Veränderung mit der »Bestätigungstaste« quittiert werden.

Jedesmal, wenn ein Arzt mit der »Arrow-Up«-Taste einige Steuerparameter im Eingabe-Menue änderte, ergab sich der verbotene Zustand. Auf dem Bildschirm des Arztes wurde sogar eine Fehlermeldung (»Malfunction 54«) angezeigt, als die Patienten direkt mit »heißen« Elektronen beschossen wurden. Allerdings war der Fehler in den Unterlagen nicht beschrieben, und bei seinem Auftreten war ohnehin der Schaden, der unbedingt hätte verhindert werden müssen, bereits eingetreten.

Als der kanadische Hersteller den Fehler nicht mehr abstreiten konnte, teilte er folgendes Verfahren zur Korrektur seines Programmfehlers mit: Man möge die »Pfeil-Tasten« nicht mehr benutzen; sicherheits halber solle man diese Tasten aus der Tastatur entfernen, die Löcher möge man zukleben.

Das Informatik-»Bestiarium«

Neben Konzept- und Programmier-Fehlern können Fehlfunktionen auch nachträglich in Programme/Systeme eingebaut werden. So kann ein verbreitetes Programm (zur Textbearbeitung zum Beispiel) als »Trojanisches Pferd« hergerichtet werden, indem ein Zusatzprogramm hinzugefügt wird, das unerkannt weitere Handlungen durchführt. Bei einem »Computer-Virus« wird ein Programmstück in das Betriebssystem oder in Anwendungsprogramme eingefügt, das zusätzliche ungewollte Handlungen bewirkt – angefangen bei Bildschirmspielereien (»Buchstaben fallen herunter wie Blätter im Herbst«) bis hin zur Zerstörung von Programmen und Daten. Bei einem »Computer-Wurm« können solche Programmstücke in Programme der vernetzten Arbeitsplatzrechnern eingepflanzt werden.

Computer-»Viren«

Computer-Viren werden von Presse und DV-Laien gern für alle möglichen, ihnen unerklärlichen Vorfälle verantwortlich gemacht. Fälschlich werden auch andere Programm-Anomalien, wie der INTERNET-Wurm oder das »Trojanische AIDS-Programm«, als Computer-Viren bezeichnet.

Erstmals von Fred Cohen 1982/83 in seiner Dissertation untersucht, sind »Programm-Viren« zusätzliche Programmstücke, die über ein »Wirtsprogramm« vornehmlich über Disketten in ein Computersystem eingeschleppt werden und die Fähigkeit haben, sich auf andere Programme auszubreiten. Daneben kennt man seit einiger Zeit die »System-Viren«, die sich über Teile des Betriebssystems, vor allem die beim Starten vieler Personal-Computer benutzten Systemtabellen (Bootsektoren und andere) in einen Computer einschleusen und ausbreiten. Beide Virenarten üben zusätzlich eine gezielte »Wirkung« aus, die bei »Spiel- und Spaß-Viren« eher als Schabernack, bei »Destruktiv-Viren« allerdings als krimineller Anschlag zu bewerten ist. Solche Wirkungen können bei einem vorgegebenen Ereignis eintreten, etwa zu einem festen Datum oder wenn der Virus eine vorgegebene Ausbreitungsschwelle (intern dargestellt durch einen »Generationszähler«) erreicht hat. Bei den »Spiel- und Spaß-Viren« treten für den Anwender überraschende Erscheinungen auf, etwa indem ein Punkt oder ein Ball über den Bildschirm hüpfen (»Italienischer Virus«) oder indem in den Herbstmonaten Oktober bis Dezember die Buchstaben statistisch ausgewählt vom Bildschirm fallen wie die Blätter im Herbst (»Herbstlaub-Virus«). Oder es werden plötzlich Melodien gespielt (»Oropax« und »Yankee-Doodle-Virus«). Solche Viren verunsichern die betroffenen Anwender, richten aber keine direkten Schäden an Programmen und Daten und Betriebssystem an.

Bei den »Destruktiv-Viren«, die gezielt Schäden auslösen, werden dagegen Programme gelöscht. Beim »Israeli-Virus« wird jedes an einem »Freitag, den 13.« gestartete Programm gelöscht, bei den »Datacrime I-Viren« Programme ab 13. Oktober 1989. In anderen Fällen werden ganze Datenträger gelöscht, wie beim »Century A-Virus«, bei dem am 1. Januar 2000 alle Platten und eingelegte Disketten mit Nullen beschrieben werden, wonach die Meldung erfolgt: »Willkommen im 21. Jahrhundert«. Bei manchen Viren wird »bloß« die Tastatur blockiert (bei den »April 1-Viren« am 1. April). Besonders bössartig ist der dBase-Virus. Sein wirtschaftlicher Schaden ist gravierend, denn dBase ist ein Datenbank-Programm, das oft, nicht bloß bei kleineren Unternehmen, für Finanzrechnungen und Lagerhaltung verwendet wird. Dieser Virus vertauscht nach statistischer Wahl je zwei benachbarte Zeichen in typischen dBase-Dateien (DBF-Files), wobei vermittelt einer verborgenen Datei die Daten scheinbar korrekt auf dem Bildschirm angezeigt werden. Nach neunzig Tagen wird jede verseuchte DBF-Datei und mit ihr die verborgene Datei gelöscht. In den zur Sicherung gespeicherten Dateien sind die Zeichen natürlich auch vertauscht.

In einzelnen Fällen richten sich Viren gegen Programme eines bestimmten Herstellers, so beim »Syslock/Macho-Virus«, bei denen im Copyright-Text der Name des Software-Hauses Microsoft durch »Macrosoft« beziehungsweise »Machosoft« (wohl der erste feministische Virus) ersetzt wird, oder schlimmer

beim »Scores-Virus« auf Macintosh-Rechnern, bei dem speziell die Programme des US-Softwarehauses EDS nach drei Tagen »zerbommt« werden.

Inzwischen sind über 150 Computer-Viren allein auf den verbreiteten Personal-Computern bekannt, davon über 80 auf IBM- und kompatiblen PCs. Hinzu kommen Computer-Viren auf mittleren und großen Rechnersystemen, die teilweise von Sicherheitsexperten zu Testzwecken erstellt wurden. Besonders gefährlich könnten sich Defizite bei Unix-Systemen auswirken, auf dem Computer-Viren besonders einfach programmiert und Unix-Netze verbreitet werden können.

Bei der Entwicklung sind inzwischen ganze »Familien« festzustellen, bei denen anfänglich einfache, leicht erkennbare und relativ harmlose Viren in der Folge so verändert werden, daß sie schwerer erkennbar sind und zugleich schwerere Schäden bewirken. Zu dieser Entwicklung hat die Verbreitung von Viren-Programmen und die Neugier vieler Lernbegieriger beigetragen, vor allem aber auch die Veröffentlichung von Virusprogrammen. Nach Ansicht der Internationalen Informatik-Organisation (IFIP) stellt die Weitergabe und Veröffentlichung von Computer-Viren einen schweren Angriff auf die öffentlich, wirtschaftlich und privat genutzte Informationstechnik dar und sollte unter Strafe gestellt werden. In der Bundesrepublik könnte dies bereits als Beihilfe zur Computersabotage (§303 b StGB) angesehen werden.

»Trojanische Pferde« und Hacker-Angriffe

Seit Mitte der achtziger Jahre werden, wie zuvor schon in Forschung und bei ausgewählten Bereichen von Staat und Wirtschaft, immer mehr zuvor alleinstehende kleine und große Rechner durch Netze verbunden. Die Gefährdung solcher vernetzter und verteilter Systeme wird durch zahlreiche Hacker-Einbrüche in Rechensysteme von Wissenschaft, Forschung sowie zunehmend der Wirtschaft deutlich.

Bei solchen Angriffen benutzen Eindringlinge – als »Hacker« ohne kriminelle Absicht, als »Cracker« mit kriminellen Zielen – vor allem sogenannte »Trojanische Pferde«. Darunter versteht man Programme, die zusätzlich zu ihren bekannten Funktionen unerkannt weitere Dienste leisten können. Bei DV-Freaks in Schule und Hochschule ist das Standardprogramm zur Eingangskontrolle (»Einloggen«) in einen größeren Rechner (zum Beispiel »Logout«) beliebt, das die Benutzerangaben – sogenannte Benutzer-Identifikation und Paßwort – abfragt und mit den gespeicherten Erlaubnissen vergleicht. Mit einfachen Tricks kann ein neues Login geschrieben werden, das die so erlangten ID/Paßwort-Paare erfragt und an einer getrennten Stelle abspeichert, wodurch der Programmautor in den Besitz (zumeist) legaler Zugangsberechtigung zu solchen Rechnern kommt.

Beim sogenannten »NASA-Hack« gelang deutschen Hackern das Eindringen in über 130 Rechner des SPANet (Space Physics Analysis Net), das Forschungsinstitute der Raumfahrt unter Führung der NASA mit deren VAX-Rechnern Castor und Pollux



verbindet. Als Einbruchswerkzeug diente ein »Trojanisches Pferd« in der Loginout-Prozedur der VAX-Rechner von Digital Equipment, das durch einen schweren Systemfehler des Herstellers installiert werden konnte. Bei der Version 4.4 des VAX-Betriebssystems VMS konnte ein Benutzer mit einfachen Tricks unberechtigt eine Zugangsberechtigung erlangen. Dieser Fehler, der spätestens Anfang 1987 bekannt war, ist von der DEC-Organisation so schleppend behoben worden, daß VAX-Benutzer im Frühjahr durch Beiträge in der elektronischen Zeitung Vaxinfo selbst Systemkorrekturen diskutierten. Auf diesem Wege haben Hacker – unter Mithilfe mindestens eines Informatikers einer renommierten deutschen Fakultät – von der »Systemschwäche« erfahren und dann über ein Heidelberger Rechenzentrum den Einstieg in die NASA- und andere Rechner gestartet. Erst ab Mai 1987 hat DEC alle VAX-Benutzer mit zusätzlichen Schutz-Informationen versorgt.

Allerdings haben zahlreiche Anwender ihre VAX-Rechner einschließlich des Betriebssystems »aus zweiter Hand« erworben. Sie haben deshalb selten ein Lizenzabkommen mit DEC und erhalten daher auch keine Korrekturen oder System-Erweiterungen. Es ist davon auszugehen, daß die fehlerhafte Version des VMS-Betriebssystems noch heute vielfach im Einsatz ist. Der Vorfall könnte sich also wiederholen, zumal nach Bekanntwerden des Netzeinbruches nicht alle »Trojanischen Pferde« eliminiert wurden, wie ein Nach-Hack, im Sommer 1988 in dem ARD-Magazin Panorama vorgeführt, bewies.

Der Computerhersteller DEC hat aus diesen Fehlern immerhin gelernt und verfolgt seit Januar 1988 eine neue »Philosophie«, wonach Software-Änderungen schnellstmöglich und geräuschlos (selbst an »second-hand«-Kunden) verteilt werden. Inzwischen sind sogar Sicherheitskorrekturen verteilt worden, bevor irgendein Benutzer überhaupt ein »Sicherheitsloch« bemerkt hatte. Inzwischen wurden auch verbesserte Sicherheitsmechanismen entwickelt; leider setzen allzu viele Rechenzentren und Anwender diese Schutzverfahren aber nicht ein.

Bei einem besonders spektakulären Einbruch gelang es einer Gruppe von Crackern aus Berlin und Hannover, sich über den Zugang zu zahlreichen militärischen und wirtschaftlichen Rechnern so reich-

haltige elektronische »Information« zu beschaffen, daß der sowjetische Geheimdienst KGB ihnen mehrfach größere Beträge bezahlte (angeblich für neun Disketten jeweils 30 000 Mark). Bei diesem Angriff wurden offenbar sowohl DEC/VMS-Rechner wie auch Unix-Rechner verschiedener Hersteller, in Einzelfällen auch einzelne Großrechner »besucht«. In verschiedenen Veröffentlichungen werden vor allem Rechner der US-Luft-, Land- und See-Streitkräfte in Japan, Europa sowie Mittel- und Nordamerika sowie die Personaldatenbank Optimis im US-Verteidigungsministerium erwähnt. Durch Spielmaterial über ein angebliches »SDI-Netzprojekt« wurde endlich auch die KGB-Verbindung offensichtlich. Neben den US-Militärrechnern »besuchten« die Hacker auch Forschungsrechner in Hochschulen und Wirtschaft. Dort sollen sie auch Betriebssysteme, Sprachübersetzer und spezielle Anwendungsprogramme, unter anderem zum Entwurf von Chips einschließlich spezieller Entwürfe und Daten gesammelt haben. Im Frühjahr 1989 wies das BKA mehrere Firmen auf derartige Einbrüche hin und bat um konkrete Mitteilung vorliegender Erkenntnisse; in den betreffenden Rechenzentren dürften dazu aber kaum Erkenntnisse vorliegen, da lediglich Daten für Betriebsstatistiken aufgezeichnet und ausgewertet werden.

Der sogenannte »KGB-Hack«, durch die Aufmerksamkeit des Astronomen und Hobby-Informatikers Clifford Stoll bereits 1986/87 erkannt und im Mai 1988 veröffentlicht (»Stalking the Wily Hacker«), konnte durch die deutschen Sicherheitsbehörden erst durch zwei Kronzeugen-Aussagen aus dem Täterkreis aufgeklärt werden. Im Verlauf des Verfahrens wurde nach einer Anzeige der Universität Bremen, die im Dezember 1987 rund 30 000 Mark an Telefonkosten der Hacker bezahlen mußte, sogar eine Telefonverbindung festgestellt, von der gerade ein Hackerangriff erfolgte. Weil aber eine richterliche Erlaubnis zum Anzapfen der Leitung nach Grundgesetz-Artikel 10 nicht eingeholt worden war, mußte der Untersuchungsrichter ein Beweisverwertungsverbot erlassen und die Löschung der erfaßten Daten veranlassen.

Während »Trojanische Pferde« lange zum direkten Arbeiten in Netzen eingesetzt wurden, tauchte im Dezember 1989 ein neuartiges »Trojanisches Pferd« auf, das über mehr als 10 000 Disketten weltweit, vor allem in Europa, bei Immunologen und Ärzten sowie Rechenzentrumsleitern von Banken, Sparkassen und Beratungsunternehmen verteilt wurde. Auf der zugesandten Diskette befand sich ein Programm zur Berechnung des eigenen Aids-Risikos (»AIDS Information Version 2.0«), wobei Fragen zu Krankheiten und zum Lebenswandel gestellt wurden. Das Programm war laut Aufkleber mit einem Installationsprogramm auf der Platte zu installieren. Dieser Anordnung folgten nicht nur Ärzte und Biologen, sondern auch Rechenzentrumsleiter und DV-Laien.

Wie sich bald herausstellte, wurde bei Befolgen der Startprozedur ein »Trojanisches Pferd« (»REMx.EXE«, wo x ein nicht druckbares Zeichen ist) installiert, das die Zahlung einer Lizenzgebühr (186

US-Dollar bei 365-maliger Nutzung) verlangte. Nach neunzigmaliger Nutzung wurden die Programme selbst sowie ihre Einträge in der Bibliothek verschlüsselt, und erst durch ein »Heilprogramm« sollte man die Daten und Programme wieder benutzen können.

Nach unterschiedlichen Schätzungen sind 50 000 bis 150 000 Dollar für die Beschaffung der Adreßlisten – darunter Teilnehmer einer Aids-Konferenz in Stockholm und die britischen Bezieher der PC Business Week, vermutlich auch ältere deutsche Verteiler – sowie für Herstellung und Versand der Disketten aufgewendet worden. Nachdem Fachleute auf der ganzen Welt bereits nach wenigen Tagen mehrere Gegenmittel (Programme wie Clearaids oder Aidsout) erstellt hatten, ist es unwahrscheinlich, daß die Verbreiter des »Trojanischen Aids-Programmes« auf ihre Kosten kommen. Ein größerer Schaden ist nicht eingetreten. Jedoch war wieder zu beobachten, wie leichtfertig Laien und manche DV-Experten eine unverlangt zugesandte Diskette einfach in ihre DV-Maschine legen und die geforderte Startprozedur ausführen. Dabei lag der Diskette ein – wenn auch sehr eng bedruckter und schwer lesbarer – Zettel mit deutlichem Hinweis bei: »These program mechanisms will adversely affect other program applications on microcomputers.«

Computer-»Würmer«

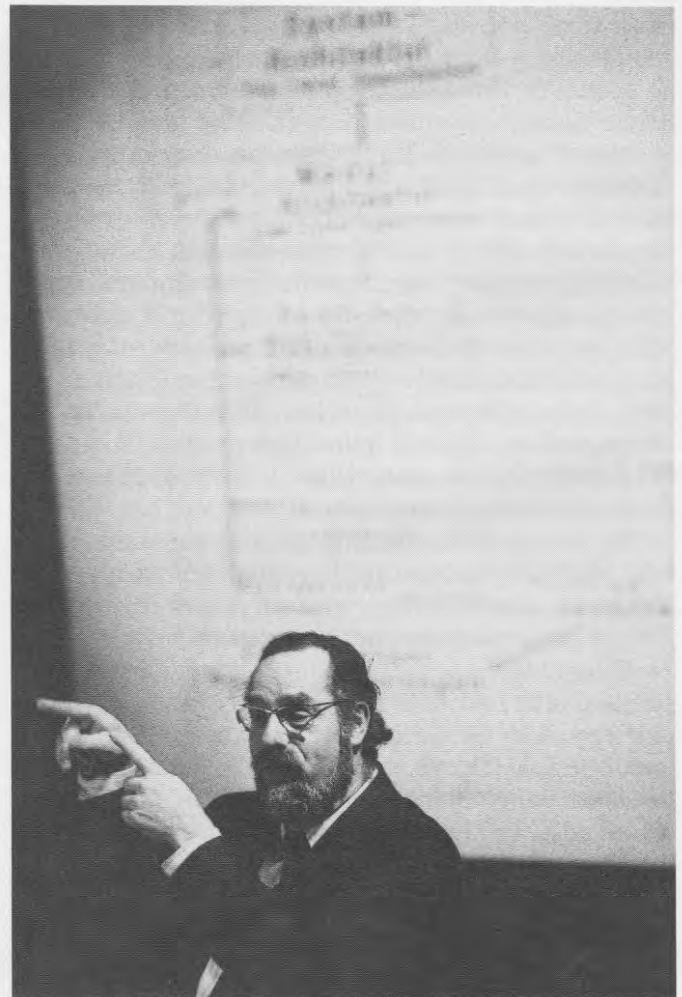
Die automatische, verdeckte Ausbreitung von Programmen in Computernetzen ist erstmals 1982 in Kalifornien von den US-Informatikern Shoch und Hupp untersucht worden. Damit realisierten sie eine Idee, die John Brunner in seiner Science-Fiction-Geschichte »Der Schockwellenreiter« schon Mitte der siebziger Jahre beschrieb: Danach können sich spezielle Programme, sogenannte »Computer-Würmer«, auf alle in einem Netz verbundenen Computer ausbreiten und beliebige Wirkungen programmiert werden. Shoch und Hupp beschrieben auch nützliche Wirkungen dieser »Wurm«-Techniken, indem zum Beispiel Fehler entfernt aufgestellter Computer einem speziell für Fehlerkorrekturen ausgestatteten Computer gemeldet werden, doch sie werden heute nur an wenigen Stellen eingesetzt. Inzwischen haben sich die Risiken dieser Technologie deutlich gezeigt.

Im Dezember 1987 mißbrauchte ein Student der Universität Clausthal-Zellerfeld die elektronische Postverteilung von IBM-Großrechnern in Wirtschaft und Hochschulen für ein »Weihnachtsbaum-Programm«, das er (CHRISTMAS-EXEC) über elektronische Post versandte. Der Empfänger dieses Programms wurde gebeten, das Programm zu starten. Während dann auf dem Bildschirm ein Weihnachtsbaum mit entsprechenden Wünschen gezeichnet wurde, durchsuchte das Programm im Hintergrund die elektronische Adreßdatei des Opfers und versendete sich an alle erreichbaren Adressen und Adreßlisten. Das Programm richtete selbst keinen Schaden an; da dieser Verteilprozeß aber lawinenartig anschwell, hat es die betroffenen Rechner weltweit »vollgemüllt«, und zahlreiche Rechner mußten vom

Netz genommen werden, bis der Müll aus den elektronischen Briefkästen entfernt war. Dieser »Studenten-Ulk« wurde schnell beherrscht, weil Fachleute in den betroffenen Großrechenzentren sofort das Problem erkannten und Gegenmaßnahmen trafen.

Ein Jahr später, im November 1988, waren vor allem mittlere Rechner und Arbeitsplatzrechner von Forschungslabors und Universitäten betroffen. Insgesamt mußten weit mehr als 1000 Systeme vom Netz getrennt und »desinfiziert« werden; einige dieser Rechner konnten erst nach Wochen wieder angeschlossen werden.

Am Mittwoch, den 2. November 1988, stellten Wissenschaftler der Harvard Universität am frühen Abend fest, daß auf einem ihrer (mittleren) Rechner »nichts mehr lief«: Aus unbekanntem Grund war die Anlage in kurzer Zeit mit Programmen vollgestopft worden, und immer neue Programmstarts ließen die Systemtabellen überfließen, bis schließlich kein normales Arbeiten mehr möglich war. Als erste Hypothese dachte man an einen »Virus«, und man begann, nach Ursache und Gegenmaßnahmen zu suchen. Dabei stellte man bereits frühzeitig fest, daß nur Rechner mit Unix-Betriebssystemen betroffen waren, und zwar nur solche mit der Variante »Berkeley Software Distribution (BSD) Version 4.2/4.3«, die auf VAX-Rechnern (Ultronix, Eunice) sowie auf zwei Versionen von SUN-Arbeitsplatzrechnern (SUN-OS) eingesetzt werden. Bald zeigte sich, daß auch Rechner



anderer Hochschulen und Forschungsinstitute betroffen waren, die über das Wissenschaftsnetz Internet mit den Harvard-Rechnern verbunden waren. Gegen 18 Uhr hatte der angebliche »Virus« Pittsburg erreicht, gegen 23 Uhr kam er in den Universitäten von Princeton und Carnegie Mellon an, während Purdue erst am Donnerstag früh (4 Uhr Westküstenzeit) »befallen« wurde. Universitäten wie das Massachusetts Institute of Technology (MIT) in Boston und kalifornische Universitäten in Berkeley, San Diego und Stanford waren ebenso betroffen wie Forschungslabors bei SRI/Stanford, das Lawrence Livermore Labor sowie das MASA Ames Forschungszentrum in Mountain View/Kalifornien.

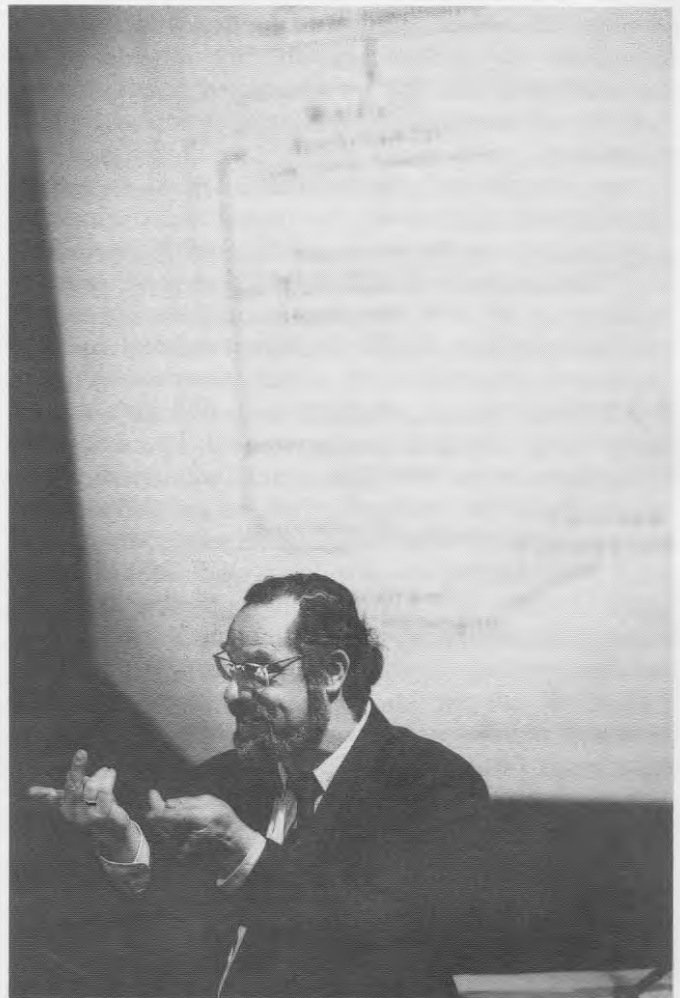
Bei Anbruch des 3. November dürften Installationen in mindestens fünfzig durchweg »größeren« Institutionen mit mehreren hundert Rechnern (jedenfalls nur einem Bruchteil der Rechner am Arpanet) betroffen gewesen sein. »Der Virus« verschonte überall die Cray- und IBM-Rechner, die VAXen mit dem Standard-Betriebssystem VMS wie übrigens auch Rechner mit Standard-Unix-V-Systemen.

Die Ausbreitung erfolgte über die Netze Internet, Arpanet sowie dessen Abkömmling Milnet (eingesetzt für vertraulichere Kommunikation militärnaher Forschung, offenbar jedoch nicht verknüpft mit den zentralen »strategischen Computersystemen«), an denen alle betroffenen Installationen hängen. Von dem Verbrauch an Computerleistung und Arbeitszeit der Rechner-Manager (die sämtlich eine harte Nachtschicht vor sich hatten) abgesehen, wurden keine Schäden an Programmen und Daten festgestellt. Insbesondere ist kein einziges Programm verändert oder sonstwie berührt worden (wie dies bei einem »Virus« unweigerlich erfolgt wäre). Wenn man die Rechner vom Netz trennte und die Systeme neu startete, war der Spuk verflogen; sobald man sie aber wieder mit dem Netz verband, trat die »Infektion« erneut auf. Immerhin konnte nach intensiver Analyse die Ausbreitung am 3. November 1988 morgens (in Harvard gegen 4 Uhr) gestoppt werden. In der Nacht zum 4. November war schließlich in Purdue zunächst ein Gegenmittel (»worm condom«) entdeckt worden, mit dem die Rechner vor einem Befall geschützt werden konnten, wenig später auch ein wirksames »Heilmittel« (»worm cure«) zur Sanierung befallener Rechner. Die schnelle Aufklärung ist zweifelsohne durch die hervorragende Zusammenarbeit der Fachleute, die übergreifende Arbeitsgruppen (Berkeley, MIT, Purdue) bildeten, erleichtert worden.

Nach intensiver Recherche wurde festgestellt, daß der INTERNET-Wurm von einem Terminal an der Cornell Universität (Ithaka/New York) ausgegangen war. Dort hatte der Informatik-Student Robert T. Morris jr., der einen Ruf als erfahrener Hacker genoß, auf einem Rechner des Labors für »Künstliche Intelligenz« am MIT (Boston/Massachusetts) in mehrwöchiger Arbeit ein Programm vorbereitet, das er am 2. November nachmittags startete. Auf dem MIT-Rechner wurde zunächst ein erstes »Opfer« gesucht. Im System haben Benutzer für ihre Netzarbeit reichlich

Adressen anderer Rechner sowie Paßworte zu deren Benutzung gespeichert, die durch »trial and error« oder »gezieltes Raten« herausgefunden werden können. Häufig verwenden Benutzer nämlich Vornamen oder bekannte Begriffe als Paßworte; durch systematisches Suchen konnte Morris so Listen bekannter Paßworte gezielt anreichern. An solche Adressen wird beim INTERNET-Wurm automatisch ein »Helfer«-Programm übersandt, welches das spezielle Wurm-, Adressenbuch- und Versendeprogramm auf dem Rechner installiert.

Der UNIX-Wurm war so programmiert, daß man ihn nicht leicht entdecken konnte; er breitete sich durch eine Fehlschätzung des Programmierers extrem schnell aus. Außerdem verschleierte er seine Verbindungen zurück zum »infizierenden« System. Bei den »Aufräumarbeiten« wurde schnell erkannt, daß der Wurm-Autor über beachtliches Fachwissen verfügte, und er kannte Tricks und Fehler von UNIX. Als wenige Tage nach dem Vorfall die Rechner-Manager von Cornell seine Identität aufgrund auffälliger Ähnlichkeiten verwendeter Paßworte feststellten, erinnerten sich die Fachleute wichtiger Arbeiten des Hacker-Vaters: Robert T. Morris Senior war lange im Bell Labor an Problemen der UNIX-Sicherheit, vor allem mit bekannten Arbeiten über Paßwort-Schutz tätig; inzwischen ist er Direktor des US-Computer-Sicherheitszentrums. Über die Absichten des Programm-Autors wird spekuliert: Morris jr., Informatik-Student, nach



deutschen Maßstäben etwa 4. bis 6. Semester, wollte wohl ausprobieren, wie sich die von Shoch und Hupp bereits 1982 beschriebenen »Würmer« wohl in einem übergeordneten Netz verhielten. Bald nach dem Start soll Morris über die unerwartet rasante Ausbreitung seines Wurms entsetzt gewesen sein, und ein Kommilitone informierte darüber die New York Times, deren elektronischer Artikeldienst später von deutschen Medien ausgewertet wurde.

Im Dezember 1988 wurde ein weiterer Wurm festgestellt, der ebenfalls Rechner der US-Firma Digital Equipment in Universitäten und Forschungseinrichtungen befiel, die durch deren Netzsystem DECNET verbunden waren. Unter derartigen Vorfällen ist 1989 der »Worm Against Nuclear Killers« (»WANK-Wurm«) wegen der Motive seiner Autoren interessant: Er wurde offenbar aus Protest gegen den Start der Galileo-Sonde produziert, von deren Plutoniumbatterien man eine Verseuchung der Erdatmosphäre befürchtete.

Mißbrauch von Computersystemen

Krimineller Mißbrauch von Computern und Programmen wird in unterschiedlichen Spielarten in den USA bereits seit langem beobachtet und systematisch ausgewertet. So hat Donn Parker am Stanford Research Institute seit den sechziger Jahren über 2000 Fälle von »Computer Mißbrauch« dokumentiert, mit Fällen von »Computer-Vandalismus« (Sprengstoffanschläge von außen, Schüsse wütender Mitarbeiter auf Computer und Peripheriegeräte) bis zum Mißbrauch von Software (von gut verschleierte Falschbuchungen bis zu betrügerischen Manipulationen an Programmen).

Längst setzt auch die internationale »Organisierte Kriminalität« Computer für ihre Zwecke ein, und man muß heute davon ausgehen, daß bei kriminellen Transaktionen großen Maßstabes (wie dem Finanzbetrug bei VW) stets auch Computer eine wichtige Rolle spielen. Solche Straftaten werden von DV-kundigen Missetätern oft derart verschleiert (etwa durch Löschen aller »Fußspuren«), daß eine spätere Aufklärung erheblich erschwert wird. Die strafrechtliche Bewertung von Computer-Delikten bereitete lange erhebliche Probleme, weil etwa »Sachbeschädigung« (§303 StGB) die Beschädigung oder Zerstörung »fremder Sachen« ahndet, wobei Juristen die Begriffe »Daten« und »Programme« offenbar nicht als »Sache« einordnen konnten. Mit dem Zweiten Gesetz zur Bekämpfung der Wirtschaftskriminalität (vom 15.5.1985) hat der Deutsche Bundestag im Strafgesetzbuch (StGB) inzwischen spezifische Straftatbestände der Computerkriminalität beschrieben:

§202a (StGB) stellt das »Ausspähen von Daten«, sofern diese »elektronisch, magnetisch oder sonst nicht unmittelbar wahrnehmbar gespeichert sind oder übermittelt werden« (Absatz 2) unter Strafe:

»Wer unbefugt Daten, die nicht für ihn bestimmt und die gegen unberechtigten Zugang besonders gesichert sind, sich oder einem anderen verschafft, wird

mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft« (Absatz 1).

§303a (StGB) stellt die »Datenveränderung« (sowie auch, in Absatz 2, den Versuch dazu) unter Strafe:

»Wer rechtswidrig Daten (§202a Absatz 2) löscht, unterdrückt, unbrauchbar macht oder verändert, wird mit Freiheitsstrafe bis zu zwei Jahren oder mit Geldstrafe bestraft« (Absatz 1).

§303b (StGB) bestraft die »Computersabotage« (wie auch, in Absatz 2, den Versuch dazu):

»Wer eine Datenverarbeitung, die für einen fremden Betrieb, ein fremdes Unternehmen oder eine Behörde von wesentlicher Bedeutung ist, dadurch stört, daß er 1. eine Tat nach §303 Abs.1 begeht oder 2. eine Datenverarbeitungsanlage oder einen Datenträger zerstört, beschädigt, unbrauchbar macht, beseitigt oder verändert, wird mit Freiheitsstrafe bis zu fünf Jahren oder mit Geldstrafe bestraft.«

Schließlich wurde der »klassische« Betrug (§263 StGB) um den Tatbestand des »Computerbetruges« erweitert (§263a):

»Wer in der Absicht, sich oder einem Dritten einen rechtswidrigen Vermögensvorteil zu verschaffen, das Vermögen eines anderen dadurch beschädigt, daß er das Ergebnis eines Datenverarbeitungsvorgangs durch unrichtige Gestaltung des Programms, durch Verwendung unrichtiger oder unvollständiger Daten, durch unbefugte Verwendung von Daten oder sonst durch unbefugte Einwirkungen auf den Ablauf beeinflußt, wird mit Freiheitsstrafe bis zu fünf Jahren oder mit Geldstrafe bestraft.«

Bei allem »Fortschritt« sind aber erhebliche Mängel dieser Vorschriften zu erkennen. Besteht die erforderliche Kompetenz von Richtern, sich (in Bewertung vielleicht gegenteiliger »Expertisen«) ein eigenes Bild von den Software-Problemen zu bilden? Immerhin war vor einiger Zeit ein bayrisches Gericht nicht in der Lage, einen von dem Programmierer »vorsorglich« in das »GURUGS-Programm« eingebauten Löschmechanismus (eine sogenannte »Zeitbombe«) rechtzeitig zu entdecken und die Münchener Bundeswehrhochschule vor der Löschung des Programmes zu schützen.

Auch die Anwendbarkeit des Paragraphen 202a, Voraussetzung für die §303a und 303b, erscheint zweifelhaft, wenn man jüngste Fälle analysiert. Kein Gericht hat bisher interpretiert, ob die Paßwort-Sicherungen der NASA- und ähnlicher technisch-wissenschaftlicher Rechner, die 1987 ein spektakuläres Eindringen bisher unbekannter (deutscher) Hacker nicht verhinderten, eine »besondere Sicherung« darstellt. Angesichts der relativ einfachen Mechanismen, mit denen die Hacker mit vielfach veröffentlichten Techniken in die Systeme eindringen, kommen erhebliche Zweifel daran auf, ob die (auf dem §202a fußenden) massiven polizeilichen Aktionen (Hausdurchsuchungen, Beschlagnahme) beim Hamburger Computer Chaos Club einer richterlichen Überprüfung standhalten können. Überdies erscheint fraglich, ob deutsche Strafbestimmungen auf das Eindringen in ausländische Rechner anwendbar sind.

Fuzzy-Logik für unsichere Inhalte

Hans-Jürgen Zimmermann

Die unscharfe Logik hält Einzug in die Informatik. Das simple Ja-Nein-Raster reicht nicht aus zur Beschreibung der meist widersprüchlichen Wirklichkeit. Fuzzy Logic macht Unsicherheiten für den Rechner kalkulierbar.

Die Bezeichnung »Fuzzy« – im Deutschen sowohl »Fassi« oder »Fuzzi« ausgesprochen – kommt aus dem Englischen. Es bedeutet »verschwommen« oder »mehrwertig«. »Fuzzy« kann auch einen Übergang beschreiben; etwa von einer Welt, in der man ganz klar abgegrenzte, schwarze, rote, weiße Pixel hat, zu einer Welt des Mehr oder Wenigers, in der es graduelle Unterschiede gibt, wo nicht mehr alles eindeutig ist. Die wissenschaftliche Fuzzy-Theorie selbst ist gar nicht so neu, sie ist rund 25 Jahre alt. Sie versucht einen Weg zu beschreiben, wie das Interface zwischen Rechner und Mensch vereinfacht und benutzerfreundlich gestaltet werden kann. Bisher fehlte beispielsweise ein Interface, um andere Sprachen per Computer zu verarbeiten, weil der Computer den Inhalt von Worten nicht versteht. Er kann zwar Worte als Symbole akzeptieren, wenn es ein beschränktes Vokabular ist. Aber er kann kein Wort mit Wissen füllen. Das müssen wir selbst tun.

Das wurde auch in den sogenannten wissensbasierten Systemen versucht, in Expertensystemen zum Beispiel. Eigentlich wurde dabei bis jetzt immer ein bißchen gemogelt, indem man die graduellen Bewertungen – ob etwas nun »sehr wahr« ist oder »wahr« ist oder »sehr attraktiv« oder »rot« oder »gelb« – nicht dem Rechner überließ, sondern in den Köpfen des jeweiligen Experten beließ. Der machte eine zweiwertige Entscheidung: ja oder nein, wahr oder falsch, und dann erst setzte der Rechner ein. Wir mußten also selbst bei wissensbasierten Systemen immer noch zwischen dem Wissen des Experten und dem Rechner eine Art Übersetzer einschieben, wobei es egal war, ob der in den Köpfen der Experten oder der Benutzer war.

Genau das soll in Zukunft weitgehend mit den sogenannten »Fuzzy-Sets« erledigt werden. Damit kann man die Worte mit einem Inhalt oder Wissen füllen und sie im Rechner entsprechend verwerten.

In Sachen Fuzzy haben wir eine Situation, die ein bißchen an die Zeit vor zehn bis fünfzehn Jahren erinnert: In Deutschland waren damals Industriezweige wie Gebrauchselektronik oder Fototechnik sehr stark, aber kurze Zeit später merkten wir, daß die Japaner in diesen Bereichen in Führung gegangen waren. So etwas ähnliches ist im Moment im Gange: Die Japaner haben auf dem Gebiet der »Fuzzy Control« ungefähr zwei bis drei Jahre Vorsprung vor den USA und vor Europa. Diesen Vorsprung nutzen sie

jetzt vor allem auf dem Markt für Gebrauchsgüter aus: Waschmaschinen, Mikrowellenherde, Staubsauger, ABS werden in Japan schon in Fuzzy-Versionen angeboten. Das Wort »Fuzzy« gilt hier als der Inbegriff neuester Technologie.

In Europa und Amerika dagegen hat man bis jetzt kaum davon Notiz genommen – weder in der Wirtschaft noch in der breiten Öffentlichkeit. Bei uns haben diverse Veröffentlichungen in der letzten Zeit das Interesse an Fuzzy geweckt. Tatsächlich beginnt man jetzt auch bei uns, diese Tools zu benutzen und einzusetzen. Die deutsche Forschung und Industrie haben sogar sehr gute Chancen, weil hier das methodische Werkzeug etwas besser ist als das, mit dem man in Japan vor ungefähr fünf Jahren angefangen hat.

»Fuzzy Logic« ist nur ein ganz kleines Gebiet innerhalb eines viel größeren Bereiches. Es ist eine andere Art des Denkens. Es bedeutet eine gewisse Schwierigkeit, sich darauf umzustellen, weil wir aufgrund unserer Ausbildung andere Denkstrukturen gewohnt sind, wenn es um Technik oder Prozeßkontrolle oder Regelungstechnik geht.

Zunächst geht es um den Bereich der Unsicherheiten. Bei Fuzzy dreht sich alles um Unsicherheitsmodellierung. Wenn wir über Unsicherheiten sprechen, dann sind wir gewohnt, an Wahrscheinlichkeiten zu denken. Jeder weiß, daß das auch gar nicht schwierig zu formulieren ist, denn wir haben ja die bewährte Wahrscheinlichkeitstheorie. Auf dieser Grundlage können wir Modelle bauen und Variablen berechnen.

Zunächst einmal müssen wir uns fragen, ob wir uns immer auf klar definierte Ereignisse konzentrieren können oder ob es um Dinge geht, die nicht so klar abgegrenzt sind. Zum Beispiel »attraktive« Projekte, »schöne« Urlaubstage, »erfolgreiche« Manager. Daneben gibt es noch Bewertungen, wie beispielsweise »0,8«. Wer sagt uns, was 0,8 ist? Sind wir sicher, daß es nicht 0,79 oder 0,85 ist? Immer wenn so etwas eintritt, bekommen wir als Ergebnis nicht mehr die schönen Wahrscheinlichkeitswerte. Dazu kommt, daß die Wahrscheinlichkeitstheorie als formale Theorie nur für ganz bestimmte Situationen gedacht ist.

Zusätzlich zu diesen sogenannten stochastischen Unsicherheiten, in denen einfach der Zufall modelliert wird, gibt es andere Unsicherheiten, denen man sich innerhalb der letzten zehn bis fünfzehn Jahre intensiv zugewandt hat.

Da gibt es zunächst die sogenannte lexikale oder sprachliche Unsicherheit, die in unserer tagtäglichen Sprache sehr viel stärker ausgeprägt ist als die stochastische Unsicherheit, beispielsweise in Ausdrücken wie »große« Männer, »heiße« Tage, »stabile« Währungen. Das sind alles unscharfe, unsichere Ausdrücke. Man kann einfach nicht genau sagen, ob ein bestimmter Mann zu den großen Männern gehört oder nicht. Wenn man ein achtjähriges Kind fragt, wird es eine ganz andere Vorstellung haben von einem »großen Mann« als ein Erwachsener. Das gleiche gilt natürlich für »junge« oder »alte« Männer.

Wenn man sich einen Satz ansieht wie »Es ist wahrscheinlich, daß wir einen guten Erfolg erreichen werden«, würde man den zunächst mal ohne weiteres als stochastische Unsicherheit einordnen. Bei genauerer Betrachtung fallen zwei Problembereiche auf. Erstens: Was ist denn der »gute Erfolg«? Und zweitens: Was heißt denn »es ist wahrscheinlich«? Ist das eine Wahrscheinlichkeit von 0,7 oder 0,2 oder 0,8, oder ist es einfach eine Wahrscheinlichkeit von 0?

Wir unterhalten uns sehr oft in solchen Formen. Das ist eben einer der Unterschiede zwischen Menschen und Rechnern: Menschen können sich sinnvoll über unscharfe Phänomene unterhalten, weil sie aufgrund dieses »common sense«, dieses Weltwissens,



der jeweiligen Aussage in einem bestimmten Kontext einen Inhalt zuordnen. Das kann der Rechner nicht. Deswegen bleiben für ihn die Ausdrücke zunächst einmal unsicher, unscharf, solange wir sie nicht definiert haben. Das Wissen, die Expertise, die Erfahrung, die wir mit der Verarbeitungsfähigkeit einer hohen Rechenleistung kombinieren wollen, dieses menschliche Wissen liegt leider nicht in einer Weise vor, daß wir es direkt in den Computer stecken könnten. Es liegt bestenfalls in natürlicher Sprache vor. Dieses Wissen müssen wir zunächst einmal in eine Art Metasprache übersetzen, die dem Abgeber des Wissens verständlich ist und gleichzeitig innerhalb des Computers verstanden wird.

Es gibt außer diesen beiden – der lexikasprachlichen und der stochastischen Unsicherheit – noch eine dritte Unsicherheit, die damit zu tun hat, daß wir in unserem Kurzzeitgedächtnis eine relativ begrenzte Aufnahmekapazität haben. Wir können nur relativ wenig Informationen im direkten Zugriff im schnellen Speicher behalten. Anders beim Langzeitgedächtnis. Nun gibt es aber eine ganze Menge Ausdrücke, die – um sie scharf zu bestimmen – eine ganze Menge beschreibende Determinanten brauchen. Zum Beispiel Begriffe wie »Paradigma« oder »Kreditwürdigkeit« oder »Marktattraktivität«.

Wir Menschen suchen uns normalerweise irgendwelche Deskriptoren heraus. Wenn zwanzig notwen-

dig wären und wir fünf abspeichern können, dann suchen wir uns diese fünf aus und charakterisieren den Begriff, zum Beispiel »Kreditwürdigkeit«, durch diese fünf Eigenschaften. Der Gesprächspartner weiß aber nicht, welche fünf Eigenschaften benutzt werden und unterstellt möglicherweise ganz andere. Dann sprechen wir über die gleichen Worte und meinen ganz verschiedene Inhalte. Das ist sicherlich außerordentlich nützlich in manchen Gebieten, in der Politik etwa. Wenn wir diese Fähigkeit zur inhaltslosen Vereinbarung nicht hätten, kämen wir wahrscheinlich nie weiter. Man kommt natürlich in Schwierigkeiten, wenn man versucht, das Vereinbarte mit Inhalt zu füllen, und dann bemerkt, daß man auf den verschiedenen Seiten die Sache eigentlich ganz verschieden gesehen hat.

Fuzzy Set Theory

»Fuzzy Set Theory« heißt auf deutsch etwa »Theorie unscharfer Mengen«. In der Mengenlehre ist eine Menge eine Gruppe von Elementen, die mindestens eine gemeinsame Eigenschaft haben. Diejenigen Elemente, die diese Eigenschaft nicht haben, gehören der Menge nicht an; und alle Elemente, die diese Eigenschaften haben, gehören der Menge an. In der Booleschen Algebra werden solche Definitionen durch die sogenannte charakteristische Funktion dargestellt. Bezogen auf die Elemente heißt der Wert 1 dann: »es gehört zur Menge«, und der Wert 0: »es gehört nicht dazu«.

Eine unscharfe Menge löst diese Zweiwertigkeit aus Zugehörigkeit/Nichtzugehörigkeit in ein Kontinuum mit fließenden Grenzen auf. Das Fließen der Grenzen wird durch den sogenannten Zugehörigkeitsgrad oder die Zugehörigkeitsfunktion ausgedrückt. Im Gegensatz zur klassischen Menge besteht das Element einer Unschärfmenge aus einem Paar: Aus dem Element selbst und dem Zugehörigkeitsgrad dieses Elementes zu dieser unscharfen Menge. Dieser Zugehörigkeitsgrad oder die Zugehörigkeitsfunktion hat viele Vorteile. Aber auch den großen Nachteil, daß man den Unterschied zur Wahrscheinlichkeitstheorie sehr viel weniger sieht, weil zufällig beide auch den Bereich $1 - \text{»gehört zur Menge«}$ – abbilden.

Wichtiger werden jetzt die sogenannten linguistischen Variablen. Die linguistische Variable kann man sich am einfachsten so vorstellen:

Der Wert A kann beispielsweise die Werte 1, 2, 3 bis 10 annehmen. Damit ist die Variable definiert. Wird eine Zufallsvariable definiert, würde man sagen, A sei normal verteilt mit den Mittelwerten in dem einen Bereich und den Streuungen in dem anderen Bereich. Eine linguistische Variable wird definiert durch Angabe der sogenannten »Term-Menge«. Diese Term-Menge sind linguistische Ausdrücke. Nehmen wir als linguistische Variable »das Alter«, dann können die Terme – die Werte, die die Variable annehmen kann – sein: »sehr jung«, »jung«, »alt« und »sehr alt«. Der Unterschied zu den normalen linguistischen Variablen ist, daß jeder dieser Terme inhaltlich durch

eine unscharfe Menge definiert ist. Bei der linguistischen Variable »Alter« geht das am einfachsten. Man nimmt das Alter in Lebensjahren und leitet daraus die Definitionen ab. Die unscharfe Menge »sehr jung« wäre also definiert auf den Bereich zwischen 20 und 30, »jung« zwischen 20 und 35 oder 45 und so weiter und so fort. Damit ist dem linguistischen Ausdruck »sehr jung« eine ganz genaue Bedeutung zugeordnet worden. Diese Bedeutung kann sich natürlich je nach Kontext ändern. Die linguistischen Terme werden einem Zehnjährigen anders interpretiert als von einem Siebzijährigen. Man kann sie immer wieder parameterisieren und beispielsweise auf die Situation von Kindern oder von Rentnern beziehen. Damit hat man die linguistischen Variablen, deren sich die natürliche Sprache bedient, und die formale Definition, wie sie der Rechner verstehen kann. Auf diese Weise wird das Interface zwischen natürlicher Sprache und formaler Computersprache hergestellt.

Wenn man mit solchen Mengen arbeitet, dann muß man zwangsläufig auch Regeln aufstellen, wie man die Mengen manipulieren oder zusammenfassen kann. Die üblichsten Regeln sind der Durchschnitt, die Schnittmenge, die Vereinigung und das Komplement. Der Durchschnitt in der Mengentheorie entspricht dem »Und« in der Logik. Die Vereinigung in der Mengentheorie entspricht dem inklusiven »Oder« und dem Komplement der »Verneinung«. Das heißt, die »Fuzzy Set«-Theorie ist auf der einen Seite eine Verallgemeinerung der Mengenlehre und auf der anderen Seite oder in dem gleichen Grade eine Verallgemeinerung der zweiwertigen, klassischen Logik.

Min/Max-Modell

In der alten, elementaren Theorie wurden diese Operatoren sehr einfach definiert. In der Theorie, wie sie etwa seit 1965 bekannt ist, die der Fuzzy-Kontrolltechnik, die aus Japan herüberkommt, weitgehend zugrunde liegt wurde der Durchschnitt oder das logische »Und« durch das Minimum ausgedrückt (immer bezogen auf die Zugehörigkeitsfunktion), die Vereinigung als das Maximum und das Komplement als 1 minus dem Zugehörigkeitsgrad. Das hat den großen Vorteil, daß alles noch stimmt, wenn man es mit klassischer Mengenlehre oder Buchstabenalgebra berechnet. Wenn man nur die Werte 0 oder 1, wahr oder falsch zuläßt, dann ist es gleichgültig, ob man das Minimum nimmt oder ob man das Produkt nimmt. Bei »nur 0« und »nur 1« kommt beides zu dem gleichen Ergebnis. Nur in dem Moment, wenn wir uns nicht auf die Werte »0«, »1«, »wahr«, »falsch«, »zugehörig« oder »nicht zugehörig« beschränken, sondern andere Werte zulassen, macht es natürlich einen großen Unterschied, welches mathematische Modell zugrunde gelegt wird.

Ein Beispiel: Das Management-Team eines Hotels soll die Schwimmbadtemperatur einstellen. Ein männliches Mitglied des Vorstandes sagt, die meisten Gäste seien Manager, die den ganzen Tag im Büro

oder in Besprechungen sitzen, also bräuchten sie eine erfrischend kühle Temperatur im Schwimmbad, etwa 18 Grad. Eine Dame im Vorstand meint, die weiblichen Gäste hätten das Bedürfnis, sich ganz gemütlich auszuruhen, man bräuchte deshalb eine gemütlich warme Temperatur im Swimmingpool. Da man nur eine Temperatur einstellen kann, muß eine Temperatur gefunden werden, die als angenehm empfunden wird und gleichzeitig erfrischend und gemütlich wirken soll. Diese beiden unscharfen Mengen müssen dann durch das linguistische und logische »Und« verbunden werden. Die Skala der erfrischenden Temperaturen reicht von etwa 17 Grad bis 23 Grad, die der gemütlichen Temperaturen von 22 Grad bis 26 Grad. Wenn man das Minimum der warmen Temperaturen und das Maximum der erfrischenden Temperaturen vergleicht, kommt als Kompromiß ein Wert von etwa 22 Grad heraus. So hätte man das Ergebnis nach einem Min/Max-Modell ermittelt.

Für die »Fuzzy Set«-Theorie ist nun wichtig, was innerhalb der letzten zwölf Jahre passiert ist: Drei ganz große Familien von mathematischen Modellen sind in diesen Jahren entwickelt worden. Zwei davon gehören eher zum Feld der reinen Mathematik – die sogenannten T-Normen und T-Konormen. Die dritte Gruppe sind die sogenannten »mittelnden Operatoren«. Sie haben den großen Vorteil, daß sie den Inhalt des menschlichen »Und« sehr viel besser abbilden. Aus der Gruppe der mittelnden Operatoren soll hier ein einziger herausgenommen werden, der sogenannte »Gamma-Operator«.

Wenn wir zwei Sachen, beispielsweise Ziele oder Nebenbedingungen, miteinander verbinden wollen, dann haben wir in unserem Kopf sogenannte »latente Operatoren«. Das sind gefühlsmäßige Faktoren, nach denen wir die Sachen miteinander verbinden. Sie liegen normalerweise zwischen den beiden Extremen des logischen »Und« und des inklusiven »Oder«. Beim logischen »Und« gibt keine Kompensation, da nimmt man immer das Minimum, ganz egal wie groß das andere ist. Beim inklusiven »Oder« wird voll kompensiert, und wir liegen irgendwo dazwischen. Da uns unsere Sprache nur zwei Worte anbietet, um eine solche Aggregation zu definieren, sind wir bereits bei dem sprachlichen Ausdruck unserer Erfahrung gezwungen, entweder das »Und« oder das »Oder« zu wählen, je nachdem, wo wir dazwischen liegen.

Um beispielsweise Kreditwürdigkeit so zu definieren, daß sie im Rechner definiert werden kann, brauchen wir diverse inhaltliche Festlegungen. Das konnten wir bis jetzt nicht, weil wir die »Unds« und »Oders« überhaupt nicht anders definieren konnten. Die Definition durch Gamma-Operatoren geschieht, indem man die Kreditanträge von Fachleuten bewerten läßt. Damit wird jeweils ein Zugehörigkeitsgrad zu den Kriterien implementiert. Daraus kann der Gamma-Wert berechnet werden. Der Kreditsachbearbeiter weiß also gar nicht, was für ein Gamma er benutzt, sondern man stellt das aufgrund der von ihm gestellten Entscheidung fest. Dabei hat sich ge-

zeigt, daß man die Entscheidung der Kreditsachbearbeiter mit einer Treffsicherheit von 97 Prozent voraussagen konnte.

Aufgrund der dualen Logik bilden wir wissensbasierte Systeme. Da gibt es nur zwei Wahrheitswerte, nämlich »ja« und »nein« oder »wahr« und »falsch«. Etwas anderes ist in der dualen Logik nicht möglich. Alle wissensbasierten Systeme – von ein paar Ausnahmen abgesehen – benutzen diese Art des Schließens.

Die Aussagen, die wir in unserer normalen Sprache benutzen, um Situationen zu schildern, sind in der dualen Logik nicht vorhanden. Wir können sie auch nicht formulieren. Damit wird klar, wie weit die duale Logik und die darauf basierenden Systeme von der Art, wie Erfahrung in Köpfen gespeichert wird, entfernt sind. Daß wir überhaupt mit diesen Systemen arbeiten können, ist darauf zurückzuführen, daß wir ein bißchen mogeln: Wir zwingen den Benutzer oder den Experten, seine Aussagen in diesen Kategorien zu machen – auch wenn das unrealistisch sein mag. In der »Fuzzy Logic« werden die Wahrheitswerte der dualen Logik als linguistische Variablen angesehen, als sogenannte »Fuzzy Sets«. Sie haben also nicht mehr 0 oder 1, ja oder nein, falsch oder wahr, sondern Werte zwischen »100prozentig wahr«, »nicht ohne Wahrheit«, »halbwahr«, »fast wahr«, »absolut wahr«. Alle diese linguistischen Werte, diese Terme, sind als unscharfe Mengen definiert. Das ist Fuzzy-Logik.



Damit kommen wir aber noch nicht sehr weit. Es fehlen noch die weiteren Bedingungen, das sogenannte »approximant reasoning« oder »approximatives Lesen«. Dabei geht man einen Schritt weiter und sagt: Außer den Wahrheitswerten, die als linguistische Variable gesehen werden, wird auch noch zugelassen, daß die Inhalte der Regeln und auch der Beobachtungen unscharf sein können. Sie können sprachliche und lexikale Unschärfe haben, also durch Zugehörigkeitsfunktion unscharfer Mengen formuliert werden. Was noch nicht erreicht wurde, ist das Fallenlassen der Identität. Das macht man im sogenannten »plausible reasoning«. Ein Beispiel: Die Regel sei »Wenn eine Tomate rot ist, dann ist sie reif«. Wir beobachten die Tomate als sehr rot. Also würden wir mit all unserem Hintergrundwissen sagen, daß sie »sehr reif« ist. Dann wird Hintergrundwissen benötigt. Denn wenn die Tomate immer reifer wird, kann sie noch andere Farbtöne annehmen, dann ist sie auch nicht mehr so attraktiv.

Die Differenzmaschine

Die Fuzzy-Systeme benutzen in den meisten Fällen das »approximant reasoning« und nur teilweise das »plausible reasoning«, um möglichst nahe an das heranzukommen, wie ein Mensch seine Erfahrungen verarbeitet.

Ein erstes Beispiel für die Anwendungen sind Expertensysteme. Was im Moment auf dem Markt ist, ist die sogenannte »Fuzzy Control«. Das sind regelungstechnische Anwendungen. Das größere Potential liegt bei den unscharfen Expertensystemen, die aber noch nicht so weit entwickelt sind. Stellen wir uns ein Expertensystem vor, das mit linguistischer Information arbeitet, also auch die Regeln linguistisch eingegeben bekommt, das in der Lage ist, die Unsicherheiten der verschiedensten Art tatsächlich adäquat zu modellieren, so daß sie auch im Rechner entsprechend verarbeitet werden können. Nach Möglichkeit sollen die Ergebnisse für den Benutzer in linguistischer Form vorliegen. Wir wollen also nicht irgendeine Aussage haben und dann einen Sicherheitskoeffizienten von 0,378, mit dem kein Mensch etwas anfangen kann, sondern wir möchten die Aussage in der Sprache haben, in der wir denken, mit der wir umgehen können. Wenn wir das erreichen wollen, dann haben wir bei einem solchen Expertensystem einige Probleme zu lösen. Wir haben zunächst einmal den eigentlichen Schließautomatismus, die Differenzmaschine.

Beim Input, bei der Eingabe der Daten, haben wir die Möglichkeit der Unsicherheitsrepräsentation. Damit wir auch Abweichungen verarbeiten können, müssen wir in der Lage sein, den Grad der Ähnlichkeit der Beobachtung mit dem Inhalt der Regeln festzustellen. Und da wir hier nur unscharfe Wissensverarbeitung haben, müssen wir immer dann, wenn unsere Eingangsdaten scharf sind – beispielsweise numerische Werte sind –, diese Daten in unscharfe Mengen umwandeln, also »fuzzyfizieren«.

Beim Output haben wir das entsprechende Komplement dazu. Hier kommen unscharfe Mengen heraus. Diese Funktionen müssen wieder in natürliche Sprache umgewandelt werden.

Unter Umständen müssen verschiedene Arten der Unsicherheiten aggregiert werden, damit sie verständlich werden.

In der »Fuzzy Control« ist der einzige Unterschied der, daß wir keine menschlichen Beobachter haben, sondern Meßgrößen. Wir beobachten an einem Regelungskreis, an einem technischen Prozeß irgendwelche Größen und nehmen die meist numerisch ab. Außerdem wollen wir in der »Fuzzy Control« aufgrund unserer Erfahrung technische Prozesse steuern. Wir wollen aber, ähnlich wie beim Expertensystem, das Innere als eine unscharfe Erfahrung halten. Nach der Fuzzyifizierung werden die ermittelten Werte ganz normal verarbeitet, genau wie im Expertensystem. Dann erhält man Ergebnisse, die aber nicht in linguistischer Form benötigt werden, sondern in numerischer Form, weil damit Ventile eingestellt werden sollen, Brenntemperaturen erreicht werden und so weiter. Also muß man sie defuzzyfizieren. Das sind zusätzliche Module, die in einem »Fuzzy Control« System benötigt werden.

Heute werden bereits Tools angeboten, die uns helfen, solche Systeme zu entwickeln. Auf der Hardwareseite gibt es mittlerweile sechs oder sieben Fuzzy-Chips, Hardware-Implementierungen der Min/Max-Logik. Auf der Softwareseite gibt es nur zwei ernstzunehmende Entwicklungen. Einmal das Fuzzy-C von Togai, eine japanisch-amerikanische Firma in Kalifornien. Da arbeitet man mit Fuzzy-Logik auf Min/Max-Basis, die aber bei komplexeren Systemen nicht mehr funktioniert. Das System eignet sich dafür, Fuzzy-Elemente auf Min/Max-Basis in ein C-Programm zu kompilieren.

Das autonome Auto

Bei einer anderen Software-Anwendung mit der Bezeichnung FIT ist eine relativ komfortable grafische Benutzeroberfläche geschaffen worden. Hier sind modular aufgebaute Methoden angegliedert worden, bei denen man sowohl zwischen der Min/Max-Fuzzy-Logik als auch dem approximativen oder plausiblen Lesen wählen kann. Es gibt dabei auch Möglichkeiten, verschiedene Zugehörigkeitsfunktionen einfach darzustellen, zu kalibrieren, verschiedene Operatoren oder verschiedene Arten der Erfahrung zu benutzen. Darüber hinaus können verschiedene Interface-Module eingesetzt werden, mit deren Hilfe die Fuzzy-Systeme mit PC und anderen Systemen verbunden werden können.

In der Anwendung sieht es in Deutschland noch etwas mager aus. Aus Japan kommen bereits eine ganze Menge Geräte, wie Waschmaschinen, Trockner, Mikrowellen, Staubsauger mit »Fuzzy Control«. Oder Video-Kameras von Sony und Panasonic mit einer sehr guten Fuzzy-Kontrolltechnik, die ein Verwackeln der Bilder verhindern soll.



Man muß allerdings bei den meisten Anwendungen feststellen, daß das dort implementierte Wissen für uns nutzlos ist, und zwar aus dem einfachen Grunde: Dieses Wissen ist auf die japanische Kultur abgestellt. Dort wird zum Beispiel kalt gewaschen, und es gibt ganz andere Stoffe. Das gibt uns natürlich eine Chance, in der Zwischenzeit selbst etwas herzustellen, was hier funktionieren würde. Der japanische Mikrowellenherd arbeitet sehr gut bei Reisgerichten und kleinen Fischeiben. Aber bei Steaks funktioniert er nicht, Auftauen geht nicht und die meisten anderen Sachen, die wir hier machen, auch nicht.

Als Anschauungsobjekt wurde an der Technischen Hochschule in Aachen ein Modellauto entwickelt. Der Wagen ist rund einen halben Meter lang, 3,5 Kilogramm schwer und hat eine Motorleistung von 1 PS, was ihn im Leistungsgewicht über die meisten Rennwagen setzt. Er hat Scheibenbremsen und einen großen Elektromotor.

Die »Fuzzy Logic« ist in einem eingebauten AT und fünf Transputern untergebracht. Das Modellauto fährt ohne jede Kommunikation aufgrund gespeicherten Wissens mit einer Geschwindigkeit bis zu 80 Kilometern pro Stunde in der Gegend herum, weicht Hindernissen aus und bremst, wenn es nicht mehr weiterkommt.

Dieses Auto wird bewußt in instabile Zustände gebracht, aus denen es sich aufgrund des in der Wissensbasis gespeicherten Wissens selbst abfangen muß. Es ist natürlich nicht beabsichtigt, daß jemand mit so einem Auto fahren soll, aber man kann damit lernen, worauf es in kritischen Fahrsituationen ankommt.

Tanz in der Feedback-Schleife

David Rokeby

David Rokebys Installation verwandelt Bewegung und Geräusche in Bilder und Musik. Der Tänzer wird zum Resonanzkörper des Systems.

Bei den Installationen, die unter die allgemeine Bezeichnung »Very Nervous System« fallen, benutze ich Videokameras, Bildprozessoren, Computer, Synthesizer und ein Klangsystem, um ein Umfeld zu schaffen, in dem Körperbewegungen Klänge und/oder Musik erzeugen. In der Installation nimmt Klang gestalterische Form an, sowohl als Ausdruck des Körpers wie auch als physikalische Wirklichkeit, der man körperlich begegnet.

Ich interessiere mich für den Klang des Systems und den Tanz der Person im Raum, und doch besteht der Kernpunkt meiner Arbeit weder im Klang noch im Tanz. Sie dreht sich vielmehr um die Beziehung, die sich zwischen der Klangeinrichtung und der tanzenden Person entwick-

kelt. (Bei einem Stück gab es einen Klang, den man nur finden könnte, wenn man umherginge, als ob man ein 40-Pfund-Gewicht schleppen würde.)

Das System beobachtet und singt; die Person hört zu und tanzt. Aber die Beziehung, die sich entwickelt, ist nicht einfach ein Dialog zwischen Person und System. Der wechselseitige Dialog setzt eine Trennung der Funktionen Empfangen und Antworten voraus. Für das System aber sind Wahrnehmung und Ausdruck simultan. Als Ergebnis formen das System und der Teilnehmer eine Feedbackschleife, die sehr eng ist, aber sehr komplex.

Wenn die Installation leer ist, steht sie da wie ein stilles Bewußtsein, ein Teil eines empfindlichen Systems: Kameras als Augen / ein computerisiertes Gehirn / in der Software ein komplexer Satz aus Filtern und Übersetzern (vielleicht gar ein Satz von Meinungen, der Ansatz einer Ästhetik?) / synthetische Stimmbänder / Lautsprecher als Mund... Mit der ersten Körperbewegung im Wahrnehmungsbereich dieses Systems wird das synthetische Bewußtsein erweckt und beginnt, Klänge zu erzeugen.

Der Klang dringt durch die Luft, gelangt ins Ohr (und in

den Körper) und wandert durch die Übersetzungen des menschlichen Gehörs, durch das Labyrinth des individuellen Gehirns und das verstrickte Interface, das Körper und Geist verbindet, um sich als weitere Bewegung auszudrücken (oder nicht), die einen Kreis vollendet und einen neuen beginnt. Aber die Kreise greifen ineinander über. Sobald man sich als Antwort auf einen Klang des Systems bewegt, modifiziert das System auch schon seine Antwort, um die Bewegungen wiederzugeben, so daß sich Körper und System beständig neu aufeinander einstellen und eine Art Schwingung entsteht.

Die Minimal-Verzögerung dieser Schleife beträgt etwa eine dreißigstel Sekunde. Aber die tatsächliche Verzögerung ist abhängig von den wirklichen Wegen durch Geist, Körper und System. Beide, System und Person, arbeiten wie Prismen, indem sie die Feedback-Strahlen in verschiedene Komponenten brechen, unterschiedlich schnell arbeiten und auf verschiedenen Frequenzen schwingen.

Und genauso wie akustisches Feedback die akustische Natur eines Raums übertreibt, verstärkt die interaktive Feedbackschleife in meinen Installationen besondere Aspekte des Systems und der Person, indem es Resonanzen erzeugt. Diese Resonanzen kann man sich als Akkorde der Interaktion vorstellen, sie sind immer unterschiedlich für jede Kombination von Installation und Individuum. (Dieses Zusammenspiel ist für mich immer wieder eine Überraschung; ich habe manchmal Leute beobachtet, die zusammen mit dem System Laute und Muster geschaffen haben, die ich als intimer Kenner dieser Installation für unmöglich gehalten hätte.) Wegen der umfassenden Komplexität des Systems bestimmen viele unterschiedliche Faktoren seinen Resonanz-Charakter. Diese Faktoren können unterschieden werden als Funktionen des Körpers, des Charakters der Person im Raum, des Klangs und der Art, wie das Sy-



stem selbst antwortet. Jeder Körper hat seine eigene Resonanzfrequenzen, je nach Größe, Gestalt, Gewicht und Muskulatur. Die im interaktiven Programm eingebauten Rhythmen wirken gegen sie oder verstärken sie. (Es ist interessant, eine größere Gruppe von Leuten beim Tanzen zu beobachten, wenn sich jeder bemüht, einen Weg zu finden, um seinen individuellen Körper in Einklang mit den Rhythmen der Musik zu bringen. Jeder bekannte Tanzstil scheint sich aus dem Bedürfnis entwickelt zu haben, den Körper so zu bewegen, daß er sich einem bestimmten Rhythmus anpaßt.)

Die Art der Klänge, die ich für das System benutze, hat eine weniger direkte, dafür umso stärkere Wirkung auf die Interaktion. Menschen empfinden komplexe Klänge als eine äußere, materielle Wirklichkeit und spüren sie körperlich, sie stellen sich vor, einzelne Klangpartikel würden den Raum füllen. Rundere, tiefere Klänge werden empfunden, als wären sie im Körper oder eine Art Ausdehnung des Körpers, etwas, was man eher im Bauch fühlt, in den Organen und Muskeln und nicht auf der Haut. Der Körper reagiert mit unterschiedlichen Beziehungen mit dem System, je nachdem, ob die Klänge innerlich oder äußerlich empfunden werden. Und jede Beziehung hat ihr eigenes Zusammenspiel.

Im harmonischen Spektrum des Systems werden verschiedene Stufen ins Spiel gebracht, da die Teilnehmer in unterschiedlichem Grad mit Reflexionen und spontaner Aktion reagieren. Wenn sie sich vorwiegend intellektuell mit dem System auseinandersetzen, neigen sie dazu, nicht unmittelbar auf das Feedback zu reagieren, denn sie arbeiten eher wie Wissenschaftler, die experimentieren und die Ergebnisse »objektiv« beobachten. Ziel einer solchen Betrachtungsweise ist meist, ein solches System kontrollieren zu lernen, so wie man ein Instrument spielen lernt. In diesem Falle reagiert

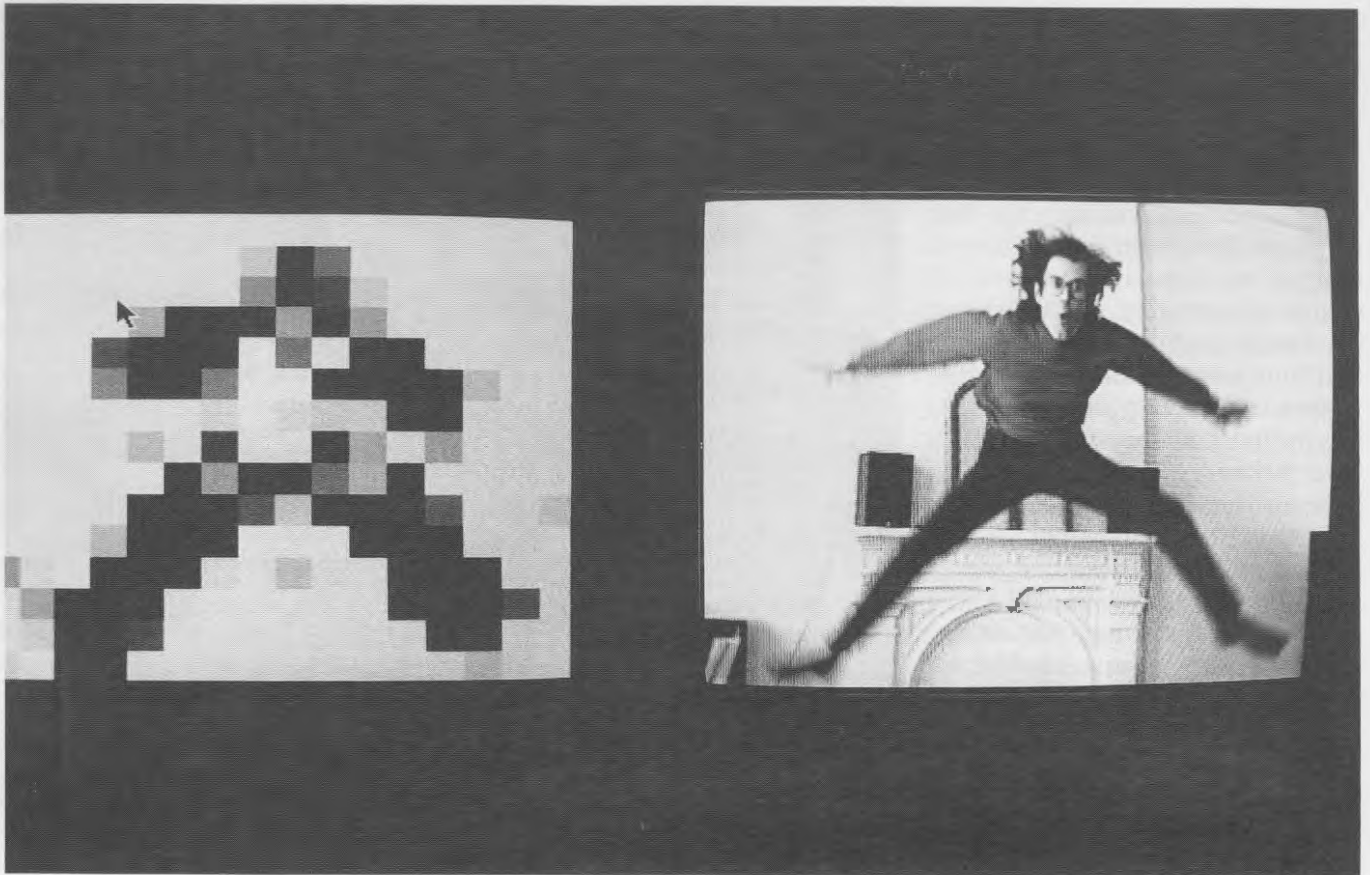


das System, wenn überhaupt, in sehr langsamen Abständen, denn die Leute gestalten ihre Bewegungen in Reaktion auf eine langsame Ansammlung von Erfahrungen. (Der Verstand arbeitet wie ein ganz feiner Filter.)

(Der Körper neigt dazu, die einen rein verstandesmäßigen Zugang haben, mit dem Ziel, Kontrolle über das System zu gewinnen. Der erste Versuch, den solche Leute mit dem System anstellen, ist: »Wenn ich dieselbe Bewegung mache, bekomme ich dann auch denselben Klang?« Bei einigen meiner einfacheren interaktiven Stücke bekommt man den gleichen Klang für genau die gleiche Bewegung. Bei solch einem System entsteht ein Muster. Es wird eine Geste gemacht, als ob eine Frage gestellt würde, und die Antwort des Systems wird registriert. Eine zweite Geste wird gemacht, eine dritte, wieder als Fragen, und jedesmal ist die Antwort die gleiche wie beim erstenmal. Dann kommt eine Pause, und das Gesicht verändert sich, als wollte es sagen: »Aha, ich verstehe...«, und eine vierte Geste wird gemacht, jetzt aber nicht mehr als Frage, sondern als Aussage, fast wie ein Befehl. Und so, fast unmerklich, wird ein neuer Klang geschaffen, und der Einstellungs-

wandel zeigt sich in einer ganz feinen, aber merklichen Änderung in der Haltung des Körpers und der Dynamik der Gesten.)

Wenn Leute aber eine spontane Reaktion auf die Musik des Systems zulassen, dann werden sie selbst von der Installation gespielt. Diese Herangehensweise setzt voraus, daß man die eigene Vorstellungskraft ins Spiel bringt, sich öffnet und der Musik gestattet, direkt durch den eigenen Körper zu kommunizieren, ohne daß viel überlegt wird, so daß die Feedbackschleife immer enger wird. Bei Leuten, die das System in dieser Weise benutzen, bemerke ich oft, wie sie auf Empfangsirrtümer der System-Software reagieren, indem sie unbewußt ihre beabsichtigten Gesten korrigieren und in Einklang mit dem antwortenden System bringen. Sie kehren das Verhältnis um und unterstellen die Ausdrücke ihrer eigenen Intelligenz dem System. (Wenn Leute unbewußt in Einklang mit einem Rhythmus fallen, drücken sie oft Erstaunen aus über die Fähigkeit des Systems, sich ihren Bewegungen anzupassen.) Natürlich sind dies grobe Verallgemeinerungen, und jeder, der in das System einsteigt, hat in unterschiedlichem Ausmaß eine Mischung aus allen Herangehensweisen. Mein



Idealszenario ist eines, bei dem Haltungen ausgeglichen sind und das Ergebnis eine »große Bandbreite« an Interaktion ist.

Wenn ich die Programme schreibe, die das Reaktionsverhalten des Systems definieren, arbeite ich wie ein Klangingenieur bei einem Konzert mit Verstärkern, der Filter benutzt und andere verzerrende Geräte, um das Feedback des Klangsystems zu steuern. Meine Ziele sind jedoch andere. Der Ingenieur strebt danach, sich überlagernde Frequenzen verschwinden zu lassen, damit die Vorstellung nicht gestört wird und Rückkopplungseffekte eliminiert werden. Ich filtere und verzerre, um Rückkoppelungen zu fördern, um versteckte physiologische und psychologische Resonanzen des Körpers und Geistes in einer physischen Interaktion zu verstärken.

Die Musik des »Very Nervous System« liegt nicht so sehr in den Klängen, die man hört, sondern im Zusammenspiel der Resonanzen, die man fühlt, wenn man die Arbeit mit dem Körper erfährt.



Übersetzung: Linda Drasal

Auf dem Weg zur Hypernetzwerk- Gesellschaft

Izumi Aizu

In Japan laufen die Drähte heiß. Ein engmaschiges Computer-Netzwerk zieht sich über die Inselnation. Izumi Aizu prophezeit die Aufhebung lokaler, sozialer, kultureller Hürden und der Trennung von Konsumenten und Produzenten.

Ich will mich zunächst einmal vorstellen. Nach meiner Schulzeit wurde ich Drucker in einem kleinen Unternehmen in Tokio, das auch internationale Werbe- und Verkaufskampagnen japanischer Firmen betreute. Als Anfang der achtziger Jahre einige Personal-Computer ins Büro kamen, war ich sehr überrascht, wie kompliziert und unverständlich die Handbücher waren. Das weckte mein Interesse für technische Dokumentation; ich begann Handbücher zu schreiben, die auf die Kunden zugeschnitten sind. Ich kündigte und wurde Freiberufler.

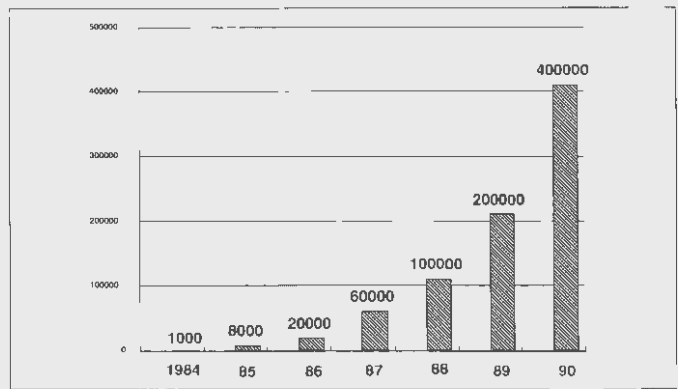
Dann beging ich einen großen Fehler: Bei einem Besuch in den USA hatte ich einen kleinen, tragbaren PC samt Modem dabei. Schon in Japan nutzte ich eifrig die US-Netzwerke, aber auf einmal wurde ich richtig süchtig. Auf dem USA-Trip erkannte ich, daß der Computer zum »Rückgrat« unserer neuen Kommunikationssysteme werden würde.

Mein nächstes Ziel ist die Gründung eines kleinen »Think Tank« mit dem Namen »Institut für Netzwerk-Design«. Meine Mitarbeiter und ich wollen die Vernetzung der Menschheit mit Hilfe der elektronischen Netzwerke populär machen. An diesem Ziel arbeite ich nun schon seit fünf Jahren.

Meine Karriere läßt sich vielleicht unter dem Begriff »Kommunikation« zusammenfassen, genauer: menschliche Kommunikation. Ich bin weder Wissenschaftler noch Techniker, nur ein ganz gewöhnlicher Mensch, der sich dafür interessiert, wohin die Entwicklung der Technologie führt – aus der Sicht des Anwenders.

Doch bevor wir uns mit dem Blick in die Zukunft beschäftigen, lohnt ein kurzer Blick in die Vergangenheit. Wie Sie auf dem Bild 1 sehen, wuchs in Japan die Anzahl der Anwender von PC-Netzwerken ganz erheblich in den letzten fünf Jahren. Es gibt rund zehn große kommerzielle Netzwerke, die das ganze Land verbinden. Das größte, PC-VAN der Firma NEC, zählt mehr als 200 000 Teilnehmer. Die Konkurrenz, das NIFTY von Fujitsu, nutzen fast ebenso viele Menschen.

Aber auch die kleinen sind populär: die sogenannten »Grass-Root«-Netzwerke, also Netzwerke auf lokaler Ebene, meist von Hobbyisten betrieben. Damit sind über das ganze Land verstreut mehr als 3000 Aktive verbunden. Die regionalen Netze sind eine Be-



PC-Netzwerk-Anwender in Japan, 1984 bis 1990

sonderheit der japanischen Netzwerk-Szene. Viele Bezirksregierungen, Städte und Gemeinden machen sich genau wie der private Sektor für den Einsatz der Kommunikationsnetze stark.

Ein gutes Beispiel für ein solches Regional-Netz ist Coara. Im Jahre 1985 begann Coara als ein winziges, lokales Netz in der Stadt Oita, die etwa 500 Meilen westlich von Tokio liegt. Am Anfang hatte Coara gerade mal dreißig Anwender. Doch die ersten Coara-Fans entdeckten, daß der Austausch mit anderen Menschen einfach mehr Spaß macht, als nur einfach Daten zwischen den PCs hin- und herzuschicken. Das klingt wie eine Binsenweisheit, aber damals war das eine wichtige Erkenntnis. Denn zu dieser Zeit wollten die Netzwerk-Anbieter in Japan ihren Kunden Information als Ware verkaufen. Die Netzwerk-Fans entwickelten jedoch ihre eigene »Netzwerk-Philosophie«: Hauptanliegen ist die Kommunikation der Menschen miteinander, nicht das Geschäft mit Information. Diese Wünsche der Anwender prägten die Entwicklung der öffentlichen wie der kommerziellen Netzwerke in Japan. Der Industrie-Riese Matsushita und die Recruit Corporation sind nur zwei der vielen großen Unternehmen, die Coara untersucht haben. Die Firmen übertrugen viele Design-Aspekte in ihre eigenen Netzwerke.

Die »glokale« Gesellschaft

Bis zum Jahre 1984 arbeiteten nur wenige Japaner mit PC-Netzwerken oder elektronischen Briefkästen (oder Mailboxen). Damals bestand die Fan-Gemeinde aus eingefleischten Computer-Freaks oder Menschen wie ich, die von diesem neuen Kommunikationsmedium durch Freunde im Ausland erfahren haben. Als die japanische Telefongesellschaft NTT (Nippon Telegraph & Telephone) im April 1985 privatisiert wurde, fiel auch ihr Monopol auf Telefonleitungen und Datenkommunikation. Die Folge: PC-Netzwerke verließen den Hobby-Bereich und wurden erwachsen.

Die Computer-Netze avancierten zum Lieblingsthema der PC-Szene in Japan. Die Netzwerke erlauben den Menschen, unabhängig von Raum und Zeit neue Leute zu treffen und neue Freundschaften zu schließen. Aber sie befreien die Anwender auch von ihren sozialen Bindungen wie Beruf, Firmen-Zugehö-

rigkeit und Geschlecht und so weiter. Diese neue Netzwerk-Gemeinde bestand nicht nur aus Computer-Fans. Auch Leute, die mit Computern nichts am Hut haben, begeisterten sich schnell für die Netze, weil sie dieses Medium als wertvolles und mächtiges Werkzeug erkannten, das ihnen eine ganz neue Dimension menschlicher Beziehungen eröffnet. Die Folge dieser Entwicklung: Unternehmen, die Netzwerke einsetzen, bewerten die wirkliche Kommunikation der Teilnehmer höher als den reinen Informationsaustausch.

Obwohl Coara ein regionales Netzwerk ist, tummeln sich darin interessanterweise viel mehr Teilnehmer von außerhalb. Sogar Anwender aus Holland, der Schweiz, Korea und den USA nutzen Coara.

Es herrscht ein ausgeprägtes Gruppenbewußtsein unter den Teilnehmern. Während eines Besuchs im letzten Jahr verglich der bekannte Zukunftsforscher Howard Rheingold (Autor von »Tools for Thought, Cognitive Connection« und Chefredakteur des »Whole Earth Review«) Coara mit dem berühmten San Francisco Netzwerk WELL (Whole Earth eLEctronic Link): Beide Netze, meinte Rheingold, seien eine »virtuelle Gesellschaft«, eine »Gesellschaft der Gedanken« (»community of minds«).

Aus WELL, Coara und vielen weiteren Netzwerken auf der ganzen Welt formen sich neue Gesellschaften, die sogar geographische und soziale Grenzen überwinden. Coara ist ein Beispiel der neuen »glokalen« (global + lokal) Gesellschaft.

Das wichtigste Ziel meiner Arbeit ist, den Austausch japanischer Netze mit Netzwerken anderer Länder voranzubringen. In letzter Zeit findet vermehrt ein Austausch zwischen koreanischen und japanischen Netzwerken statt. Schließlich müssen wir unsere Nachbarn kennenlernen. Das erste Seminar über »koreanisch-japanische PC-Netzwerke« fand im November 1989 in Seoul statt. Mittlerweile wurde daraus ein beliebter, jährlicher Treffpunkt für japanische und koreanische Netzwerk-Anwender. In den letzten zwei Jahren begannen wir, weltweite Bindungen zu knüpfen: Zwischen Japan und Ländern wie den USA, Korea, Singapur, Australien und der UdSSR. So wurden die Netzwerke in Japan zu globalen Kommunikationspfaden – trotz der Sprachbarriere.

Ein kleines Projekt vom Herbst 1989 hat mein Denken sehr beeinflusst: »Unsere Vision der Zukunft« (»Our Future Vision«), ein Pilotprojekt, gesponsort vom japanischen Elektronikunternehmen NEC. Es ging darum, das Potential der weltweiten Kommunikation via Computer auszuloten. Mit Hilfe vorhandener globaler Netzwerk-Verbindungen befragten wir Anwender in der ganzen Welt, wie sie sich die Zukunft der Kommunikation vorstellen. In weniger als vier Wochen erreichten uns elektronische Briefe aus über dreizehn Ländern, aus Amerika, Asien und Europa (sogar aus der UdSSR), die alle den gleichen Tenor hatten: Die neuen Kommunikationssysteme sollen für normale Menschen zugänglich sein und uns allen ermöglichen, enger in einer friedlichen Welt miteinander zu leben.

Multi-Media auf der Datenautobahn

Computer-Netzwerke sind asynchron. Die Menschen diskutieren miteinander, tauschen Wissen aus und stellen so richtige »Online-Gesellschaften« her. Doch die Entwicklung bleibt nicht stehen. Schon rückt die »Multi-Media«-Kommunikation in das Zentrum unserer Aufmerksamkeit. In weniger als fünf Jahren steht einigen von uns die schnelle Datenautobahn B-ISDN zur Verfügung. Auf diesem Träger können enorme Datenmengen transportiert werden, sogar bewegte Video-Bilder. Im Moment wird in Japan gerade intensiv die Kommunikation von Arbeitsteams über bewegte Video-Bilder erforscht. Das japanische Forschungsministerium MITI (Ministry of International Trade and Industry) leitet das »Friend 21 Project«, ein Joint-Venture von Regierung und Industrie. Die japanische Telefongesellschaft NTT will bis zum Jahre 2005 den Service »VI&P« (Visual, Intelligent and Personal) vorstellen, der eine solche Verknüpfung ermöglicht.



Eines der interessantesten Felder der Computer-Technologie ist »Virtual Reality« (VR). Das junge Forschungs-Unternehmen ATR (Advanced Telecommunications Research) befaßt sich damit, VR-Technologie in ihre sogenannte »Kommunikation mit realistischen Eindrücken« zu integrieren. An ATR sind mehr als 120 Kommunikations- und Computer-Unternehmen beteiligt, und das jährliche Forschungsbudget liegt bei 50 Millionen Dollar, wobei die VR-Forschung jedes Jahr einen höheren Anteil zugesprochen bekommt. Auch ich möchte die Auswirkung der VR-Technik in Kombination mit Netzwerken näher studieren. Zu diesem Zweck planen wir, im Jahre 1993 das »Institut für Hypernetzwerk-Gesellschaft« zu gründen. Das neue Projekt wird sowohl vom MITI und dem Gemeinderat von Oita (der Coara-Heimat) gesponsert. Das Institut soll ein Forschungszentrum werden, in dem die soziologischen und technischen Folgen der neuen Technologie untersucht werden. Auf diese Weise wollen wir soweit wie möglich den Weg für die Umwälzungen der nächsten technologischen Revolution ebnen. Wir haben drei Forschungsgebiete im Auge:
– ein kleines Gruppen-Netzwerk, in dem wir »Groupware«-Technologie testen und auswerten möchten

(unter »Groupware« versteht man Software und Hardware, die Arbeitsgruppen die Verständigung und Dokumentation ihrer Projekte erleichtert),
 – ein Netzwerk auf Gemeinde-Basis, um die Auswirkungen neuer Netzwerke auf das tägliche Leben einer Gemeinde zu studieren,
 – ein Netzwerk mit »Virtual Reality«. Sowohl ein lokales wie ein globales Netzwerk soll mit VR-Techniken realisiert werden.

Die individuelle Hyperzeitung

Mein Kollege Toru Ono und ich prägten vor etwa einem Jahr den Begriff »Hypernetzwerk«, der das Zusammenwachsen von »Multi-Media« (Bilder, Sprache, Text, Daten und so weiter), VR-Technologie und computergestützten Kommunikationssysteme beschreibt. Darunter verstehen wir eine um Bild- und Sprach-Fähigkeiten erweiterte Form der heute existierenden globalen Computer-Netze.

Unsere Auffassung des Hypernetzwerks deckt sich weitgehend mit der Doug Engelbarts, des amerikanischen Wissenschaftlers, der in seiner Pionierarbeit auf diesem Gebiet folgende These vertritt: Das Hypernetzwerk ermöglicht vollständige Kommunikation zwischen Gruppen, die die komplexen Probleme der modernen Welt lösen wollen. Dazu müssen die menschlichen Fähigkeiten mit Hilfe der Computer-Werkzeuge und mit mehr Informationen ergänzt werden. Die dazu nötigen Werkzeuge nennen wir »Hypertext«, »Hypermedia« oder »Groupware«. Sie sollen uns helfen, effektiver miteinander zu kommunizieren, denn kreative Einfälle entstehen durch den Austausch mit anderen Menschen.

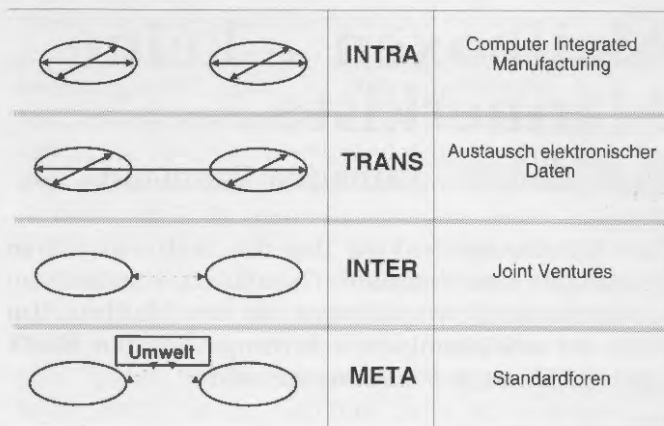
Dank der neuen Kommunikationstechniken können wir Video-Bilder genauso leicht bearbeiten, wie wir es heute schon mit Texten machen. Die Kombination aus Text-Daten und bildlichen Nachrichten schafft eine »reichere« Kommunikationsumgebung.

Wir können unsere eigene »Tagesschau« zusammenstellen und auf die Netze kopieren. Besonders in Krisensituationen, wenn Länder sich nationalistisch einigeln, wäre der Austausch individueller Nachrichten ein wirkungsvolles Mittel gegen politische oder kommerzielle Falschmeldungen. Ich wünschte, wir hätten diese Hypernetzwerk-Werkzeuge schon gehabt zur Zeit des Massakers am »Tiananmen Square« (dem Platz des himmlischen Friedens in Peking), während der rumänischen Revolution oder vor Beginn des Golf-Kriegs.

Das Netz der »Prosumenten«

Die soziale Seite der Vernetzung hat der französische »Think Tank« PROMETHEE unter der Leitung von Dr. Albert Bressand näher erforscht. Er nennt seine Vision »NetWorld« – eine sozio-ökono-techo-politische Vision des 21. Jahrhunderts.

Die Definition des Begriffs »Netzwerk« ist längst nicht so einfach, wie sie klingt. Bild 2 stellt eine mögliche Definition dar. »Intra-Netzwerke« und »Trans-



Netzwerke« sind meistens auf elektronischer Basis, während »Inter-« und »Meta-Netzwerke« soziale und organisatorische Aspekte beschreiben. Es geht darum, die technologischen Netzwerke mit den passenden sozialen und organisatorischen Netzwerken in Einklang zu bringen. Denn diese sozialen und organisatorischen Systeme brauchen die richtige Technologie, um funktionieren zu können.

In der Wirtschaft weiß jeder, daß weder Aktien- noch Devisenmärkte ohne Computer funktionieren. Die Wirtschaft ist immer engermaschiger vernetzt, und die elektronischen Netzwerke sind fester Bestandteil der wirtschaftlichen Infrastruktur.

Organisationen, große Unternehmen und sogar Behörden entwickeln sich immer mehr zu anpassungsfähigen Einheiten. Immerhin sind sie gezwungen, so flexibel wie möglich zu sein, um sich den immer schnelleren Änderungen der Gesellschaft besser anpassen zu können. Albert Bressand schlägt eine neue Form der Produktivität vor, die er »Massen-Maßanfertigung« nennt. Sie soll die für die Hersteller relevanten Vorteile der Massenproduktion verbinden mit den für die Verbraucher relevanten Vorzügen der individuell angepaßten Produktion. Dafür wären die globalen Netzwerke der Zukunft ein wichtiges Produktionshilfsmittel. Bislang finden sich Beispiele für diese »Massen-Maßanfertigung« nur auf den Finanzmärkten und der Reisebranche (bei weltweiter Flugticket-Buchung). In nicht allzu ferner Zukunft könnte es sie auch im Konsumgüterbereich geben, und wir könnten maßgeschneiderte Autos bestellen.

Stellen Sie sich ein Computer-Terminal vor, das über »Super-B-ISDN« mit moderner 3D-Grafik und virtuellen Systemen in Echtzeit mit mächtigen CAD/CAM-Anwendungen verbunden ist, dazu noch Computer Integrated Manufacturing (CIM) und elektronische Kanban-Produktion (»Just-in-time«-Produktionssystem). Damit hätten wir den Grundstein für Massen-Maßanfertigung – und für eine Welt der »Prosumenten«: Konsumenten, die Produktion, Bestellung und Design ihrer gewünschten Produkte selbst bestimmen und steuern.

Die Welt der Zukunft wird engermaschig vernetzt sein. Jeder Mitspieler kann seinen individuellen Standpunkt genießen, ohne die eigene Kultur aufzugeben.

Übersetzung: Klaus Madzia

Mailboxen – keine Männerkiste

Rena Tangens, Katharina Baumann

Das Schubladendenken, das die Welt einteilt in männliche und weibliche Domänen, scheitert im Computerbereich spätestens an der Mailbox. Im Netz der »elektronischen Zeitung« können KURT oder KATE auch Pseudonyme sein.

Ein »Und« kann alles miteinander verbinden: »Frauen und Computer« ebenso wie »relative Ordnung und intergalaktisches Chaos«, »Zen und die Kunst ein Motorrad zu warten« oder »Existenzialismus und der Wille zur Macht«. Bei den Frauen und den Computern gibt es indes einige wirkliche Berührungspunkte.

Was eine Mailbox ist ...

... weiß noch nicht jeder. Hier zum Einstieg eine kurze Erklärung. Wer die Mailboxen bereits kennt, kann diesen Absatz überspringen.

Eine Mailbox ist ein (meist) ganz normaler Computer, der mit Hilfe eines Modems⁽¹⁾ mit der Telefonleitung verbunden ist. Vierundzwanzig Stunden am Tag hängt der Rechner am Telefon und nimmt Anrufe von anderen völlig normalen Rechnern entgegen. Er fungiert – wie das englische Wort »mailbox«⁽²⁾ schon nahelegt, als ein elektronisches Postfach. Hier können Nachrichten empfangen und versendet werden, für andere hinterlegt, weitergeleitet und archiviert werden. Nicht nur innerhalb der lokalen Mailbox (also dem Computer, den ich selbst zum Telefonortstarif anrufe), sondern auch überregional oder sogar international. Allerdings nur, wenn die Mailbox Teil eines entsprechenden Netzwerks ist. Die einzelnen Rechnerknoten rufen sich nachts vollautomatisch über das normale Telefon an und tauschen Mitteilungen untereinander aus. Auf diese Weise wird meine Nachricht weitergeleitet, bis sie die gewünschte Empfänger-Mailbox erreicht.

Neben der privaten Post gibt es die öffentlichen Nachrichten, die in sogenannten »Brettern« (bulletin boards) abgelegt werden und dort von allen Teilnehmern gelesen werden können. Diesen Teil einer Mailbox könnte man mit einer überregionalen Tageszeitung vergleichen. Es finden sich Politik- und Wirtschaftsseite, Feuilleton, Regional- und Kleinanzeigen sowie ein vielfältiges Fachzeitschriftenangebot, zum Beispiel aus dem Bereich Umweltschutz oder Telekommunikation. Aus diesem Angebot kann sich jeder eine Wunschzeitung zusammenstellen; Seiten und Themen, die nicht interessieren, braucht niemand zu lesen oder zu »kaufen«. Eine Mailbox ist »eine Zeitung ohne Altpapier«, schrieb Hans-Heinrich Pardey in der Frankfurter Allgemeinen Zeitung.

Der wichtigste Unterschied sowohl zur Zeitung als auch zu fast allen sogenannten Neuen Medien ist die

echte Zweiwegkommunikation via Mailbox. Hier ist im Medium die Trennung zwischen Konsumenten und Produzenten aufgehoben. Die Mailbox gestattet allen Teilnehmern, jederzeit selbst zu veröffentlichen, zu berichtigen, zu kommentieren. Nicht in einer Leserbriefecke, sondern dort, wo der ursprüngliche Artikel steht. Wobei bei aller paradiesischen Freizügigkeit immer bedacht werden muß, daß die Qualität der Kommunikation hier natürlich entscheidend von den Benutzern abhängt.



Zudem gibt es Konferenzsysteme, bei denen mehrere Telefonleitungen gleichzeitig zur Verfügung stehen. Hier treffen sich die Teilnehmer zum direkten Dialog »online«.

Ein Mailbox-Netzwerk ist gleichzeitig Treffpunkt, Informationssystem und Publikationsmedium für alle. Zumindest alle, die Zugriff auf einen Computer, ein Modem und ein Telefon haben. Das sind zur Zeit noch nicht viele. Die Mailbox ist leider immer noch weitgehend ein Insidermedium an der Schwelle zur weitgehenden Verbreitung durch alle Bevölkerungsschichten. Mit öffentlich zugänglichen Terminals käme ein Mailbox-Netzwerk dem »socially beneficial information processor« (Marianne Brun in ihrem Buch »Designing Society«) schon recht nahe. Nur gibt es bei dem Mailbox-Netz kein maschinelles Superhirn, sondern einen lebendigen Organismus, der aus einer Vielzahl autonomer menschlicher Einzelwesen besteht. Alle Teilnehmer bilden in ihrer Gesamtheit das kollektive Wissen und Bewußtsein des Netzwerkes – die Matrix: »Frag doch das Netz ...«

Das elektronische Caféhaus

Der Anruf bei einer guten Mailbox entspricht schon fast einem Caféhausbesuch. Ich kann in Ruhe meine Post oder eine der vielen ausliegenden Zeitungen lesen, Geschäfte abschließen, mit Freunden plaudern. Ich lausche ein wenig der Diskussion am Nebentisch,

setze mich auch mal woanders hin und lerne neue Leute kennen. Ich kann mich und meine Ideen präsentieren. Natürlich finden sich auch Leute, die nur hierher kommen, um über Rechner zu fachsimpeln, Flugblätter zu verteilen oder schnell ein Veranstaltungsplakat aufzuhängen.

Was macht nun ein gutes Caféhaus aus? Die Leute, die dort verkehren. Sie kommen, weil sie sich unter dem Caféhaus-Publikum wohl fühlen. Aber was veranlaßt dieses Publikum, gerade dort hinzugehen? Bevor wir in eine endlose Schleife geraten, ein Erklärungsversuch: Neben offensichtlichen Faktoren wie gute Lage, eigener Stil und freundlichem Service sind es die passende Möblierung und die angenehme Atmosphäre. Und neben diesem »Rahmenbau« bestimmt vor allem die Präsenz der Menschen, die dort arbeiten, das Klima: das Caféhausteam.

Sie sollten (idealerweise) mit Freude und Interesse dabei sein, freundlich, kommunikativ, aber ohne Hang zum Alleinunterhalter, aufmerksam, immer ansprechbar und diskret sein; die Gäste kennen, sich auch mal mit an den Tisch setzen, mitdenken, auf Wünsche eingehen, ohne aber den Charakter des Lokals umzukrempeln, stets den Überblick behalten und, und, und... Nun beginnen wir uns dem Thema von der anderen Seite zu nähern. Bevor Sie weiterlesen, bestellen Sie sich doch noch eine Melange...

Frauen in den Informationstechnologien

Es ist gewiß notwendig, hier Ada Lovelace, Grace M. Hopper und die Eniac-Girls zu erwähnen. Doch wollen wir keinen historischen Rückblick oder Gemeinplätze à la »Alle Männer sind...« formulieren. Die Unterschiede zwischen Einzelpersonen innerhalb jeder Gruppe sind größer als die Unterschiede zwischen Frauen und Männern allgemein. Und für jede Beobachtung läßt sich ein Gegenbeispiel finden⁽³⁾. Trotzdem gibt es einige Beobachtungen, die Strukturen erkennen lassen. Im folgenden will ich berichten, was mir auffiel, als ich merkte, daß ich als »Haeckse«⁽⁴⁾ eine Ausnahmeerscheinung bin.

Feminines Computer Handling

»Feminines Computer Handling« wählten wir als Titel für eine Veranstaltung auf dem Hamburger Chaos Communication Congress. Es ging um den anderen Umgang von Frauen mit Rechnern und Informationstechnologien⁽⁵⁾.

– Mehr Männer als Frauen besitzen einen eigenen Computer. Frauen verfügen im Schnitt über geringere finanzielle Mittel – da wird die Anschaffung eines Computers meist erst dann in Angriff genommen, wenn eine sinnvolle Anwendung auf der Tagesordnung steht. Der eigene Computer ist eine Voraussetzung für einen guten Einstieg und Experimentierfreude – ganz ohne Angst, jemandem anderen etwas zerstören zu können.

– Frauen beschäftigen sich selten mit der gleichen Ausschließlichkeit mit dem Rechner wie Männer; sie

erreichen selten den Zustand der inneren und äußeren Verwahrlosung, in den echte Freaks manchmal fallen. Welcher Mann ist aber schon bereit, sich neben seinen sonstigen Interessen um Haushalt oder Kinder zu kümmern, während seine Frau ganz in ihrem Beruf aufgeht? Oder welche Frau kann damit rechnen, daß ihr jemand etwas zu essen hinstellt, wenn sie mal nächtelang am Rechner sitzt und sich einem kniffligen Programmierproblem widmet?

– Frauen gehen zielorientierter mit Technik um. Sie basteln zum Beispiel nicht ewig an einem Werkzeug (das später unter Umständen nie wieder benutzt wird), wenn sie das Ergebnis auch mit weniger Aufwand erreichen können. Diese Lösung mag provisorisch und holprig, oft aber auch wesentlich eleganter sein. Manchmal liegt die Lösung sogar außerhalb des Rechners.

– Frauen vertrauen weniger auf ihr Können. Fragt man eine Frau nach ihren EDV-Kenntnissen, kann man folgende Antworten hören: »Ich kenne mich nicht so gut aus, ich kann nur Graphik« oder »nur Textverarbeitung« oder »nur Unix«. Das ist nicht nur Bescheidenheit, Vermeidung von Wettbewerb oder Koketterie, sondern schlicht und einfach das Bewußtsein, daß es sich um einen begrenzten Teilbereich handelt und nicht die Lösung aller Dinge.

– Neben all diesen positiven Seiten bauen Frauen sich auch selbst Hindernisse auf. Mangelnde Experimentierfreude, die Angst, Geräte durch das Drücken falscher Tasten in die Ewigen Technikgründe zu befördern, und die Furcht, durch »dumme Fragen« als allzu weiblich zu erscheinen, machen es Frauen in einer zunehmend von Technik bestimmten Welt nicht gerade leichter.

– Während Männer fast erotische Beziehungen zu ihrem Spielzeug und Arbeitsinstrument entwickeln können, ist Frauen meistens die Beziehung zu und zwischen anderen Menschen wichtiger. Das könnte sich auf Softwaregestaltung und Programmierstil auswirken. Hier ist ein zukünftiges Aufgabenfeld für Frauen, die aktiv an der Entwicklung teilnehmen wollen. »Einfluß nehmen« meine ich wirklich im weitesten Sinne, denn Hard- und Softwaregestaltung greift in alle möglichen Lebensbereiche ein. Es gibt Männer, deren Phantasie in Sachen Technikgestaltung von und für Frauen nicht weiter reicht, als an pinkfarbene Monitore zu denken. Wir sehen, ...

Frauen und Macht

... es gibt immer noch starre Vorstellungen davon, was »weiblich« ist. Um sich von »Kinder, Küche, Kirche« freizumachen und den Lebensmittelpunkt im Beruf zu finden, mußten Frauen zu allen Zeiten einen außergewöhnlich starken Willen haben. Virginia Woolfs Gedankenexperiment, wie es William Shakespeares hochbegabter Schwester ergangen wäre, sagt mehr als jede soziologische Untersuchung.

Um ihre Beteiligung an der Technik mitgestalten zu können, müssen Frauen zunächst auf die Position mit entsprechendem Einfluß kommen. Hier gibt's

nichts geschenkt: Viele Errungenschaften, die wir als selbstverständlich ansehen, sind noch gar nicht so alt – sei es das Wahlrecht für Frauen oder das Recht, studieren zu dürfen.

Um in Spitzenpositionen in Wirtschaft, Wissenschaft oder Politik erfolgreich zu sein, müssen Frauen hart und rücksichtslos sein wie ein Mann, aussehen wie ein junges Mädchen, sich geben wie eine Dame (hin und wieder auch umgekehrt) und arbeiten wie ein Pferd. Die wenigen, die diese Anforderungen unter einen Hut bekommen, sind vielfältigen Vorwürfen ausgesetzt, vor allem aber dem, »unweiblich« zu sein. Dieser Typ »unabhängige, erfolgreiche, gutaussehende Karrierefrau« wird von einigen Frauenzeitschriften propagiert, nicht ohne vorsorglich auf die vielen Möglichkeiten des privaten Unglücklichseins hinzuweisen. Von den wenigen Frauen, die sich so bisher tatsächlich einen Platz im Zentrum der Macht erkämpft haben, können wir nicht auf das Verhalten der Frauen überhaupt schließen. Während die Quotierung im öffentlichen Dienst beim Posten- und Beförderungsverteilen behindert wird und somit Unmut erregt, sucht das Management hier und da schon händeringend nach weiblichen Bewerbern.

Die Komplexität einer vernetzten Gesellschaft wird neue Arbeits- und Lebensbedingungen schaffen und andere Umgangsformen erfordern. Wie Frauen mit Macht und Technik umgehen, wenn sie die Chance haben, sie selbst zu gestalten, ist offen. Dazu zum Abschluß noch ein Zitat: »Solange es nicht ebensoviele absolut unfähige Frauen wie absolut unfähige Männer in höchsten Positionen gibt, ist die Gleichberechtigung nicht erreicht« (sagte Ilona Staller, genannt La Cicciolina, Abgeordnete im italienischen Parlament).

Frauen in Mailboxen

Wieviele Frauen bereits in Mailboxen aktiv sind, läßt sich nicht so ohne weiteres feststellen. Hinter Usernamen wie D.LOEBNER oder B.THOENS können sowohl Dietrich und Barbara als auch Doris und Bernd stecken. Aber auch bei U.WE, MARION oder KATE kann ich beim Geschlecht nicht sicher sein.

So erhielt ich von einem Teilnehmer aus Hamburg nach drei Monaten eingehender Korrespondenz über das Netzwerk und der Frage, ob RENA ein Pseudonym sei (Antwort: »Nein, ich heiße wirklich so«), plötzlich folgende Nachricht: »Du bist ja 'ne Frau!!!«. Was wieder zeigt, daß auch bei aufgeschlossenen Menschen der Normalfall der männliche ist.

Paradoxerweise hat diese Erwartungshaltung zunächst ihre positiven Nebeneffekte: Bei vermuteten männlichen Gesprächspartnern bleibt man doch eher beim Diskussionsthema und unterläßt die störenden Rückfragen nach Haarfarbe, Maßen und Familienstand. »Anmache« gibt es – ganz wie im täglichen Leben – auch in der Mailbox. Aber ich kann mir auch ganz in Ruhe meine Reaktion überlegen (ANTWORTEN, IGNORIEREN oder LOESCHEN?), ohne unangenehme Folgen.

Für Vorurteile gibt es neben dem (selbst-) gewählten Usernamen und der bevorzugten Mailbox wenig Anhaltspunkte. Ich kann nicht sehen, ob mein Gegenüber weiblich oder männlich, alt oder jung, Rugbyspieler oder Rollstuhlfahrer ist. Ich weiß nicht, ob er oder sie einen Irokesenschnitt, Ohrring oder eine Krawatte trägt. Von der Person kann ich nur aufgrund des geschriebenen Dialogs eine Vorstellung entwickeln. Der spätere Realitätsabgleich ist immer spannend.

Weiterhin brauche ich für die Kommunikation in einer Mailbox keine laute Stimme. Ich kann meine Beiträge einem großen Personenkreis zugänglich machen, kann dabei in Ruhe formulieren und werde nicht unterbrochen (Frauen werden in öffentlichen Diskussionen häufiger übergangen oder unterbrochen – und wehren sich oft nicht dagegen). Ich kann mich jederzeit zu dem Beitrag eines oder einer anderen äußern und kann mich entscheiden, ob ich das öffentlich in einem Brett oder privat via Brief will.

Nach kurzer Lernzeit ist die Bedienung eines Mailbox-Programms übrigens erheblich einfacher als etwa die einer ausgewachsenen Textverarbeitung. Im Zerberus-Mailbox-Programm kann ich zum Beispiel mit den fünf Befehlen BRETT, INHALT, LESEN, SENDEN und ENDE wesentlich mehr mitteilen als nur »Hello world«. Im Vordergrund steht nicht die Technik, sondern der Inhalt. Immer mehr Frauen nutzen das Medium als Teilnehmer und zunehmend auch als Systembetreiber, als offenes Forum für Diskussion; als Informationsquelle zu allem, was sie interessiert. Es finden sich keineswegs nur die klassischen Frauenthemen – die laufen eher am Rande mit –, sondern die ganze Bandbreite von Musik bis Politik über Programmierfragen bis Umweltschutz, Witzecke und Wissenschaft. Das Exotenimage, das Frauen in der Mailbox-Szene genießen (oder auch nicht genießen), muß revidiert werden. Es lesen und schreiben hier und jetzt schon viel mehr, als »man« denkt.

Sprache und Zeichen :-)

Schreiben und Sprache werden bei den anderen »Neuen Medien« eher vernachlässigt, in der Mailbox kommt ihnen zentrale Bedeutung zu. In einer Mailbox gilt es, vorsichtiger zu formulieren als anderswo. Dieselbe Bemerkung, die direkt zu jemandem gesagt, amüsiert aufgenommen wird, und durch Mimik, Gestik und den Klang der Stimme interpretiert wird, könnte als schriftliche Nachricht in einer Mailbox einen heftigen Streit auslösen. Meist beruht dies auf einem Mißverständnis und dem Unwissen, wieviele Bedeutungen schon ein einfacher Aussagesatz mit sich tragen kann. Auch echte Differenzen können in der Mailbox glücklicherweise ohne Gewaltanwendung ausgetragen werden. Nur: Auch Streitkultur will gelernt werden.

Kommunikative Fähigkeiten wie aktives Zuhören und Nachfragen sind bei der Kommunikation via Telefon und Computer noch mehr als sonst gefordert. Diese Eigenschaften, die bisher traditionell eher bei

Frauen gefördert worden sind, könnten so endlich allgemeine gesellschaftliche Anerkennung erhalten.

Bei der Kommunikation ohne Blickkontakt sollte ich nie vergessen, daß auf der anderen Seite auch ein Mensch sitzt, der vielleicht gerade im Programmierstreß, frisch verliebt oder im Clinch mit seinem Vermieter ist.

In Ermangelung von Ton und Bild weichen die Teilnehmer oft auf Comicsprache aus. So wird auf den Zeilen zwischen den Zeilen noch etwas mitgeteilt. Die eckigen Klammern ersetzen die Sprechblase: < staun >, < grins >, < arrrrglll >. Neben Abkürzungen (»cu« heißt »see you« oder »-« bedeutet »bis gleich!«) gibt es noch die Minimalgrafik mit ASCII-Zeichen wie die unendlich vielen Smileyvariationen: :) :-)) :-(und ;-)

(Den Kopf beim Lesen bitte nach links drehen!)

Systembetreuung

Um die Betreuung der Mailbox kümmert sich das Team des elektronischen Cafés. Aufgaben und Anforderungen sind überaus vielfältig:

- technische Grundkenntnisse von Betriebssystem und Programm,
- kaufmännisches Grundwissen,
- Organisationstalent für die Verwaltungsarbeiten,
- journalistische Fertigkeiten (zur Recherche, für eigene Beiträge und zur Gestaltung des Archivs),
- Aufmerksamkeit und vielseitige Interessen,
- Hilfsbereitschaft und Geduld (jede Minute ist die erste Minute in der Mailbox - für irgendeinen Teilnehmer).
- Am wichtigsten: die Fähigkeit, mit anderen zusammenzuarbeiten, mit den anderen im Team und den vielen Menschen innerhalb und außerhalb des Netzwerkes.

Ein hochqualifizierter Beruf also für Frauen - das Kaffeekochen allerdings sollten sie dann auch mal andere machen lassen...

Zukunftsvisionen

Ganz allgemein wird das Sammeln, Aufbereiten und Vermitteln von Information zur wichtigsten Handelsware. Ein beträchtlicher Teil dieser Information wird ohne Zweifel sprachliche Information sein. Die Kompetenz von Frauen in diesem Bereich wird ihnen in Zukunft sehr viel bessere Chancen eröffnen. Und dies ist gleichzeitig eine große Chance für die Gesellschaft.

Mailbox-Nutzung wird in einigen Jahren für viele so einfach und selbstverständlich sein wie heute telefonieren. Die Vernetzung schreitet allerorten in Riesenschritten voran. Dies ist noch kein kultureller Fortschritt an sich. Aber wie diese Vernetzung aussehen wird, wer sie nutzt und was über sie mitgeteilt werden wird, ist noch offen. Die Mailbox ist im Moment ein Labor, in dem experimentiert wird, ein Mikrokosmos, in dem sich eine neue Kultur entwickelt, neue Regeln und Umgangsformen erfunden und er-



probt werden, die das Leben im »Globalen Dorf« bestimmen werden.

Meine Vorstellung vom Dorfbrunnen: Ein elektronisches Caféhaus mit Anschluß an die elektronische Stadtbibliothek. Wobei reale Räume mit Caféhausqualitäten der ideale Rahmen für das virtuelle Caféhaus sein könnten. Dies wird die Schnittstelle zwischen Mensch und Netz sein - ein Ort, wo Austausch und Orientierung zwischen Alltagserfahrung und Wissenschaft, Spezialistentum und Allgemeinbildung gefunden und gemeinsame Projekte entstehen können und wo ich mich vergewissern kann, daß die anderen Menschen noch real vorhanden und keine Simulation sind. (Die wichtigen Begriffe »Cyberspace«, »interdisziplinär« und »interaktiv« tauchten bisher nicht auf in diesem Artikel. Das sei hiermit nachgeholt ;-)

Rena Tangens

Anmerkungen

- 1) Modem: Kunstwort aus Modulator-Demodulator. Wandelt die digitalen Signale des Computers in akustische Signale um, die über die Telefonleitung übertragen werden. Und umgekehrt.
- 2) Während sich im Deutschen die Bezeichnung »Mailbox« eingebürgert hat, heißt die gleiche Einrichtung in englischsprachigen Ländern »bulletin board system« oder kurz »bbs«.
- 3) Als Abhilfe setzen Sie bitte in Gedanken statt beispielsweise »Frauen beschäftigen sich weniger mit Computern« das exaktere »Es gibt weniger Frauen als Männer, die sich mit Computern beschäftigen« ein. Kein Unterschied? Nochmal lesen!
- 4) weibliche Form von »Hacker«. Hacken bedeutet nicht, in fremde Rechnersysteme einzudringen, sondern sich intensiv mit einer Sache zu beschäftigen, dabei kreativ und kritisch Möglichkeiten abseits der ausgetretenen Pfade zu erkunden. Dafür braucht es nicht unbedingt einen Computer - »real hacking«.
- 5) Mehr zu diesem Thema findet sich in den ersten drei Büchern der Literaturliste (zusammen mit den Biographien der Referentinnen im Anhang).

Literarische Automaten

Herbert Heckmann

Schon lange hatten die Dichter vom hilfreichen Automaten geträumt, der dem Luxusgeschöpf Mensch mechanische Arbeiten abnimmt. Doch erst die Renaissance-Künstler schufen Figuren mit einer Mechanik, als hätten sie Lieb' im Leib.

Die ersten Automaten und Androiden entstanden in der Phantasie der Dichter und verraten etwas von der unermüdbaren Hoffnung des Menschen, das Lebendige künstlich nachgestalten zu können. Daß die Poeten dabei recht praktische und nützliche Nebengedanken hatten, beweist schon der große Homer, der in seiner Ilias künstliche »goldene Mägde« beschreibt, die dem schmiedenden Gott Hephaistos emsig zur Hand gingen. Die Stelle lautet in der Übersetzung von Johann Heinrich Voss: »...auch stützten geschäftige Mägde den Herrscher. Goldene, Lebenden gleich, mit jugendlich reizender Bildung: Diese haben Verstand in der Brust und redende Stimme. Haben Kraft und lernten auch Kunstarbeit von den Göttern.« (Ilias XVIII, S. 417-420)

Homer erwähnte diese mechanische Dienerinnen eher nebenbei, aber die Griechen, die einen ausgesprochenen Sinn für die Muße hatten – erlaubte diese ihnen doch all das, was sie gern und kundig taten, nämlich zu philosophieren, zu spekulieren, ihren Neigungen nachzugehen und, nicht zuletzt, das Leben zu genießen –, eben jene Griechen werden diese Nebensächlichkeiten ohne jeden Zweifel mit sehnsuchtsvollen Seufzern quittiert haben.

Was den Poeten billig war, konnte den Philosophen nur recht sein. So entwarf Aristoteles in seiner »Politik« (I,3) das Bild einer von Automaten bevölkerten Welt, in der sich der Mensch nicht mehr mit Werkzeugen, Dienern und einer komplizierten Arbeitsteilung abplagen müsse, um für sein leibliches wie geistiges Wohl zu sorgen. »Wenn nämlich jedes einzelne Werkzeug auf einen Befehl hin oder einen solchen vorausahnend eine Aufgabe erfüllen kann (...), wenn also auch das Weberschiffchen so webte und das Plektron die Kithara schlüge, dann bedürften weder die Baumeister der Gehilfen noch die Herren der Sklaven.«

Androiden als perfekt funktionierende, jedem Befehl sich bequembende Diener – das muß für die Zeit des Aristoteles, die auf die Hilfe und Arbeit der Sklaven bauen mußte, weil es eben immer einfacher ist, andere für sich arbeiten zu lassen als selbst zuzugreifen, schon eine verlockende Utopie gewesen sein. Die Wirklichkeit sah freilich anders aus. Die ersten Automaten, die man in der Antike bastelte, waren amüsante technische Spielereien, die weiter keinen Nutzen brachten. So gab es, wie Lukian nicht ohne Spott berichtete, allerlei automatischen Hokuspokus im

Tempel der assyrischen Hera, der die Gläubigen in ehrfürchtiges Staunen versetzte.

Schauplatz der ersten Automaten, bei denen nicht mehr die Phantasie allein Pate stand, war das ptolemäische Alexandria, ein Stadtstaat von zehn Millionen Menschen, die von einer allgewaltigen Bürokratie verwaltet wurde. Hier befand sich auch eine riesige Akademie, die das Erbe der Antike mit enzyklopädischem Eifer pflegte. In keiner Stadt der Alten Welt wurde das Studium der Naturwissenschaften und der Technik so gefördert wie in Alexandria. Die großzügigen Ptolemäer versorgten ihre Gelehrten mit einer Staatspension und überließen sie ungestört ihren Studien. Zentrum dieser Akademie, die wir uns als eine Universität vorstellen müssen, war das Museion im Stadtteil Brucheion am großen Hafen. Zum gemeinschaftlichen Gebrauch der Gelehrten diente eine große Bibliothek, die, wenn man den alten Angaben Glauben schenken darf, 700 000 Rollen besaß.

Es war wohl kein Zufall, daß gerade hier der Bau von Automaten Interesse erregte – die Arbeitsteilung der mächtigen Bürokraten bot das beste Vorbild für die automatischen Maschinen und Konstruktionen, die überdies eine aufwendige Künstlichkeit zeigten. Der durch den Handel erworbene Luxus hatte den Sinn der Alexandriner für das Raffinement geschärft.

Ktesibios, ein Zeitgenosse des großen Archimedes im dritten Jahrhundert v. Chr., war wohl der erste in dieser Stadt, der nähere Angaben machte über den Bau von Automaten. Seine Schrift über die Pneumatik, in der er seine Automatenkonstruktionen erläuterte, ist jedoch verschollen. Beschreibungen seiner Erfindungen finden sich in den zehn Büchern über die Architektur des Vitruv, die um 27 v. Chr. verfaßt wurden. Ktesibios war der Sohn eines Barbiers und soll schon in jungen Jahren die Kunden seines Vaters mit allerlei mechanischen Spielereien unterhalten haben. Theoretische Konsequenzen hat er aus seinen Konstruktionen nicht gezogen. Ktesibios war ein überaus geschickter Bastler, mehr nicht. Im Prinzip wurden seine Maschinen durch das Verhalten von Luft und Flüssigkeit unter Druckeinwirkung angetrieben.

Wie weit man damals in der Technik war, verrät das pseudoaristotelische Traktat »Über mechanische Probleme«. In ihm werden als mechanische Hilfsmittel genannt: der Hebel, der Schwengel mit Gegengewicht, die gleicharmige Waage, die Schnellwaage, der Keil, die Axt, die Walze, das Wagenrad, die Rolle und Drehräder aus Erz und Eisen zur Umkehrung der drehenden Bewegung, worunter wahrscheinlich Zahnräder zu verstehen sind. Aus der Einfachheit der Handwerkzeuge darf keineswegs den Schluß ziehen, die Automaten seien klapprig gewesen. Bei keinem antiken Autor wird man eine schadenfrohe Schilderung einer Automatenpanne finden können. Berühmt wurde vor allem die Wasseruhr des Ktesibios, die nach der Beschreibung des Vitruv auch ein Automaten-theater vorführte. Ein Putto zeigte mit einem Stab die Stunden auf einer Säule an, während ein anderer sich abkehrte und Tränen über die Vergänglichkeit



vergoß. Mehr noch als Ktesibios hat sich Heron von Alexandria der Nachwelt eingepägt. Da seine Schriften zu einem großen Teil erhalten sind, und da er recht genaue Angaben über seine Konstruktionen machte, können wir uns ein recht präzises Bild von seinen Automaten machen. Heron lebte im ersten Jahrhundert n. Chr. Sein Werk ist also eine späte, großangelegte Zusammenfassung des technischen Wissens und Könnens der Alexandriner, die, wie wir sehen werden, ihre Kenntnisse vielfältig vererbten. Herons Buch über die Automatentheater, das wohl im 13. Jahrhundert ins Lateinische übersetzt worden ist, und seine Abhandlung über die Pneumatik haben den Automatenbau über Jahrhunderte, ja über mehr als ein Jahrtausend beeinflusst.

Heron konstruierte unter anderem einen fahrenden Automaten, der die Apotheose des Gottes Bacchus inszenierte; ein anderer führte die Nauplius-Sage vor, die einen recht komplizierten Szenenverlauf erforderte. Nicht minder bemerkenswert sind die automatischen Spielereien, die Heron in seiner Schrift über die Pneumatik aufführt. Da ist die Rede von zwitschernden Vögeln, vom Pfeilschießenden Herkules und einem zischenden Drachen, von einem Münzautomaten und dergleichen mehr. Heron besaß eine unerschöpfliche Phantasie. Mit Hilfe des Wasserdrucks, des Drucks atmosphärischer oder erwärmter Luft oder mit Dampfdruck gelangen ihm die tollsten mechanischen Spiele, und die wundergläubigen Zuschauer werden nicht aus dem Staunen herausgekommen sein. Einen industriellen Gewinn brachten diese Erfindungen nicht. Die Antike beschränkte sich

auf spielerische Demonstrationen naturwissenschaftlicher Entdeckungen.

Als die Araber 641 Alexandria eroberten, das sich vierzehn Monate lang tapfer verteidigt hatte, übernahmen sie das technische Erbe der Antike und bildeten es weiter aus. Ihr Interesse galt vor allem den Wasseruhren, die sie zum regelrechten Zeittheater ausbauten. Uhren hatten im Islam eine kultische Bedeutung, zeigten sie doch die Stunde des Gebets an. Was Wunder also, daß man sie mit einem Automaten-theater verquickte, das sich symbolisch auf den Zeitverlauf bezog. Eine solche kunstvolle Wasseruhr erhielt übrigens Karl der Große von einer Gesandtschaft des Kalifen zum Geschenk. Durch den Bericht Einhards, des Biographen Karls des Großen, wissen wir, wie sie ausgesehen hat: »Auch ein höchst kunstvoll aus Messing gearbeitetes Uhrwerk war bei den Geschenken, in dem der Lauf der zwölf Stunden nach einer Wasseruhr sich bewegte mit ebensovielen Eisenkugeln, die nach Ablauf der Stunden herunterfielen und dadurch ein darunter liegendes Becken zum Erklingen brachten, außerdem waren dann zwölf Reiter, die am Ende der Stunden aus zwölf Fenstern hervorkamen und durch ihre Bewegung ebensoviel zuvor verschlossene Fenster aufstießen.«

Die Ähnlichkeit der chinesischen Wasseruhren mit den arabischen lassen den Schluß zu, daß Reisende aus China Automatenkenntnisse mit nach Hause brachten. Eine genaue Untersuchung dieses Kulturaustauschs liegt leider noch nicht vor. Auch die Beschreibung von Automaten in der mittelalterlichen Literatur geht wohl in den meisten Fällen auf

arabische Vorbilder zurück, wobei die abendländischen Poeten die morgenländischen Märchenerzähler mitunter noch weit übertreffen. Technische Angaben und präzise Details wird man da jedoch vergeblich suchen. Die Frage, wie diese Automaten überhaupt funktionieren konnten, hat die Dichter kaum beschäftigt. Ihnen kam es einzig und allein darauf an, Wunderbares zu schildern. Daß die Automaten auch etwas mit den Wissenschaften zu tun haben müssen, war ihnen jedoch nicht gänzlich unbekannt.

So nannte Heinrich von Veldeke den Erfinder des Bogenschützenautomaten in seinem Versroman »Eneit« (zwischen 1180 und 1190) einen Meister Geometrias. Zu welcher technischen Vollkommenheit es die Araber beim Bau ihrer automatischen Wasseruhren brachten, zeigt das technologische Compendium Gazaris, das aus dem Jahre 1206 stammt. Zu dieser Zeit ging man in Europa daran, Räderuhren zu entwerfen, ohne jedoch zunächst praktischen Erfolg zu haben. Die ersten Turmuhren, von denen wir sichere Kunde haben, stammen aus dem frühen 14. Jahrhundert, und es sollte nicht lange dauern, bis diese Turmuhren auch ein automatisches Theater vorführten: Szenen aus der Bibel, den Aufmarsch der Heiligen Drei Könige, Christi Kreuzestod, die Prozession des Todes, eben das, was zur frommen Erbauung der Turmuhrgucker gehörte.

Eines der berühmtesten Automatenwerke besaß das Straßburger Münster. Es wurde in den Jahren 1352 bis 1354 von einem unbekanntem Meister erbaut und nach und nach immer weiter verbessert. Ein aus Eisen geschmiedeter Hahn krächte die Stunden an, schlug mit den Flügeln und bewegte die Kehlappen. Es ist die älteste uns erhaltene Automatenfigur einer Uhr. Als sich die Federuhr, die im 15. Jahrhundert aufkam, durchgesetzt hatte, begann man auch kleinere Automatenuhren und hauptamtliche Automaten zu bauen. Als Triebkraft verwendete man die Elastizität eines langen, spiralförmig gewundenen Stahlbandes, der Stahlfeder, die in einem zylindrischen Gehäuse eingeschlossen wurde. Ein Ende der Feder befestigte man an der Wand, das andere an der Achse des Federhauses. Wurde nun die Feder aufgezo- gen, so drehte die Elastizität der Feder das Federhaus sowie das auf diesem angebrachte Zahnrad, das wiederum das Uhrwerk in Bewegung setzte.

Der Federantrieb wurde zur Seele des Automaten. Uhrmacher und Feinschmiede des Jahrhunderts haben wahre Automatenkunstwerke geschaffen, die uns noch heute in Erstaunen zu setzen vermögen. Der Automatenbau wurde damals zum Inbegriff des manieristischen Kunstschaffens. Der Mensch hatte durch einen Ritz in die Werkstatt der Natur geschaut und glaubte nun selbst, als zweiter Schöpfer (die manieristischen Theoretiker gebrauchten diese Bezeichnung nicht ohne Stolz) alles mechanisch nachschaffen zu können. Was die nachgerade genialen Handwerker des Manierismus erstrebten, war so etwas wie ein Gesamtkunstwerk, eine vollkommene Nachahmung der Natur. Automatische Planetarien hatten schon Giovanni de Dondi zwischen 1348 und 1363

sowie der deutsche, handwerklich sehr geschickte Reiomontanus gebaut, der die Himmelsuhr des Italieners 1461 während sein ersten Italienaufenthalts besichtigen konnte und sich wohl durch sie anregen ließ.

Die Erfindung der Räderuhr wie ihre Vervollkommnung führten sehr bald zu der Zuversicht, die Welt »mechanisch« nachschaffen zu können. Zunächst mag das eine rein technische Spielerei und ein handwerkliches Experimentieren gewesen sein, aber dahinter steckte auch die Hoffnung, daß es einmal gelingen könne, ein vollkommenes mechanisches Wesen zu basteln.

Fast jeder bessere Uhrmacher und Feinschmied versuchte sich damals an Automaten. Selbst Leonardo da Vinci war sich nicht zu gut, Automaten zu bauen. So wird erzählt, daß er für ein Fest, das zu Ehren Ludwigs XII. 1509 in Mailand gefeiert wurde, die Figur eines Löwen konstruiert habe, der auf den Thron des französischen Königs zuschritt, sich mit der Pranke die Brust aufriß und in schmeichelnder Symbolik einen Strauß Lilien freilegte.

Es waren vor allem die Fürsten und Herrscher im 16. Jahrhundert und in der Barockzeit, die eine geradezu besessene Freude an Automaten bekundeten und keine Kosten scheuten, um ihre Melancholie mit derlei Spielereien zu vertreiben. Ein wenig Imponiergehabe und Machtzauber mögen dabei mit im Spiel gewesen sein. So wissen wir von dem schwermütigen Habsburger Rudolph II., daß er ein magisches Vergnügen an Automaten hatte und die besten Handwerker beschäftigte, die für ihn unter anderem ein automatisches Krippenwerk und einen aktionsreichen Turm von Babel bauten – Automaten, die unter Musikgeklingel die zierlichsten Bewegungen ausführten. Musik und mechanisches Spiel vereinigten sich zu einem kunstvollen Maschinentheater. Zu festlichen Gelegenheiten gehörte der Auftritt verblüffender Automaten, wie wir von vielen Chroniken und barocken Romanen wissen.

Der Jesuit Faminian Strada erzählt, daß Kaiser Karl V. nach seiner Abdankung im Kloster San Yuste sich mit automatischen Spielereien seines Mechanikers und Uhrmachers Juanelo Turriano die melancholische Grillen vertrieben habe. So ließ er nach dem Mittagmahl Soldatenfiguren auf dem Tische kämpfen, auf dem auch künstliche Pferde herumtrabten; Soldaten schlugen die Trommel, andere bliesen die Trompete. Schließlich flogen noch künstliche Sperlinge durch den Raum. Der Kaiser herrschte nur noch über Automaten.

Die wohl bedeutendsten Automatenbauer stammten aus Nürnberg und Augsburg, damals die beiden Hauptstädte europäischen Handwerkskunst in Europa. So baute der Nürnberger Kunstschlosser Jakob Bulmann eine Himmelsmaschine, die mit einem komplizierten Automatentheater verbunden war. Johann Doppelmayr bemerkte dazu in seiner »Historischen Nachricht von den Nürnberger Mathematicis und Künstlern« (Nürnberg 1730): »Die Himmelsmaschine stellte ferner Manns- und Weibspersonen mit

Uhrwerken dergestalt beweglich vor, daß sie hin und her gingen und nach der Mensur auf die Pauke oder auf die Laute schlugen. Ferdinandus, der damalige römische Kaiser, ließ diesen Künstler noch in seinem hohen Alter in einer Sänfte nach Wien bringen, da er sich wegen verschiedener Uhrwerke, die ihm präsentiert worden, seines Rates bediente. Seine Rückkehr geschah mit gleicher Bequemlichkeit, darauf er nach einiger Zeit und zwar um 1535 gestorben.«

Für das Publikum des 16. und 17. Jahrhunderts war die Zeit stets ein Drama der Veränderung, der Vergänglichkeit. So ist es nur zu verständlich, daß den meisten automatischen Spielereien ein allegorischer Sinn zugrunde gelegt wurde. Die automatischen Schiffe, die der Augsburger Uhrmacher Hans Schlottheim (1544/47 bis 1625) baute, sind als mechanisch inszenierte Metapher vom Lebensschiff zu verstehen, die seit der Antike im Schwange war und immer wieder aufgegriffen wurde. Handwerkliches Können überzeugte sehr bald einige Philosophen, daß das Leben – sieht man einmal vom immateriellen Charakter der Seele ab – ein mechanischer Automatismus sein müsse.

So schreibt René Descartes in seinem »Discours de la Méthode« (1637), daß bei der Betrachtung des Aufbauprinzips eines Körpers diejenigen weniger Schwierigkeiten haben, die wissen, wieviele verschiedenartige Automaten oder bewegungsfähige Maschinen die Geschicklichkeit der Menschen zustande bringen können, wobei sie nur recht wenige Stücke verwenden, im Vergleich zu der großen Vielheit der Knochen, Muskeln, Nerven, Arterien, Venen und all der anderen Teile, die im Körper jedes Tieres vorhanden sind. Descartes sah in dem Tier schlechterdings einen Automaten.

Es war dann Julien Ofray de la Mettrie (1709 bis 1751), der auch im Menschen eine Maschine am Werk sah, wie es vor ihm schon der spanische Arzt Gomez Pereira 1554 getan hatte. Er verstand die Materie nicht wie Descartes als bloße Ausdehnung (*res extensa*), sondern sie besaß nach seiner Auffassung auch die Kraft der Bewegung und die Fähigkeit der Empfindung. So habe auch das Tier durchaus Gedanken und Gefühle. Schon diese Bemerkungen zeigen, daß La Mettrie keineswegs groben und platten mechanistischen Vorstellungen huldigte, wie es ihm seine Gegner vorwarfen. Er glaubte an die mechanische Kopie des Menschen, wie er auch glaubte, daß Affen zum Sprechen zu bringen seien.

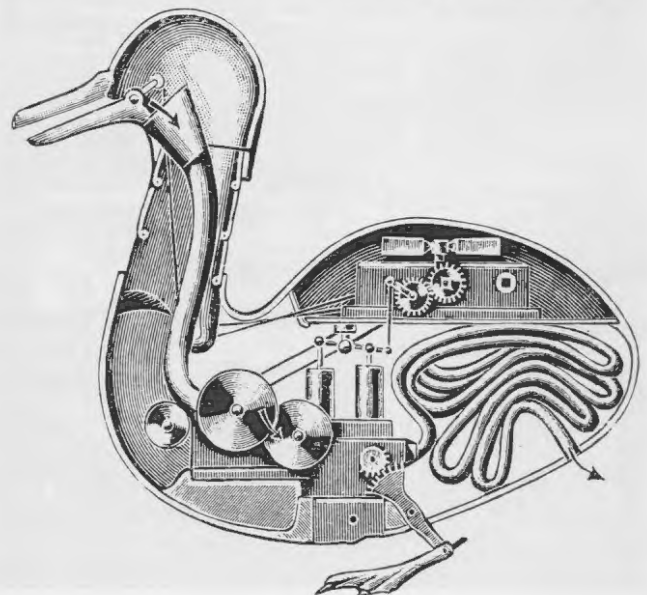
Im Unterschied zu den Polyhistoren der Barockzeit, die die ganze Welt mit all ihren Geheimnissen und Kuriositäten in eine pansophische Hierarchie unterzubringen suchten, zog er seine Schlüsse aus der Erfahrung. Er war überzeugt, daß alles machbar sei, sofern man es nur, mit genügend Kenntnissen gewappnet, anpackt. Eine zweite Schöpfung aus dem Geiste der Mechanik lag für ihn, der sich übrigens wie Descartes an den Automaten seiner Zeit orientierte, durchaus im Bereich des Möglichen. So hat La Mettrie die Automaten Vaucansons zum Vergleich herangezogen, sich jedoch keineswegs von ihnen zur

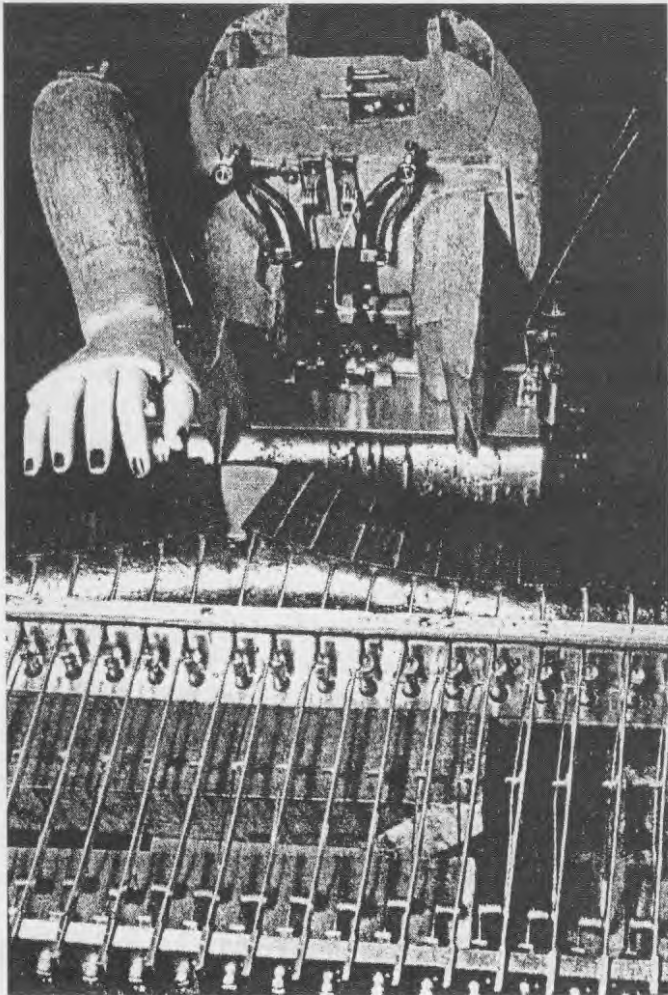
Theorie von der Menschenmaschine anregen lassen. Bemerkenswert ist freilich die Tatsache, daß Vaucansons Automaten und La Mettries Abhandlung »Der Mensch eine Maschine« (*L'homme machine* 1748) fast zur selben Zeit entstanden sind. Beides sind wohl die extremsten Ergebnisse eines konsequent mechanistischen Denkens.

Jacques Vaucanson (1709 bis 1782) hatte 1738 die Akademie zu Paris mit einer mechanischen Ente überrascht, die nicht nur schwimmen und sich bewegen, sondern auch fressen, verdauen und das Verdauete auf die natürlichste Art und Weise wieder von sich geben konnte. Mit der Ente präsentierte Vaucanson einen Flötenspieler sowie einen Trommler. Beide sahen höchst possierlich aus und brachten die Damen dazu, sich in die mechanische Grazie zu verlieben.

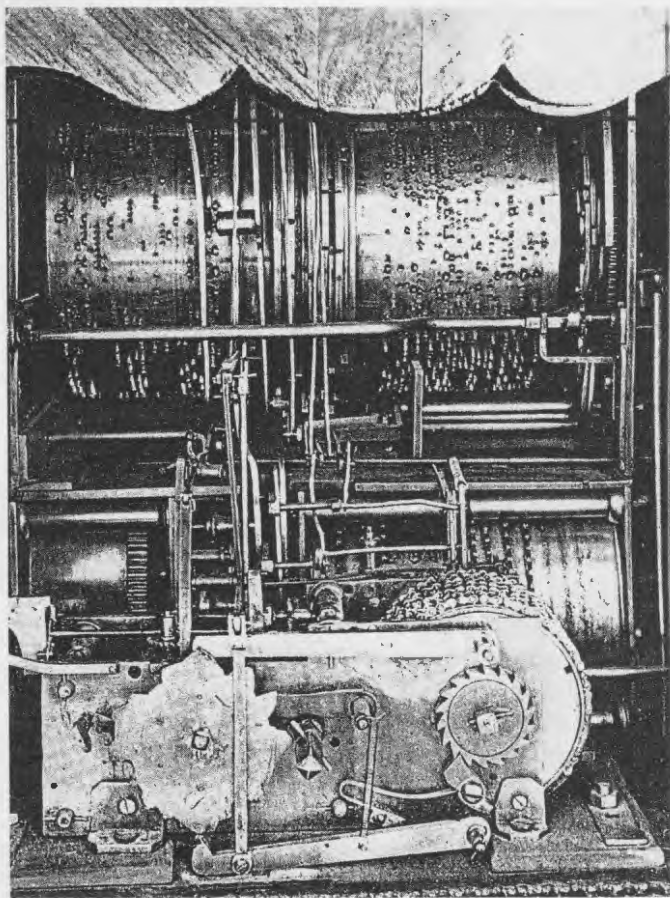
Nur wenige wagten Kritik, wie der deutsche Erzaufklärer Friedrich Nicolai, der so weit ging, den Ingenieur Jacques Vaucanson, der sowohl eine Seidenmaschine als auch einen mechanischen Webapparat erfunden hatte, einen ausgemachten Windbeutel zu nennen. Aber Vaucanson, dem Diderot vorwarf, über die Maßen eitel und geldgierig zu sein, verstand sein Handwerk. Die Präsentation seiner Automaten vor erlauchtem Kreise erregte ungeteilte Bewunderung. Die Ente aus Messing zeichnete sich durch die Bewegung von Hals und Flügeln aus. Vor allem wurde das Sträuben der Federn bewundert. Was die Verdauung angeht, so glaubten die Kritiker, daß am unteren Teil des Körpers ein Blasebalg angebracht worden sei, dessen Luftsauger durch eine Röhre bis an den Hals der Ente gehe und hier die Futterkörner in den Schnabel ziehe, während im Hinterteil eine vorbereitete Masse liege.

Nach seinem spektakulären Erfolg, von dem alle Zeitungen in gebührender Übertreibung berichteten, verkaufte Vaucanson seine Automaten sehr günstig. Sie setzten ihren Triumphzug auf Jahrmärkten fort. Schließlich wurden sie ersteigert von dem gelehrten Kauz und Sammler Gottfried Christoph Beireis, bei dem Johann Wolfgang von Goethe sie 1805 in Helm-





Das Geheimnis der Orgelspielerin von La Chaux-de-Fonds



stedt höchstpersönlich inspizieren konnte. Sein Bericht freilich klingt recht melancholisch und läßt kaum etwas von der ehemaligen Würde der Automaten erahnen: »Die Ente, unbefiedert, stand als Gerippe da, fraß den Hafer noch recht munter, verdaute jedoch nicht mehr, an allem dem ward Beireis aber keineswegs irre, sondern sprach von diesen veralteten, halb zerstörten Dingen mit solchem Behagen und solch wichtigem Ausdruck, als wenn seit jener Zeit die höhere Mechanik nichts frisches Bedeutenderes hervorgebracht hätte.«

Den Höhepunkt der mechanischen Automatenbauerei stellen zweifellos die Werke dar, die Pierre Jaquet-Droz (1721 bis 1790, sein Sohn Henri-Louis (1752 bis 1791) und Jean Frédéric Leschot (1746 bis 1842) konstruierten. 1774 hatten die Automatenfiguren Schriftsteller, Zeichner und Orgelspieler ihren ersten Auftritt vor heimatlichem Publikum. 1775 stellten ihre Erfinder, die gemeinsam in La Chaux-de-Fonds die Uhrmacherei ausübten, sie in Paris vor, und im selben Jahr konnte man in der Vossischen Zeitung lesen: »Herr Jaquet-Droz aus Neuchâtel hat viele sonderbare mechanische Werkzeuge erfunden. Wir tun einiger Erwähnung. Eine Figur stellt ein Kind von zwei (!) Jahren vor, es sitzt an einem Tische und schreibt an einem Pult. Es tunkt seine Feder in die Dinte, spritzt den Überfluß weg und schreibt deutlich und korrekt alles, was man ihm vorsagt, ohne daß es von einer Person angerührt wird. Es setzt die Anfangsbuchstaben an den rechten Ort und läßt zwischen den Worten den gehörigen Raum. Wenn es eine Linie zu Ende hat, dann fängt es die folgende an und beobachtet auch die gehörige Entfernung zwischen den Linien. Solange es schreibt, wendet dasselbe die Augen auf sein Werk, wenn es aber einen Buchstaben oder ein Wort beendet hat, dann wirft es die Augen auf die Vorschrift, wo es die Charaktere nachzumachen scheint. Ein anderes zeichnet und trägt den ersten Umrissen den Schatten auf, es verbessert auch die Unvollkommenheiten seines Werkes und sieht vollkommen natürlich auch in allen Bewegungen aus. Eine Mädchenfigur von zehn oder zwölf Jahren spielt auf einem Flügel.«

Dem Schreiber ist vor Erstaunen die Wahrheit aus den Händen geraten. Die Figur des Schriftstellers, ein Kerlchen von drei bis vier Jahren, der recht anmutig auf einem Hocker im Louis-XV.-Stil sitzt, war natürlich außerstande, nach Diktat zu schreiben. Was er zu Papier bringen konnte, war ihm von Jaquet-Droz, der die Systematik des Mechanismus mit Hilfe Leschots entworfen hatte, eingegeben worden. Der Mechanismus besteht aus Räderwerken.

Den Zeichner, der nach dem Aussehen ein Bruder des kleinen Schriftstellers sein könnte, hat Henri Jaquet-Droz zum großen Teil in nur zwei Jahren fertiggestellt. Auch er besitzt zwei Mechanismen. Wenn ein Nockensystem innehält, um seine Position zu wechseln, bläst der Zeichner behutsam über das Blatt, um es vom Staub des Bleistiftes zu befreien. Um dies immer wieder tun zu können, zieht sich der Blasebalg, der im Kopf des Automaten sitzt, in gewissen

Abständen abrupt zusammen. Die Musikerin, ein hübsches Mädchen mit feinen Zügen, ist größer als ihre beiden Brüder. Die fünf Stücke, die sie auf einer Orgel (und nicht auf einem Clavecin, wie es den Anschein hat) mit Flötenklang spielt, sollen von Henri Jaquet-Droz selbst komponiert worden sein, der Musik studiert hatte. Sie sind jedoch weniger anmutig als die Interpretin, die tatsächlich mit ihren Bewegungen den automatischen Ursprung vergessen läßt. Der recht komplizierte Mechanismus besteht aus vier Teilen. Die drei Künstler erfreuen sich noch heute bester mechanischer Gesundheit und sind im Museum für Kunst und Geschichte zu Neuchâtel zu bewundern. Bemerkenswert ist, daß sowohl die Schöpfungen Vaucansons als auch die der Jaquet-Droz auf die Phantasie der Dichter einen großen Einfluß gehabt haben, die doch sonst eher selbst derlei Wunderdinge vorauszuahnen pflegen. Mit Jean Paul und E.T.A. Hoffmann begann nachgerade ein literarischer Automatenboom, wobei man nie sicher sein kann, ob es Beunruhigung oder die phantastischen Möglichkeiten waren, die die Dichter und Schriftsteller beflügelten.

Die Automatenlust führte auch, vor allem im 19. Jahrhundert, zu den listigsten Fälschungen. So versetzte der von Wolfgang von Kempelen konstruierte und von dem Wiener Musiker Maelzel gekaufte und vorgeführte automatische Schachspieler ganz Europa und später auch noch Amerika in Erstaunen, bis einige findige Köpfe, darunter auch der scharfsinnige Edgar Allan Poe, herausbrachten, daß es sich bei dem fast immer triumphierenden Schachspieler nicht um ein automatisches Wunder handeln könne. Tatsächlich verbarg sich in dem Kasten, der den Automaten darstellte, ein menschliches Schachtalent.

John N. Maskelyne verblüffte das Publikum in der Londoner Egyptian Hall gleich mit zwei mechanischen Wunderwerken, mit dem stets siegreichen Kartenspieler »Psycho« und der koketten Künstlerin »Zoe«, die Kunstfreunde aus dem Publikum porträtierte – beide so raffinierte Fälschungen, daß schon die mechanische Dramaturgie größte Bewunderung verdiente. Automaten wurden im 19. Jahrhundert zu unentbehrlichen Jahrmarktsbelustigungen, und für die Kinder zahlungskräftiger Eltern avancierten sie zum beliebten Spielzeug: Da gab es Stehaufmännchen, Purzelbaumschläger, Balancierfigürchen, Tänzerinnen, Musikanten und Akrobaten.

Im 20. Jahrhundert lösten die Roboter die in die Museen abgewanderten Automaten ab. Auch sie hatten zunächst einen literarischen Auftritt. Karel Čpek ließ am 25.1.1921 in seinem zu Prag uraufgeführten utopischen Kollektivdrama R.U.R. (Rossum's Universal Robots) Maschinenwesen auftreten, die er Robots nannte – nach dem tschechischen Wort »rotota« (Fronarbeit). Wenn auch Čpek die Roboter in seiner Weltuntergangsutopie sehr kritisch einschätzte, so hat das die immer auch sehr spielfreudigen Techniker nicht daran gehindert, ihre immer komplizierteren Androiden jetzt Roboter zu nennen, die elektrisch ferngesteuert, menschliche Tätigkeiten vorführten.

So bewies der Roboter »Voder« 1939 in Philadelphia, daß er sich nicht nur wie ein Mensch bewegen, sondern daß er auch zugreifen und sprechen konnte, wie es ihm der »operator« eingab. Und es sollte nicht lange dauern, bis man Roboter zustande brachte, die mit elektronischer Unterstützung, wenn auch nicht vollkommen, Sinneswahrnehmungen, menschliche Motorik, Erinnerung und Lernfähigkeit demonstrieren. Der vollkommene, künstliche Mensch: Mr. Roboter scheint endlich Wirklichkeit geworden zu sein, besonders in der Industrie, wo er dem Menschen, wie es sich Aristoteles erträumte, so unauffällig zu Hand geht.

Der Amerikaner Hubert L. Dreyfus warnt jedoch davor, die künstliche Intelligenz, die der Mensch nach dem Modell seines intellektuellen Selbstverständnisses konstruiert, mit der tatsächlichen Arbeit unserer grauen Gehirnzellen gleichzusetzen. Was die besten Denkmaschinen auch immer leisten können, bestimmte Denkkakte bleiben ausschließlich dem Menschen vorbehalten. »Große Künstler haben schon immer die Wahrheit gespürt, die so hartnäckig von Philosophen und Technikern geleugnet wird, daß nämlich die Grundlage der menschlichen Intelligenz nicht isoliert und explizit verstanden werden kann«, resümierte Hubert L. Dreyfus. Die Roboter bleiben also Roboter – und die Automaten nur Automaten und ohne Schicksal.



Maria Stuart hätte ihren Kopf behalten können, wenn...

Thomas von Randow

Die sichersten Techniken zur Verschlüsselung von Nachrichten und die genialsten Dechiffrier-Methoden verrät Thomas von Randow in einem Streifzug durch die Geschichte der Kryptologie.

Wollten Sie nicht schon immer ein vielfacher Millionär werden? Es geht um einen Gold-, Silber- und Juwelenschatz im Wert von 24 Millionen Mark. Er liegt in Virginia vergraben und die Regierung dieses US-Staates hat verbindlich erklärt, daß der Finder des Schatzes ihn auch ohne irgendeinen Abzug behalten darf.

Um an dieses Vermögen zu kommen, brauchen Sie nur den Fundort zu kennen, sich mit einer Schaufel zu bewaffnen, nach Virginia zu reisen – die Flüge dorthin sind zur Zeit äußerst billig – und die Kostbarkeiten auszugraben. Das wichtigste, den Fundort, erfahren Sie jetzt, allerdings verschlüsselt. Sie müssen also zuerst eine Chiffre entschlüsseln. Sie besteht aus 555 Zahlen, die auf dem ersten der drei Blätter geschrieben stehen, die als »Beale Cypher« in die Geschichte der Kryptologie eingegangen sind.

Die Geschichte der Kryptologie sollte heute mein Thema sein, aber als ich damit begann, ein Manuskript dafür zu schreiben, bemerkte ich sehr bald: Sie ist viel zu komplex, als daß man sie in zwanzig Minuten auch nur in Stichworten abhandeln könnte. Also habe ich beschlossen, zum Teil aus dieser Geschichte, teils aber auch aus eigener Erfahrung, ein paar Episoden zu erzählen, die das Thema einleiten sollen, welches nach mir Professor Widmann von der Schweizer Firma Crypto A.G. und seine beiden Mitarbeiter, die Herren Schwarz und Siemer, seriös und aktuell behandeln werden.

Zunächst freilich möchte ich schildern, was es mit der Beale-Cypher auf sich hat: Im Januar 1822 traf ein auffallend gut aussehender junger Mann mit schwarzem Haar in einer Länge, die das zu jener Zeit als schicklich geltende Maß erheblich überschritt, in Lynchburg ein, einem kleinen Städtchen in Virginia. Dort bezog der Jüngling, der sich Thomas Jefferson Beale nannte und nach eigenem Bekunden Goldgräber war, im Hotel »Washington« Quartier. Doch früher als geplant, mußte der Fremde den Ort wieder verlassen, weil ihm die Polizei auf den Fersen war. Der Grund: Beale sei, so hieß es, gesehen worden, als er die Wohnung einer Weibsperson verließ, mit der er nicht verheiratet war. Doch ehe sich der Goldgräber aus dem Staube machte, übergab er dem Besitzer des Hotels, einem unbescholtenen Bürger mit Namen Robert Morriss, eine versiegelte Blechbüchse zur Aufbewahrung. Diese sollte Morriss nach Ablauf

von zehn Jahren öffnen, falls Beale bis dahin nicht nach Lynchburg zurückgekommen sei.

Beale ließ sich nie wieder sehen. Morriss wartete noch bis zum Jahr 1845 ehe er sich traute, die Büchse zu öffnen. Darin fand er drei mit insgesamt 1901 Zahlen beschriebene Zettel und einen Brief, aus dem sich folgendes ergab: Beale habe in der Nähe von Bedford, Virginia, einen Schatz vergraben, der ihm und seinen 29 Kumpeln gehöre. Woraus der Schatz bestünde, wo er vergraben sei und wie er verteilt werden solle, all das ginge aus den drei beigelegten Zetteln hervor. Die freilich galt es zu entziffern.

Morriss widmete sich fortan einer unerwarteten Beschäftigung, der Kryptanalyse – vergeblich. Enttäuscht vertraute er die Geschichte seinem Barmixer, James Ward, an, weil der doch ein passionierter Zahlenpuzzler war. Ward vertiefte sich in die Beal'sche Zahlenchiffre so sehr, daß er alsbald seine Gesundheit, seine Familie und seinen Beruf sträflich vernachlässigte. Immerhin hatte er einen beachtlichen Erfolg. Er konnte das zweite Blatt mit der Beschreibung des Schatzes entziffern. Dieser, so verriet ihm die Chiffre, bestünde aus 2 912 Pfund Gold, 5 100 Pfund Silber und einer Menge Juwelen, im Gesamtwert von 24 Millionen Mark nach heutigen Preisen.

```
71, 194, 38, 1701, 89, 76, 11, 83, 1629, 48, 94, 63, 132, 16, 111,
95, 84, 341, 975, 14, 40, 64, 27, 81, 139, 213, 63, 90, 1120, 8,
15, 3, 126, 2018, 40, 74, 758, 485, 604, 230, 436, 664, 582, 150,
251, 284, 308, 231, 124, 211, 486, 225, 401, 370, 11, 101, 305,
139, 189, 17, 33, 88, 208, 193, 145, 1, 94, 73, 416, 918, 263, 28
500, 538, 356, 117, 136, 219, 27, 176, 130, 10, 460, 25, 485, 18,
436, 65, 84, 200, 283, 118, 320, 138, 36, 416, 280, 15, 71, 224,
961, 44, 16, 401, 39, 88, 61, 304, 12, 21, 24, 283, 134, 92, 63,
246, 486, 682, 7, 219, 184, 360, 780, 18, 64, 463, 474, 131, 160,
79, 73, 440, 95, 18, 64, 581, 34, 69, 128, 367, 460, 17, 81, 12,
103, 820, 62, 116, 97, 103, 862, 70, 60, 1317, 471, 540, 208,
121, 890, 346, 36, 150, 59, 568, 614, 13, 120, 63, 219, 812,
2160, 1780, 99, 35, 18, 21, 136, 872, 15, 28, 170, 88, 4, 30, 44,
112, 18, 147, 436, 195, 320, 37, 122, 113, 6, 140, 8, 120, 305,
42, 58, 461, 44, 106, 301, 13, 408, 680, 93, 86, 116, 530, 82,
568, 9, 102, 38, 416, 89, 71, 216, 728, 965, 818, 2, 38, 121, 195,
14, 326, 148, 234, 18, 55, 131, 234, 361, 824, 5, 81, 623, 48,
961, 19, 26, 33, 10, 1101, 365, 92, 88, 181, 275, 346, 201, 206,
86, 36, 219, 320, 829, 840, 68, 326, 19, 48, 122, 85, 216, 284,
919, 861, 326, 985, 233, 64, 68, 232, 431, 960, 50, 29, 81, 216,
321, 603, 14, 612, 81, 360, 36, 51, 62, 194, 78, 60, 200, 314,
676, 112, 4, 28, 18, 61, 136, 247, 819, 921, 1060, 464, 895, 10,
6, 66, 119, 38, 41, 49, 602, 423, 962, 302, 294, 875, 78, 14, 23,
111, 109, 62, 31, 501, 823, 216, 280, 34, 24, 150, 1000, 162,
286, 19, 21, 17, 340, 19, 242, 31, 86, 234, 140, 607, 115, 33,
191, 67, 104, 86, 52, 88, 16, 80, 121, 67, 95, 122, 216, 548, 96,
11, 201, 77, 364, 218, 65, 667, 890, 236, 154, 211, 10, 98, 34,
119, 56, 216, 119, 71, 218, 1164, 1496, 1817, 51, 39, 210, 36, 3,
19, 540, 232, 22, 141, 617, 84, 290, 80, 46, 207, 411, 150, 29,
38, 46, 172, 85, 194, 36, 261, 543, 897, 624, 18, 212, 416, 127,
931, 19, 4, 63, 96, 12, 101, 418, 16, 140, 230, 460, 538, 19, 27,
88, 612, 1431, 90, 716, 275, 74, 83, 11, 426, 89, 72, 84, 1300,
1706, 814, 221, 132, 40, 102, 34, 858, 975, 1101, 84, 16, 79, 23,
16, 81, 122, 324, 403, 912, 227, 936, 447, 55, 86, 34, 43, 212,
107, 96, 314, 264, 1065, 323, 328, 601, 203, 124, 95, 216, 814,
2906, 654, 820, 2, 301, 112, 176, 213, 71, 87, 96, 202, 35, 10, 2,
41, 17, 84, 221, 736, 820, 214, 11, 60, 760.
```

Beale-Chiffre, Blatt 1: Die exakte Beschreibung des Ortes, an dem der Schatz vergraben ist

Was Ward herausgefunden hatte war dies: Dem Zahlencode lag der Text der amerikanischen Unab-

hängigkeitserklärung zugrunde. Sie beginnt mit den Worten: »When In The Course Of Human Events It Becomes ...« Beale hatte die Anfangsbuchstaben der Wörter laufend nummeriert, also bekam das W die Nummer 1, I wurde 2, T wurde 3, ..., »It« beginnt wieder mit einem I und so erhielt I eine zweite Nummer, nämlich 8 ... Wegen des langen Textes der Unabhängigkeitserklärung hatte am Ende jeder Buchstabe nicht nur eine, sondern eine ganze Menge von Zahlen zugeordnet bekommen. An Stelle der Buchstaben seines Klartextes schrieb Beale die ihnen entsprechenden Zahlen, wobei er unter den verschiedenen zugeordneten Zahlen jeweils eine willkürliche Auswahl traf. Damit verhinderte er, daß man seiner Chiffre mit den Mitteln der Sprachstatistik beikommen konnte.

Mehr als Blatt 2 der Beale Cypher ist bis zum heutigen Tag nicht entziffert worden. Generationen von Tüftlern haben Tausende von Texten durchforstet, die dem nicht sonderlich belesenen Beale hätten bekannt sein können – ohne Erfolg. Das Interesse an der Chiffre erlahmte allmählich, bis in den sechziger Jahren die Computerei begann. Seither sind unzählige Computer mit unzähligen Texten gefüttert worden, haben Legionen von Informatikern alles herangezogen, was die Statistik an Hilfsmitteln bietet. Längst ist Computerzeit im Wert eines Vielfachen der 24 Millionen Mark verpulvert worden.

Immerhin hat es zwei Ergebnisse gegeben: Carl Hammer, ein Computerexperte, ermittelte in einer preisgekrönten Arbeit, daß die Zahlenkolonnen der Beale Chiffre keineswegs etwa zufällig zusammengewürfelt sind, sondern eine Struktur besitzen, mithin wohl eine Bedeutung haben müßten. Das andere Ergebnis bezieht sich auf die Tatsache, daß wir alles über Beale, Morriss, die Blechbüchse und die Zahlenblätter nur aus einer einzigen Quelle erfahren haben, den Aufzeichnungen des James Ward, der das zweite Blatt entziffert haben will. Kein Original der Beale-Chiffre liegt vor. Dies veranlaßte einen Linguisten, Louis Kruh, zu einer statistischen Untersuchung: Er verglich den Text des angeblich wortgetreu zitierten Briefes von Beale an Morriss mit dem übrigen, von Ward verfaßten Text und befand, daß beide in den mittleren Satzlängen auffallend übereinstimmen: In Wards Schilderungen beträgt sie 28,82 Wörter pro Satz, im Beale-Brief 28,75 Wörter. Eine so gute Übereinstimmung ist höchst ungewöhnlich. Sollte die ganze Beale-Story nur ein von Ward inszenierter Bluff sein?

Viele Leute, die das nicht glauben möchten, haben sich zu einem Verein zusammengeschlossen, der »Beale Cypher Association«. Sie schickt jedem, der ihr schreibt und zwei Dollar für die Unkosten beilegt, eine Abschrift der drei Beale-Blätter. Für den Fall, daß Sie daran interessiert sind, gebe ich Ihnen die Anschrift:

Beale Cypher Association
Box 236
Warrington, PA 18976
U.S.A.



Einer jedenfalls hat an der Beale-Affaire gut verdient: Warren Holland, ein verkrachter Bauunternehmer mit schönggeistigen Interessen. Als er im Jahre 1981 von der Geschichte erfuhr, setzte er sich hin und zählte – wie weiland Beale – einen Text aus, aber nicht die Unabhängigkeitserklärung, sondern das sechste Kapitel des Buches »Cosmos« von Carl Sagan, dem bekannten Astrophysiker und Wissenschaftsjournalisten. In dem so erhaltenen Zahlenalphabet schrieb er eines seiner Lieblingsgedichte auf, »A Poets Advice« von e e cummings. Dafür mag er gerade einen Nachmittag gebraucht haben. Es gelang Holland einen Spielzeughersteller zu finden, der die Zahlen auf eines jener Puzzles druckte, die man aus vielen Einzelteilen zusammensetzen soll. Von diesem Spiel mit dem Titel »Decipher« wurden fast eine Viertelmillion Exemplare verkauft. Sein Reiz lag darin, daß demjenigen, der die Chiffre bis März 1985 als erster geknackt hätte, ein Gewinn von hunderttausend Dollar winkte. Beim Kaufpreis von 9.95 Dollar pro Spiel war dies für Holland wahrlich kein großes Opfer. Am MIT in Cambridge, Massachusetts, hatte sich der Informatiker Alan Sherman vorgenommen, im Wintersemester 1984/85 ein Seminar für Kryptologie abzuhalten. Dafür schien ihm Hollands Chiffre ein guter Forschungsgegenstand zu sein. Also machte sich eine Gruppe junger, brillanter Wissenschaftler, die zudem über die leistungsfähigsten Computer verfügen konnten, an die Arbeit. Sie bestand zunächst darin, von einer großen Zahl Freiwilliger in Tag- und Nachtarbeit Texte in die Computer zu tippen – dies ging –

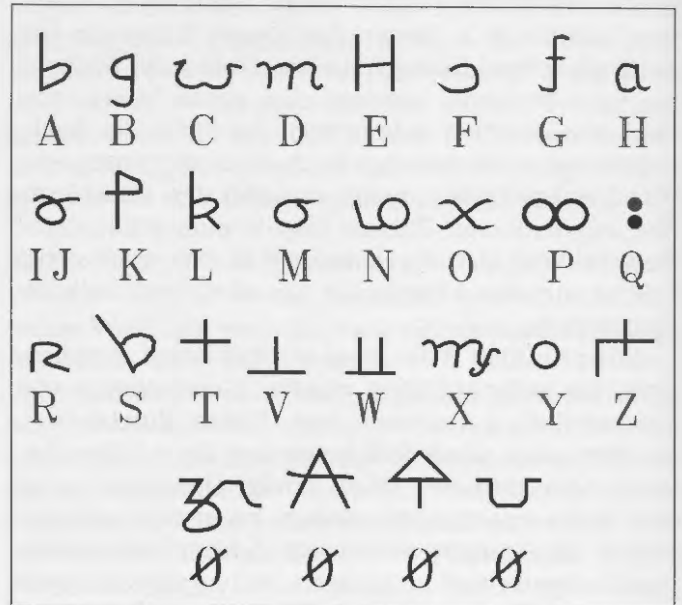
wen wundert's? – immer noch schneller als die Versuche, mittels Scanner und OCR-Software die Schriftsätze elektronisch einzulesen. Sodann wurden alle statistischen Tricks herangezogen, um die Chiffre zu brechen. Im März 85, als die Zeit knapp wurde, baten die Wissenschaftler Holland um einen Hinweis, den er auch gab. Er nannte die beiden Zahlen 3 und 19 und schrieb außerdem: »If you knew, it began with C, would it help you?«

Es half in der Tat. Der dritte und der 19. Buchstabe im Alphabet sind C und S von Carl Sagan, und das zudem genannte »C« bezieht sich auf Cosmos. Das errieten die Informatiker, und ihre Computerprogramme machten im sechsten Kapitel von »Cosmos« verräterische Textstrecken aus. Dennoch gab es noch allerlei Schwierigkeiten. Aber am späten Abend des 29. März war – endlich – die Lösung gefunden: das Gedicht von e e cummings: »A Poet's Advice«, ein ganz und gar ungewöhnlicher und darum sprachstatistisch kaum zu fassender Text. Gleich am nächsten Morgen wurde Warren Holland aus Cambridge angerufen. Er freute sich über die mitgeteilte richtige Lösung, doch den Hunderttausend-Dollar-Preis gab es nicht. Der nämlich war schon verfallen, wie im Kleingedruckten der Ausschreibung nachzulesen war. Dort hieß es, daß die Lösung bis zum letzten Werktag des März 1985 mitgeteilt werden müsse – der 30. März aber war ein Samstag. Ich fürchte, ich habe mich allzu lange mit der Beale-Story aufgehalten – ich liebe sie nun einmal –, so daß ich meine weiteren Schnappschüsse aus der Geschichte der Kryptologie auf ein paar wenige beschränken muß.

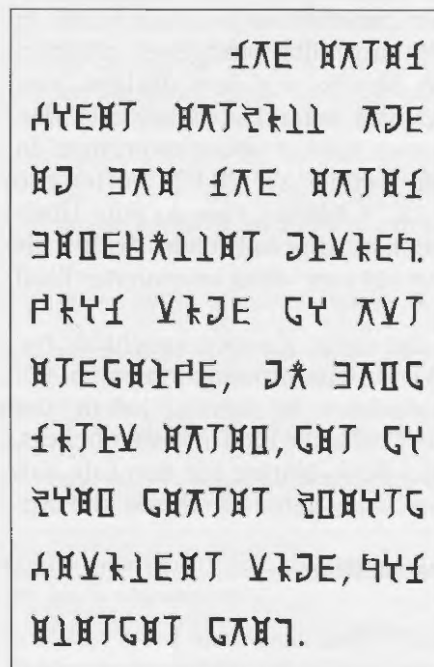
Das älteste Zeugnis einer Kryptographie (griech. kryptos = geheim), also der Verfremdung von Mitteilungen zum Zweck der Geheimhaltung, finden wir – na, wo schon – in der Bibel. Dort steht in Jeremia 25,26 und 51,41 das Wort »SESACH«. Lange haben Bibelkundige darüber nachgedacht, was dieses Wort wohl bedeuten könnte, bis ein Pfiffikus auf die goldrichtige Idee kam: Das »S« (eigentlich ist es eher ein »Sch«, weil es so gesprochen wird) ist der zweitletzte Buchstabe des hebräischen Alphabets und das »CH« ist der zwölftletzte. Wird nun der zweitletzte Buchstabe »S« an den beiden Stellen des Wortes durch den zweiten des hebräischen Alphabets ersetzt, nämlich durch das »B«, und der zwölftletzte, das »CH«, durch den zwölften hebräischen, »L«, so ergibt sich BEBAL. Die Vokale – sie spielen im Hebräischen eine untergeordnete Rolle – sind vertauscht. Also ist BABEL gemeint, das Babylon – und das macht im Zusammenhang mit dem Jeremia-Text auch Sinn.

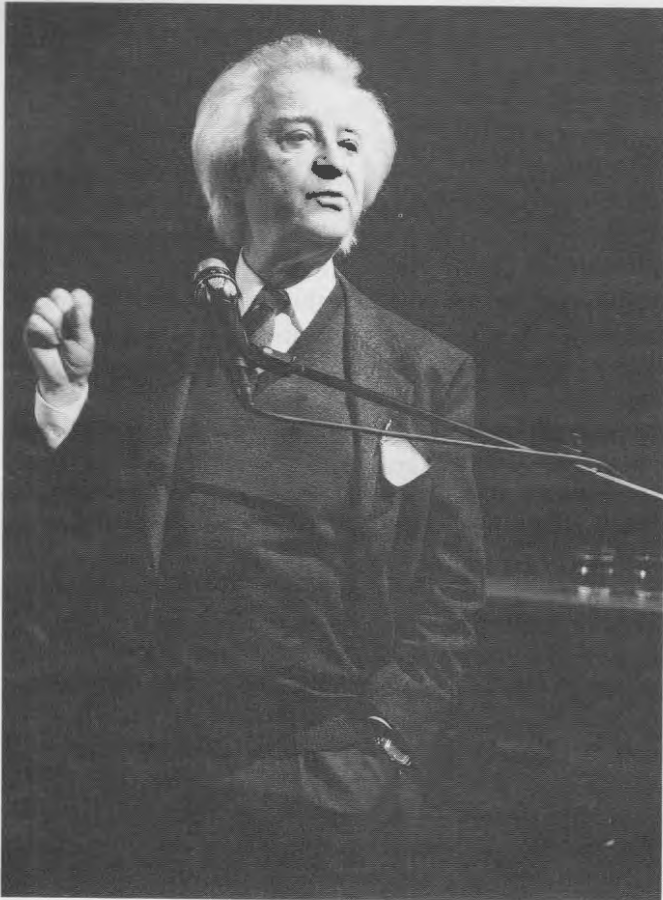
Eine frühe Anweisung zur Kryptographie hat Julius Cäsar gegeben. Er verschob ganz einfach das Alphabet um drei Buchstaben. So schrieb er für das A ein d, für B ein e, ... für W ein z, für X ein a, ... für Z ein c. Sein Name »CAESAR« wurde mithin zu »fdhvd«. Der Feldherr Cäsar verfügte, daß diese Chiffre beim Nachrichtenverkehr seiner Legionen zu verwenden sei, und als Kaiser schrieb er in diesem Code Briefe an Cicero. Offenbar hielt Cäsar seine Chiffre für unbrechbar.

Tatsächlich gehört nicht viel dazu, Cäsars Code zu knacken. Da war seine monarchische Kollegin Mary Stuart, die Königin von Schottland, schon etwas listiger. Sie verfremdete ihre Briefe mit einem von ihr selbst erdachten Alphabet. Wir sehen es hier:



Unten stehen vier zusätzliche Zeichen, die keine Bedeutung hatten. Maria Stuart verstreute diese willkürlich unter ihren Text und meinte, ihn dadurch unentzifferbar zu machen. Dummerweise benutzte sie ihre Leerzeichen außerdem als Markierung für Wortabstände, und das war ihr großer Fehler. Dies nämlich machte die Entschlüsselung leicht. Seit mehr als einem Jahr mache ich es Woche für Woche der Mary Stuart nach: Ich erfinde allerlei Alphabete und chiffriere damit kurze Texte. Sie erscheinen unter dem Namen »Findling« als Rätsel im Magazin der Frankfurter Allgemeinen Zeitung und erfreuen sich großer Beliebtheit – eben weil sie lösbar sind. Hier ein Beispiel:





Trotz Kasiskis Methode waren Polyalphabetische Substitutionen fortan nicht etwa »out«. Ist die Anzahl der Zielalphabeten sehr groß, dann braucht ein Kryptanalytiker entsprechend viel Text, um eine erfolgreiche Faktorenanalyse zu betreiben. Das haben sich die Konstrukteure der Chiffriergeräte zunutze gemacht, zum Beispiel bei der legendären ENIGMA der deutschen Streitkräfte im Zweiten Weltkrieg – sie stellte 17 576 Zielalphabeten zur Verfügung. Die Chiffre dieser Maschinen wären kaum zu knacken gewesen, wenn ...

ja, wenn die Bedienungsmannschaften nicht törichte Fehler gemacht hätten. Im zweiten Weltkrieg war ich eine Zeitlang als Soldat zum Schlüsselknacken abkommandiert worden, weil ich vor meiner Einberufung Mathematik studiert hatte – es war gerade erst ein Semester gewesen!

Interessant war diese Tätigkeit, bei der unendlich lange Buchstaben und Zahlenkolonnen an Handrechenmaschinen statistisch auszuwerten waren, nur in den Fällen, in denen uns die Dummheit der Gegenseite zur Hilfe kam. Die Engländer probierten ihre Chiffriergeräte mit dem unter Funkern bekannten Satz: »the big brown fox jumps over the lazy dog's back« aus. Danach wußten wir fast alles über die Chiffre. Die deutsche Seite stand dem an Dummheit nicht nach. In den Funkfernschreiben der Wehrmacht redete man sich höflich an: »Sehr geehrter Herr General«, und militärische Dienststellen wie etwa »Generalkommando« wurden voll ausgeschrieben. Solche Redundanz macht die listigste Chiffrierkunst zunichte.

»Den militärischen Nachrichtenverkehr«, so sagten wir damals, »darf man nicht den Militärs überlassen.« Das galt im Zweiten Weltkrieg offensichtlich auch für die Japaner, von denen die amerikanische Aufklärung wußte, daß sie einen gewaltigen Flottenangriff planten. Aus dem Funkverkehr war klar ersichtlich, daß sich der Angriff auf »AF« richten sollte, aber was in aller Welt hieß AF. Es hätte Alaska, Kalifornien, oder Midway Island heißen können. Um es herauszubekommen, veranlaßte das US-Oberkommando der Region, daß von Midway Island ein Funkpruch nach Pearl Harbor geschickt wurde mit dem verschlüsselten Text: »Wir haben hier eine Wasserknappheit«; die Japaner hatten den Schlüssel längst geknackt, was die Amerikaner wußten. Es dauerte nicht lange, da funkten die Japaner in ihrem – ebenfalls geknackten – Schlüssel: »Sie haben Wasserknappheit in AF.« Also war der Angriff auf Midway geplant – und er konnte erfolgreich abgewehrt werden.

Unterhaltsamer als die Kryptanalyse war die Suche nach steganographischen Verschlüsselungen. Darunter versteht man das Verstecken von Mitteilungen in angeblich unverfänglichen Mitteilungen. Bildpostkarten, die vom Gegner als Agenten angeheuerte Soldaten an ihre angeblichen Bräute schickten, waren besonders beliebt. Irgendwo auf dem Bild befand sich das Versteck. Es konnte zum Beispiel ein Stück Wiese mit langen und kurzen Grashalmen sein, die in Wahrheit Morsezeichen waren. Beliebt sind extrem

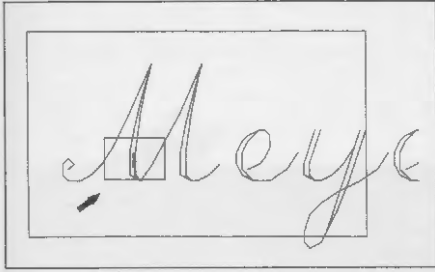
Fällt Ihnen hier etwas Verstecktes auf?

Lieber Freund,
 ich hoffe es geht Ihnen so wohl wie mir an diesem herrlichen Urlaubsort.
 Bald bin ich wieder zu Hause.
 Alles Gute
 Ihr
 Anton Meyer

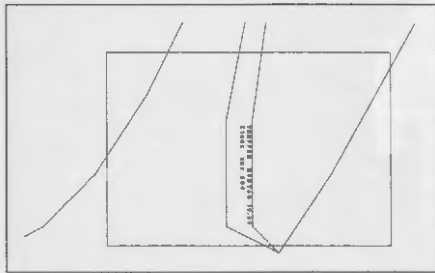
Hier am »M« ist das Versteck. Jetzt zoomen wir den eingerahmten Bereich:

Lieber Freund,
 ich hoffe es geht Ihnen so wohl wie mir an diesem herrlichen Urlaubsort.
 Bald bin ich wieder zu Hause.
 Alles Gute
 Ihr
 Anton Meyer

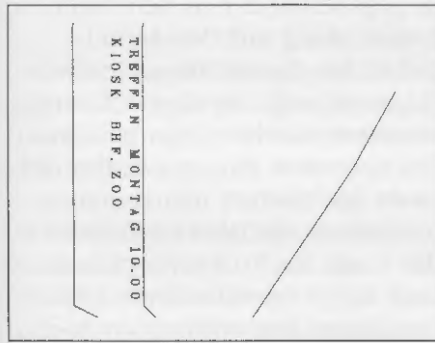
Da ist schon etwas zu sehen.



Noch etwas mehr:



Und ein weiterer Zoom bringt es an den Tag:



verkleinerte Texte – Sie glauben nicht, was man alles in einem Punkt unterbringen kann. Ich haben Ihnen ein sehr primitives Beispiel der steganographischen Verwendung eines CAD-Programms mitgebracht.

Beim Verstecken von Mitteilungen kennt die Phantasie kaum Grenzen. Einer denkbar einfachen List bedienten sich amerikanische Stäbe im Zweiten Weltkrieg; sie stellten 450 Navaho-Indianer als »Nachrichtensprecher« für den Funkverkehr in den Dienst. Deren überaus komplizierte Sprache kennt außer ihren Stammesbrüdern fast niemand. Mithin war das Abhör-Risiko verschwindend gering. Die Nordvietnamesen machten es musikalisch. Zuerst verschlüsselten sie ihre Nachrichten digital, dann setzten sie die Zeichen in Töne um, für Null einen tiefen, für Eins einen hohen. Diese »Musik« überlagerten die Chiffrierer dann noch mit anderen Tönen ehe das Tongemisch über den Rundfunk verbreitet wurde. Der Empfänger mußte die zusätzlichen Tonfolgen wieder herausfiltern, und das war nur möglich, wenn er diesen »Schlüssel« kannte. Übrig blieb das nachrichtenträchtige Gedudel.

Akustische Manipulationen spielen in der Kryptologie vor allem beim »Scrambling« eine große Rolle, der Verfremdung gesprochener Mitteilungen mittels

elektronischer Tricks. Im Oktober 1985 wurde diese Form der Steganographie jedoch zum Problem. Damals hatten palästinensische Terroristen das Schiff Archille Lauro überfallen und den amerikanischen Passagier Leon Klinghoffer ermordet – einige von Ihnen werden die Verfilmung dieses Dramas gesehen haben. Die Täter wurden in Ägypten gefaßt, aber die dortigen Behörden hatten der US-Regierung versichert, die Terroristen seien außer Landes geflohen. Das stimmte nicht. Vielmehr wußte der US-Geheimdienst, daß sich die Terroristen noch in Kairo befanden und heimlich mit einem Linienflugzeug ausgeflogen werden sollten. Um dies zu verhindern, könnten Jagdflugzeuge diese Maschine an einem Ort zur Landung zwingen, wo die Verbrecher dingfest gemacht würden. Das aber bedeutete ein Risiko für andere Passagiere in diesem Flugzeug.

Darum mußte Präsident Reagan entscheiden, ob die Aktion stattfinden sollte oder nicht. Reagan zögerte; als aber die Entscheidung nicht mehr aufzuschieben war, befand sich der Präsident in seinem Flugzeug »Air Force One«. Von hier aus wollte er nun den Befehl zum Einsatz geben – an seinen Verteidigungsminister Caspar Weinberger. Der aber befand sich just zu diesem Zeitpunkt auch in einem Flugzeug, einem der Army. Also rief Reagan von Flugzeug zu Flugzeug an. Selbstverständlich dürfen derartige delicate Gespräche nicht im Klartext gefunkt werden. Sie werden »gescrambled«. So jedenfalls sollte es sein. Doch stellte sich heraus, daß die Air Force, die den Präsidentenjet fliegt, ein Scrambling-System verwendet, das mit dem der Army nicht kompatibel ist. Mithin war Reagan gezwungen, die Details des Einsatzes mit Weinberger im Klartext zu besprechen.

Diese Unterhaltung aber hatte ein Funkamateurliebling mitgehört. Er fand sie so aufregend, daß er sofort bei der Fernsehstation CBS anrief, um die Sensation mitzuteilen. Der CBS-Redakteur war jedoch vorsichtig. Wer garantierte ihm, daß der Funkamateurliebling die Wahrheit sagte.

Ein Tonbandmitschnitt war nicht vorhanden. Also brachte CBS die Meldung nicht – und verhütete auf diese Weise, daß die Operation noch im letzten Augenblick vereitelt wurde – sie verlief übrigens ohne Schaden für die Passagiere.

Aus solchen kryptologischen Unfällen können wir schlichten Bürger, die weder Agenten noch Spione sind, durchaus Nützliches lernen. Wir hantieren mit Scheckkarten, verwalten unser Konto per Datenfernverarbeitung, sind vielleicht gar am Arbeitsplatz in Computernetze eingebunden und müssen uns deshalb mit allerlei Paßwörtern und PINs identifizieren. Die Erfahrung lehrt, daß wir diese lästigen Vorkehrungen gegen unbefugte Eingriffe im Laufe der Zeit nicht mehr so ernst nehmen, wie sie es verdienen. Das aber ist die große Chance für Unholde. Die wichtigste Erfahrung, die jeder macht, der einmal das Geschäft des Codeknackens betrieben hat, lautet: Es sind fast immer die vermeintlich harmlosen, winzigen Nachlässigkeiten, die verhindern, daß Geheimes geheim bleibt.

Zur Geschichte der Logik-Maschine

Werner Künzel

Eine universelle Maschine zur Erzeugung von Wahrheit (ein Computer) wurde schon im 13. Jahrhundert erfunden (damals noch als religiös motivierter Vorläufer). Werner Künzel entdeckte die Frühgeschichte der künstlich intelligenten Maschinen: die *Ars Magna Combinatoria et Communicativa*.

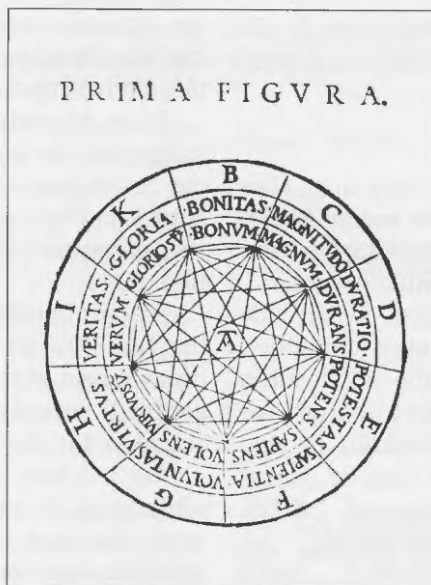
Beginnend mit den Systementwürfen des spanischen Philosophen Raymundus Lullus (1235 bis 1315) taucht im abendländischen Denken die Konzeption einer logischen Maschine auf, die alle folgenden Jahrhunderte hindurch die Diskussionen der Theoretiker beschäftigt hat. In den technischen Phantasien dieser Philosophen hat sich vorbereiten können, was wir heute unter dem Begriff der Computerkultur begreifen und zum Zeichen eines ganzen Zeitalters proklamiert haben.

Die Ars Combinatoria des Raymundus Lullus

Raymundus Lullus entwickelte eine »Ars Generalis Ultima« zum Begreifen der Offenbarungswahrheit: Mit Hilfe von verknüpften



geometrischen Figuren, die in einem präzise definierten Regelwerk miteinander kommunizieren, sollten alle nur möglichen gültigen Aussagen erzeugt und in Zeichenketten repräsentiert werden. Diese logische Kombinatorik sollte zur Missionierung al-



einzel in Kammern angeordnet, so daß ein Drehen der Kreise gegeneinander in beliebiger Leserichtung auf den Mittelpunkt hin immer neue Zeichenfolgen erzeugt. In diesen Kombinationen taucht so der gesamte Zeichenvorrat des von Lullus auf neun Buchstaben reduzierten lateinischen Alphabets auf. Jeder der neun Buchstaben repräsentiert auf je verschiedenen Ebenen einen fest umrissenen Bedeutungsgehalt. Mit der Vollzähligkeit der gesetzten Bedeutungsebenen steht und fällt der Wahrheitsanspruch dieses logischen Systems, denn nur die Bedeutungsgehalte können kombiniert werden, die vorher als Material bestimmt wurden.

Die Ars Magna Sciendi des Athanasius Kircher

Zur Zeit des Dreißigjährigen Kriegs und des Aufruhrs in der Theoriebildung aller Wissenschaften ging der deutsche Jesuit Athanasius Kircher (1602 bis 1680) daran, in Anknüpfung an Lullus' »Ars Combinatoria« eine universale Entzifferung des verborgenen Wissens längst vergangener Kulturen zu betreiben. Kirchers »Ars Magna Sciendi« ist aber eine radikale Umdeutung der dreihundert Jahre älteren Begründung kombinatorischer Logik. Er

ler Andersgläubigen Beweise liefern, die nicht länger an den Gültigkeitsbereich der natürlichen Sprache des Argumentierenden gebunden wären.

Eine dreifach gegliederte Kreisfigur bildet das zentrale Stück der »Ars Combinatoria«. Auf ihren ineinander geschachtelten drehbaren Kreisen sind alle zur Verfügung stehenden Zeichen

five COMBINATORIÆ, LIB. IV.
TABULA GENERALIS.
*Ex qua omnes rerum simpliciter combinatoriarum coniugationes perfidetur
citantur.*

1.	A.
2.	B.
3.	C.
4.	D.
5.	E.
6.	F.
7.	G.
8.	H.
9.	I.
10.	K.
11.	L.
12.	M.
13.	N.
14.	O.
15.	P.
16.	Q.
17.	R.
18.	S.
19.	T.
20.	V.
21.	X.
22.	Y.
23.	Z.
24.	aa.
25.	ab.
26.	ac.
27.	ad.
28.	ae.
29.	af.
30.	ag.
31.	ah.
32.	ai.
33.	aj.
34.	ak.
35.	al.
36.	am.
37.	an.
38.	ao.
39.	ap.
40.	aq.
41.	ar.
42.	as.
43.	at.
44.	au.
45.	av.
46.	aw.
47.	ax.
48.	ay.
49.	az.
50.	ba.
51.	bb.
52.	bc.
53.	bd.
54.	be.
55.	bf.
56.	bg.
57.	bh.
58.	bi.
59.	bj.
60.	bk.
61.	bl.
62.	bm.
63.	bn.
64.	bo.
65.	bp.
66.	bq.
67.	br.
68.	bs.
69.	bt.
70.	bu.
71.	bv.
72.	bw.
73.	bx.
74.	by.
75.	bz.
76.	ca.
77.	cb.
78.	cc.
79.	cd.
80.	ce.
81.	cf.
82.	cg.
83.	ch.
84.	ci.
85.	cj.
86.	ck.
87.	cl.
88.	cm.
89.	cn.
90.	co.
91.	cp.
92.	cq.
93.	cr.
94.	cs.
95.	ct.
96.	cu.
97.	cv.
98.	cw.
99.	cx.
100.	cy.
101.	cz.
102.	da.
103.	db.
104.	dc.
105.	dd.
106.	de.
107.	df.
108.	dg.
109.	dh.
110.	di.
111.	dj.
112.	dk.
113.	dl.
114.	dm.
115.	dn.
116.	do.
117.	dp.
118.	dq.
119.	dr.
120.	ds.
121.	dt.
122.	du.
123.	dv.
124.	dw.
125.	dx.
126.	dy.
127.	dz.
128.	ea.
129.	eb.
130.	ec.
131.	ed.
132.	ee.
133.	ef.
134.	eg.
135.	eh.
136.	ei.
137.	ej.
138.	ek.
139.	el.
140.	em.
141.	en.
142.	eo.
143.	ep.
144.	eq.
145.	er.
146.	es.
147.	et.
148.	eu.
149.	ev.
150.	ew.
151.	ex.
152.	ey.
153.	ez.
154.	fa.
155.	fb.
156.	fc.
157.	fd.
158.	fe.
159.	ff.
160.	fg.
161.	fh.
162.	fi.
163.	fj.
164.	fk.
165.	fl.
166.	fm.
167.	fn.
168.	fo.
169.	fp.
170.	fq.
171.	fr.
172.	fs.
173.	ft.
174.	fu.
175.	fv.
176.	fw.
177.	fx.
178.	fy.
179.	fz.
180.	ga.
181.	gb.
182.	gc.
183.	gd.
184.	ge.
185.	gf.
186.	gg.
187.	gh.
188.	gi.
189.	gj.
190.	gk.
191.	gl.
192.	gm.
193.	gn.
194.	go.
195.	gp.
196.	gq.
197.	gr.
198.	gs.
199.	gt.
200.	gu.
201.	gv.
202.	gw.
203.	gx.
204.	gy.
205.	gz.
206.	ha.
207.	hb.
208.	hc.
209.	hd.
210.	he.
211.	hf.
212.	hg.
213.	hh.
214.	hi.
215.	hj.
216.	hk.
217.	hl.
218.	hm.
219.	hn.
220.	ho.
221.	hp.
222.	hq.
223.	hr.
224.	hs.
225.	ht.
226.	hu.
227.	hv.
228.	hw.
229.	hx.
230.	hy.
231.	hz.
232.	ia.
233.	ib.
234.	ic.
235.	id.
236.	ie.
237.	if.
238.	ig.
239.	ih.
240.	ii.
241.	ij.
242.	ik.
243.	il.
244.	im.
245.	in.
246.	io.
247.	ip.
248.	iq.
249.	ir.
250.	is.
251.	it.
252.	iu.
253.	iv.
254.	iw.
255.	ix.
256.	iy.
257.	iz.
258.	ja.
259.	jb.
260.	jc.
261.	jd.
262.	je.
263.	jf.
264.	jj.
265.	jk.
266.	jl.
267.	jm.
268.	jn.
269.	jo.
270.	jp.
271.	jq.
272.	jr.
273.	js.
274.	jt.
275.	ju.
276.	jv.
277.	jw.
278.	jx.
279.	jy.
280.	jz.
281.	ka.
282.	kb.
283.	kc.
284.	kd.
285.	ke.
286.	kf.
287.	kg.
288.	kh.
289.	ki.
290.	kj.
291.	kl.
292.	km.
293.	kn.
294.	ko.
295.	kp.
296.	kq.
297.	kr.
298.	ks.
299.	kt.
300.	ku.
301.	kv.
302.	kx.
303.	ky.
304.	kz.
305.	la.
306.	lb.
307.	lc.
308.	ld.
309.	le.
310.	lf.
311.	lg.
312.	lh.
313.	li.
314.	lj.
315.	lk.
316.	ll.
317.	lm.
318.	ln.
319.	lo.
320.	lp.
321.	lq.
322.	lr.
323.	ls.
324.	lt.
325.	lu.
326.	lv.
327.	lw.
328.	lx.
329.	ly.
330.	lz.
331.	ma.
332.	mb.
333.	mc.
334.	md.
335.	me.
336.	mf.
337.	mg.
338.	mh.
339.	mi.
340.	mj.
341.	mk.
342.	ml.
343.	mm.
344.	mn.
345.	mo.
346.	mp.
347.	mq.
348.	mr.
349.	ms.
350.	mt.
351.	mu.
352.	mv.
353.	mw.
354.	mx.
355.	my.
356.	mz.
357.	na.
358.	nb.
359.	nc.
360.	nd.
361.	ne.
362.	nf.
363.	ng.
364.	nh.
365.	ni.
366.	nj.
367.	nk.
368.	nl.
369.	nm.
370.	nn.
371.	no.
372.	np.
373.	nq.
374.	nr.
375.	ns.
376.	nt.
377.	nu.
378.	nv.
379.	nw.
380.	nx.
381.	ny.
382.	nz.
383.	oa.
384.	ob.
385.	oc.
386.	od.
387.	oe.
388.	of.
389.	og.
390.	oh.
391.	oi.
392.	oj.
393.	ok.
394.	ol.
395.	om.
396.	on.
397.	oo.
398.	op.
399.	oq.
400.	or.
401.	os.
402.	ot.
403.	ou.
404.	ov.
405.	ow.
406.	ox.
407.	oy.
408.	oz.
409.	pa.
410.	pb.
411.	pc.
412.	pd.
413.	pe.
414.	pf.
415.	pg.
416.	ph.
417.	pi.
418.	pj.
419.	pk.
420.	pl.
421.	pm.
422.	pn.
423.	po.
424.	pp.
425.	pq.
426.	pr.
427.	ps.
428.	pt.
429.	pu.
430.	pv.
431.	pw.
432.	px.
433.	py.
434.	pz.
435.	qa.
436.	qb.
437.	qc.
438.	qd.
439.	qe.
440.	qf.
441.	qg.
442.	qh.
443.	qi.
444.	qj.
445.	qk.
446.	ql.
447.	qm.
448.	qn.
449.	qo.
450.	qp.
451.	qq.
452.	qr.
453.	qs.
454.	qt.
455.	qu.
456.	qv.
457.	qw.
458.	qx.
459.	qy.
460.	qz.
461.	ra.
462.	rb.
463.	rc.
464.	rd.
465.	re.
466.	rf.
467.	rg.
468.	rh.
469.	ri.
470.	rj.
471.	rk.
472.	rl.
473.	rm.
474.	rn.
475.	ro.
476.	rp.
477.	rq.
478.	rr.
479.	rs.
480.	rt.
481.	ru.
482.	rv.
483.	rw.
484.	rx.
485.	ry.
486.	rz.
487.	sa.
488.	sb.
489.	sc.
490.	sd.
491.	se.
492.	sf.
493.	sg.
494.	sh.
495.	si.
496.	sj.
497.	sk.
498.	sl.
499.	sm.
500.	sn.
501.	so.
502.	sp.
503.	sq.
504.	sr.
505.	ss.
506.	st.
507.	su.
508.	sv.
509.	sw.
510.	sx.
511.	sy.
512.	sz.
513.	ta.
514.	tb.
515.	tc.
516.	td.
517.	te.
518.	tf.
519.	tg.
520.	th.
521.	ti.
522.	tj.
523.	tk.
524.	tl.
525.	tm.
526.	tn.
527.	to.
528.	tp.
529.	tq.
530.	tr.
531.	ts.
532.	tt.
533.	tu.
534.	tv.
535.	tw.
536.	tx.
537.	ty.
538.	tz.
539.	ua.
540.	ub.
541.	uc.
542.	ud.
543.	ue.
544.	uf.
545.	ug.
546.	uh.
547.	ui.
548.	uj.
549.	uk.
550.	ul.
551.	um.
552.	un.
553.	uo.
554.	up.
555.	uq.
556.	ur.
557.	us.
558.	ut.
559.	uu.
560.	uv.
561.	uw.
562.	ux.
563.	uy.
564.	uz.
565.	va.
566.	vb.
567.	vc.
568.	vd.
569.	ve.
570.	vf.
571.	vg.
572.	vh.
573.	vi.
574.	vj.
575.	vk.
576.	vl.
577.	vm.
578.	vn.
579.	vo.
580.	vp.
581.	vq.
582.	vr.
583.	vs.
584.	vt.
585.	vu.
586.	vv.
587.	vw.
588.	vx.
589.	vy.
590.	vz.
591.	wa.
592.	wb.
593.	wc.
594.	wd.
595.	we.
596.	wf.
597.	wg.
598.	wh.
599.	wi.
600.	wj.
601.	wk.
602.	wl.
603.	wm.
604.	wn.
605.	wo.
606.	wp.
607.	wq.
608.	wr.
609.	ws.
610.	wt.
611.	wu.
612.	wv.
613.	ww.
614.	wx.
615.	wy.
616.	wz.
617.	xa.
618.	xb.
619.	xc.
620.	xd.
621.	xe.
622.	xf.
623.	xg.
624.	xh.
625.	xi.
626.	xj.
627.	xk.
628.	xl.
629.	xm.
630.	xn.
631.	xo.
632.	xp.
633.	xq.
634.	xr.
635.	xs.
636.	xt.
637.	xu.
638.	xv.
639.	xw.
640.	xx.
641.	xy.
642.	xz.
643.	ya.
644.	yb.
645.	yc.
646.	yd.
647.	ye.
648.	yf.
649.	yg.
650.	yh.
651.	yi.
652.	yj.
653.</	

Die Muse in der Steckdose

David Galloway

Viele Künstler der klassischen Moderne versuchten, industrielle Methoden und Materialien mit herkömmlichen Mitteln zu einem Gesamtkunstwerk zu verbinden. Heute holen sich die Künstler einen Computer aus dem Supermarkt.



Dank internationaler Ausstellungen, Kongresse und Wettbewerbe ist der Ausdruck »Computergraphik« ein gängiger Kunstbegriff geworden. Auch wenn dies auf eine wachsende Akzeptanz der elektronischen Avantgarde deutet, ist es dennoch keine glückliche Formulierung. Für ein breiteres Publikum deutet Computergraphik auf eine mechanische, seelenlose Bildherstellung, die allein von Hard- und Software abhängt. Darüber hinaus wird die Terminologie mit gleicher Bedeutung auf kippende Fernsehlogos, Nasa-Simulationen und technische Zeichnungen angewandt. Und tatsächlich wurde der Ausdruck »Computergraphik« 1960 von William A. Fetter, einem For-

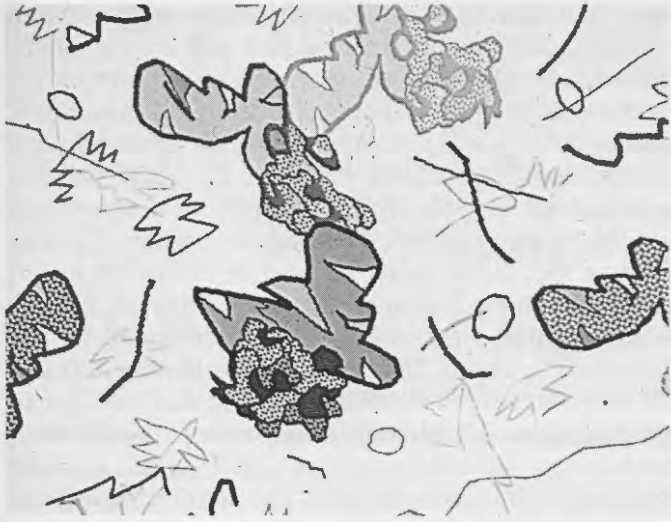
scher des Flugzeugherstellers Boeing in Renton, Washington, gemünzt, um seine rein technischen, computergenerierten Plotterzeichnungen eines Flugzeugcockpits zu beschreiben. »Jedermann ein Künstler«, laut Joseph Beuys, jeder Programmierer ein Picasso. Doch Picassos Genie hing nicht von den Werkzeugen ab, die ihm zur Verfügung standen. Vielmehr prägte er jedes Werkzeug und jedes Material – ob Farbe, Draht, Zeitungspapier oder Strandgut – mit seiner unverkennbaren Handschrift.

Bei allen Unterschieden im persönlichen Ausdruck, in Haltungen und Zielsetzungen teilen die Verfechter der Computerkunst eine Faszination mit neuen »Tools« und Materialien. Vielleicht ist das einer der wenigen Fixpunkte in der ästhetischen Fahrt mit der Achterbahn, die die Kunst dieses ausgehenden Jahrhunderts charakterisiert. Noch sind es lediglich solche posttechnologischen Wunder wie Laser oder Mikrochip, die weiter um sich greifende Wechsel in der bildlichen Erfahrungswelt auslösen.

Die einfachen Tubenfarben befreiten die Impressionisten aus ihren Ateliers und ermöglichten ihnen, in der Natur zu malen. Die wasserlöslichen, schnell-trocknenden Hausfarben, die 1954 auf den amerikanischen Markt kamen, lösten einen neuen bildlichen Gestus aus. Dank dieser Akrylfarben, wie sie bald genannt wurden, konnten Andy Warhol und Roy Lichtenstein ihren Konsum-Kultur-Ikonen jene flachen, anonymen Oberflächen der amerikanischen Werbung verleihen. Später sollte gar die schnöde Sprühdose jene monumentalen Graffiti in der New Yorker U-Bahn ermöglichen.

So verwundert es kaum, daß junge Künstler, die mit den elektronischen Medien groß geworden sind, diese ästhetisch erforschen wollen. Es ist das erste Mal, daß der Künstler sich des technischen Fortschritts bedient, ihn in Frage stellt oder vielleicht sogar mißbraucht. Das 19. Jahrhundert ist reich an Beispielen, wie Künstler die industriellen Früchte stibitzten, wenn auch nicht immer mit Erfolg. Die industrielle Herstellung von Asphalt – eine Folge der schnellen Urbanisierung – führte zu der glücklosen Mode der »Asphaltmalerei«. Das neue Medium wurde besonders an der Düsseldorfer Kunstakademie bevorzugt, deren Studenten es in der ganzen westlichen Welt verbreiteten. Leider haben Gemälde mit einer Grundierung aus Asphalt den Hang, sich mit der Zeit wie asphaltierte Straßen zu verhalten, Farbänderungen, Brüche, Verwerfungen und allgemeine Zerfallserscheinungen aufzuweisen. Restauratoren konnten bislang keine Kur für solche Leiden finden.

Wenngleich als Malgrund falsch angewandt, so nimmt Asphalt doch in der Kunstgeschichte eine respektable Stellung ein. Rembrandt benutzte Asphalt, um Glasuren für seine Gemälde vorzubereiten, und noch heute wird er zuweilen als Grund für Radierungen eingesetzt. Mit wenigen Ausnahmen, so könnte man folglich argumentieren, haben neue Medien die alten nicht ersetzt, auch wenn manche eine Zeitlang in Vergessenheit geraten waren. Als Warhol und Lichtenstein ihre Experimente mit Akrylfarben anfangen,

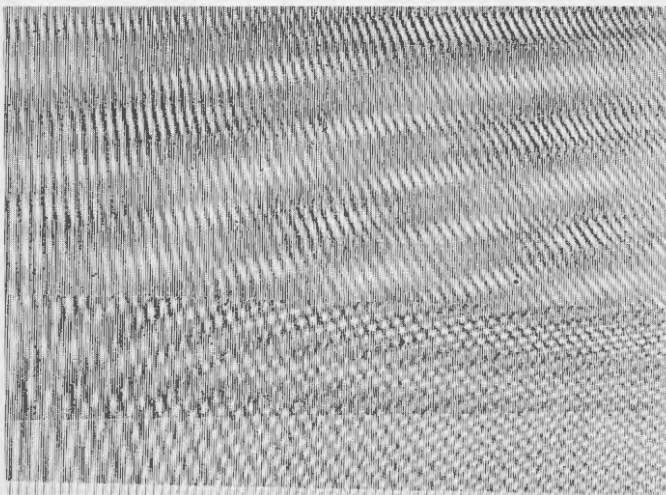


Jim McLean, »Einmal fix gedreht«, 1988. Siebdruck nach computergenerierten Figuren. 26 x 33 cm

überhäufte Willem de Kooning seine belgischen Leinwand mit satter Ölfarbe, und Jasper Johns erforschte das uralte, komplexe, zeitaufwendige Medium Enkaustik, das in der hellenistischen Portraitalmalerei seinen vorläufigen Höhepunkt erreicht hatte.

Das Aufkommen elektronischer Werkzeuge macht ihre Vorläufer nicht überflüssig. Auch heute zeichnet Konrad Klapheck mit den gleichen Kohlezweigen, die der vorgeschichtliche Mensch benutzte, um seinen Höhlenmalereien Konturen zu geben. Kein Medium ist einem anderen »überlegen«, auch wenn jedes seine zeitgemäße Anwendung hat und seine eigene Formsprache (wie in der Pop Art oder in der Graffiti-Kunst) herausbildet. Jedes befriedigt auf seine Weise den ewigen Hunger nach Bildern. So scheint es nur logisch, daß eine Generation, die mit Video und Computerspielen groß wurde, eine ästhetische Anwendung der Werkzeuge einer posttechnologischen Gesellschaft sucht. Sie fahren also in jenem Geist der Einbeziehung fort, der die Kunst des 20. Jahrhunderts kennzeichnet, seit Picasso seine ersten Assemblagen fertigte. Und sie können sich auf würdige Vorläufer in den progressiven Visionen der Futuristen und

Eudice Feder, »Pfützen«, 1986. Inkjet-Computerzeichnung. 50 x 63,5 cm

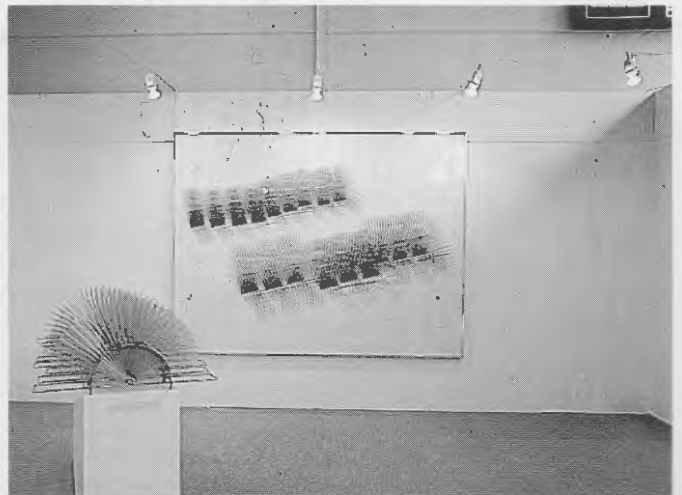


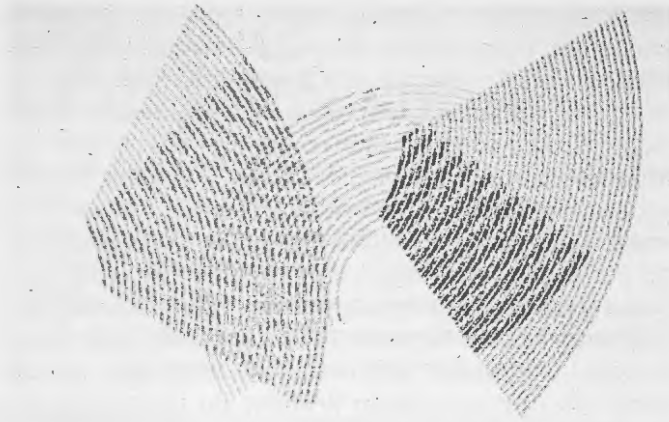
Konstruktivisten berufen. Diese Pioniere träumten von einer Welt, in der die neuen industriellen Methoden und Ressourcen mit herkömmlichen Mitteln zu einem Gesamtkunstwerk sich verbinden sollten. Heute rücken solche Visionen greifbar nah. Dank der Mikroelektronik, der vereinfachten Software, benutzerfreundlichem Design und wachsender Konkurrenz unter Hardware-Herstellern ist der Personal-Computer auf dem besten Wege, ein alltägliches, erschwingliches Werkzeug zu werden. Zum Vergleich: Als der Bildhauer Milton Komisar 1973 versuchte, sich einen Personal-Computer anzuschaffen, war der einzig gangbare Weg, sich einen Rechner im Do-it-yourself-Verfahren zusammenzubasteln.

Heute holt man seinen elektronischen Assistenten aus dem Supermarkt im Sonderangebot. Genauso fing der Graphiker und Kunsterzieher Jim McLean aus Chamblee, Georgia, an. Neben herkömmlichen Techniken wollte er seinen Studenten einen posttechnologischen Vorsprung geben, ohne selber jemals mit solchen Mitteln gearbeitet zu haben. Auch wenn sein Billig-Atari ihm nur grob gerasterte Figuren mit der Maus ermöglichte, hatte er das Gefühl, eine neue Bildersprache zu entdecken. Statt die kantigen Konturen zu vertuschen, sah er sie eher als Herausforderung an, schnitt sie aus und klebte die Formen zu pfiffig gedrehten Collagen. McLean beweist, wie auch die 73jährige Künstlerin Eudice Feder, daß die Muse in der Steckdose nicht nur auf dreiste Computerkids hört.

Eudice Feder war bereits eine anerkannte Malerin, als sie im Alter von 60 Jahren Informatik zu studieren begann. Heute erschafft sie zarte, moiréhafte Zeichnungen mit Hilfe eines Programms, das SIMULA heißt und darüber hinaus die offizielle Computersprache der skandinavischen Streitkräfte ist. »Es macht mir immer Freude, wenn Künstler etwas positiv einsetzen, was potentiell eine erschreckende Technologie darstellt«. Darin klingt die Hoffnung des Propheten Isaias, Schwerter in Pflugscharen zu verwandeln. Feders Erfahrung ist keine Ausnahme; viele elektronische Weiterentwicklungen stammen, kaum

Wolfgang Zach, ohne Titel, 1989. Skulptur mit Plotterzeichnung





Anna Solecka-Zach, ohne Titel, 1989. Plotterzeichnung; Buntstift und Ölkreide auf Papier. 100 x 150 cm

überraschend, aus der Militär- oder Weltraumforschung. Der Bildhauer Wen-Ying Tsai erinnert sich noch genau an den Schock, den er verspürte, als er erfuhr, »daß die Auslösemechanismen, die ich verwende, um das Strobe-Licht und den Puls meiner kybernetischen Skulpturen zu synchronisieren, im Grunde die gleichen sind, die die frühen Atombomben zündeten.«

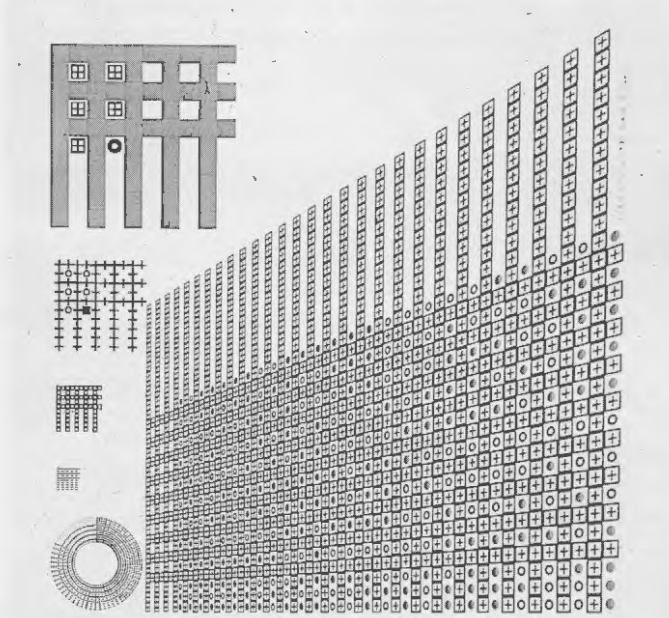
Wie Jim McLean und Eudice Feder entdecken viele Künstler die ästhetischen Eigenschaften des Computers aus reiner Neugier. Bei dem deutschen Künstler Wolfgang Zach war der Weg entgegengesetzt. Erst belegte er Informatik, bevor er sich für Bildende Kunst an der Karlsruher Hochschule einschrieb. Fasziniert von den Eigenschaften des Lichts begann er Skulpturen aus feinem, schimmerndem Draht zu bauen, die die von einem zentralen Punkt ausgehenden Lichtstrahlen andeuteten. Diese Kompositionen können häufig vom Betrachter verändert werden, indem er die Achse, um die sich die Drähte reihen, verschiebt. Als Zachs Arbeiten immer komplizierter wur-

den, brauchte er in steigendem Maße präzise Zeichnungen, um sicher zu stellen, daß sich die Drähte in allen Lagen frei bewegen. Während er sich mit einem Satz neuer Entwürfe quälte, fiel dem Bildhauer ein, daß er dieselben Ergebnisse wesentlich effizienter mit einem Computer erstellen könne. Er kaufte sich ein gebrauchtes Modell und entwickelte zusammen mit einem Programmierer seine eigene Software. Nachdem er die ganze Breite möglicher Bewegungen auf dem Bildschirm betrachtet hatte, wählte er die befriedigendsten Figurationen, um sie über den Plotter zeichnen zu lassen. Die neue Technik bewies sich also als mehr als nur eine Abkürzung. Die dichten Linienzeichnungen offenbaren eine innere Poesie, und Zach zeigte sie bald zusammen mit seinen fertigen Skulpturen in Ausstellungen. Die kleinen Ausdrücke konnten jedoch kaum die Monumentalität seiner Skulpturen andeuten.

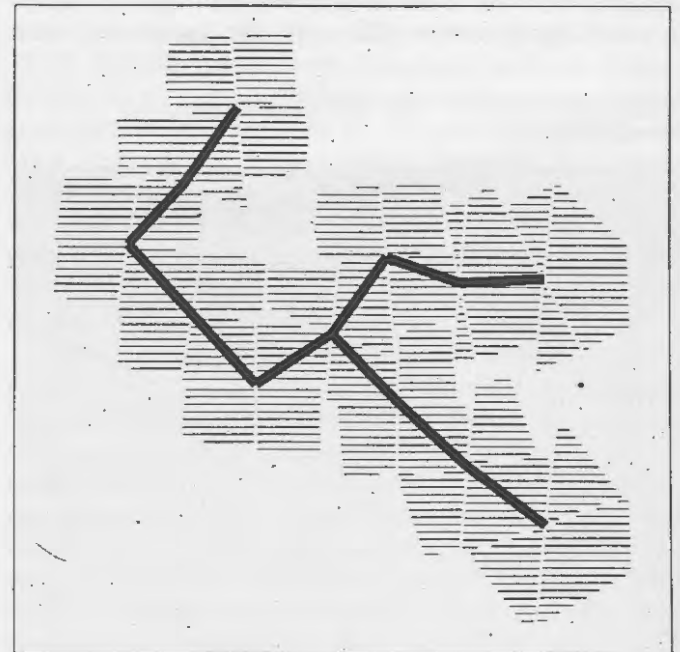
Ein neues Programm hat die Möglichkeiten jetzt erweitert: Jeder Ausdruck ist ein horizontales Segment der Gesamtzeichnung, so daß die Bildgröße praktisch unbegrenzt ist, und die zusammengesetzten Bildsegmente reihen sich mit einer Präzision aneinander, die selbst ausgefuchste Techniker überrascht. Zu Zachs größten Kritikern gehörte seine Frau, die abstrakte Malerin Anna Zolecka, die anfangs den Computer als Feind ihres eigenen, intuitiven Kompositionsansatzes betrachtete. Probesthalber entwickelte dann ihr Mann ein Programm, das ihre eigene, ausgeprägte Formsprache reflektierte; heute streiten sie sich um die »Computer-Zeit«.

Mit illustrer Ahnenschaft und wachsendem Zugang scheint es zunächst erstaunlich, daß die elektronische Avantgarde noch immer so wenig Zuspruch findet. Kuratoren fürchten häufig lästige Installationen. Galeristen sehen kaum Profit in der »Kunst aus der Steckdose«. Kritiker, gefangen in ihrem vorge-

Mark Wilson, ohne Titel, 1988. Plotterzeichnung; Tusche auf Papier, 91 x 91 cm



Manfred Mohr, »P-395-B«, 1986. Plotterzeichnung auf Papier. 60 x 60 cm





Harold Cohen, ohne Titel, 1986. Computergenerierte Zeichnung, India Tusche auf Papier, 56,5 x 77 cm



Harold Cohen, ohne Titel, 1986. Computergenerierte Zeichnung, India Tusche auf Papier, handkoloriert. 77 x 111 cm

prägten Vokabular, werfen wiederholt die matte Frage auf: »Aber ist das Kunst?« Solche Haltungen erinnern an eine Zeit, als Fotografie als kalte, mechanische Form der Reproduktion verunglimpft wurde, die keine müde Muse auch nur zum Augenzwinkern bewegen konnte.

Ein Textverarbeitungssystem macht noch keinen zu einem guten Romancier, aber es hilft einem guten Autoren, die sich wiederholenden Aufgaben zu vermeiden und sich besser, intensiver auf seine kreativen Energien zu konzentrieren. Ein Computer kann keine Kunst produzieren, aber er kann einzelnen Künstlern helfen, ihre ästhetischen Impulse zu verwirklichen. Die Vorstellung, daß ein Computer unausweichlich kalte, roboterhafte Ergebnisse hervorbringe, ist ein Klischee, das von den Leistungen einer neuen Generation eindeutig widerlegt wird.

Solche Klischees verdrängen ebenfalls ein zentrales Problem der elektronischen Bildherstellung. Jeder hier erwähnte Künstler »malt« auf dem Bildschirm. Manche beginnen mit fotografischen Vorlagen, andere erzeugen ihre Kompositionen rein rechnerisch, doch jeder schafft entmaterialisierte Arbeiten, die nicht im üblichen Sinne ausgestellt oder gesammelt werden können. Der Künstler muß daher entscheiden, wie er sein Endresultat »übersetzt« und vermittelt. Einige lassen den Plotter rattern, andere fotografieren direkt den Bildschirm, ziehen Hardcopies oder produzieren Ektachrome. Über Scanachrome kann auch ein normales Diapositiv zu einer großformatigen Leinwand werden. Diese Vielfalt an technischen Möglichkeiten führt uns in einen Grenzbereich, wo die wahre Ästhetik der Neuen-Medien-Künste erforscht wird.

Es wird immer deutlicher, daß es kein Genre, keinen Stil – realistisch, expressionistisch, abstrakt, geometrisch oder konzeptuell – gibt, der nicht mit Hilfe der elektronischen Medien ausgedrückt werden kann. Sicher, selbst in einer posttechnologischen Welt ist weniger manchmal mehr. Und die Ölmalerei mag noch immer überzeugendere Ergebnisse hervorbringen als die Elektronik. Die notwendigen Richtlinien werden jedoch nur entwickelt, wenn die elektronische Avantgarde ein adäquates Forum gefunden hat. Es geht hier nicht darum, ein neues Medium auf Ko-

sten eines alten voranzutreiben, sondern darum, ihre Gleichwertigkeit als Potential künstlerischen Ausdrucks zu gewährleisten. Es geht auch darum, den Computer zu entmystifizieren, ihn als Hilfsmittel zu sehen. Auch das elektronische Werkzeug ist nicht besser als die Hand, die es führt.

Man muß nur sicher sein, daß die Werkzeuge auch dieser Meinung sind. Bei den neuesten der Neuen Medien geht es jedoch um Werkzeuge, deren volles Potential selbst die Techniker nicht immer begreifen. Früher wurden Tools entwickelt, um ganz bestimmte Aufgaben zu meistern, spezifische Probleme zu lösen. Heute werden Werkzeuge erschaffen, deren Anwendung erst noch erforscht und entwickelt werden muß; andere, im Gegensatz zu Kamelhaarpinsel oder Kamera, sind lernfähig und weiten so ihre eigene Kapazität aus. So hat etwa der englische Maler Harold Cohen eine Maschine erfunden, die AARON heißt und großformatige Arbeiten auf Papier, teils selbständig, produziert. AARON wurde mit jeder Nuance von Cohens eigenem Zeichenstil programmiert ebenso wie mit seiner grundsätzlichen Haltung in Bezug auf Komposition. In seiner letzten und fortgeschrittensten Version beginnt AARON, an Schlüsselpunkten der Bildkomposition zu zögern und dann Entscheidungen zu treffen, die Cohen möglicherweise selbst nicht getroffen hätte, wengleich sie mit dem Werk des Künstlers übereinstimmen.

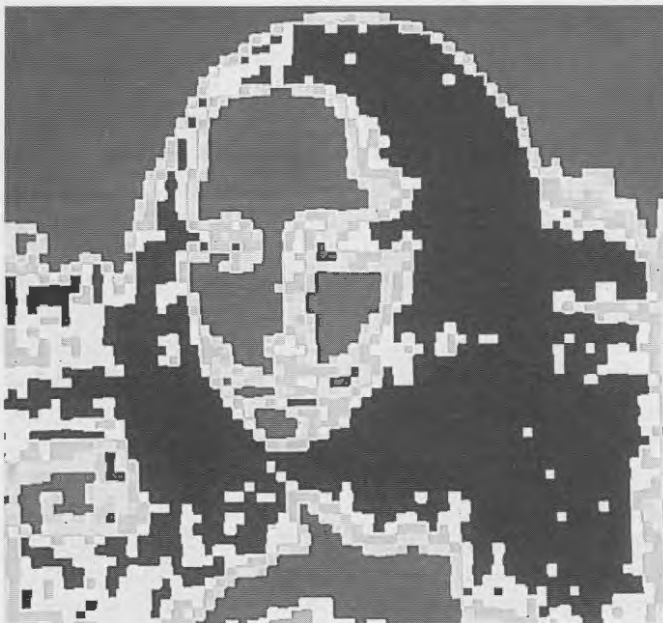
Jede Arbeit, die AARON produziert, ist ein Original, und es gibt keinen Grund, warum er nicht Zeichnungen produzieren sollte, wenn Harold Cohen längst gestorben ist. Trotz solcher elektronischer »Wunder« sind wir glücklicherweise noch weit entfernt von der Zeit, in der Kunst sprichwörtlich von Maschinen hergestellt wird. AARON ist schließlich nicht mehr als Cohens cleverer Gehilfe. Kunst wird von Künstlern gemacht, egal ob ihre Werkzeuge die eigenen Finger oder ein Computer mit Plotter sind, genauso wie Romane von Romanciers und nicht von Schreibsystemen verfaßt werden. Dennoch bewegen wir uns in Bereich des kreativen Ausdrucks, für den die traditionellen Vorläufer rapide an Bedeutung verlieren. Zweifelsohne stehen wir, mit den Worten von Herbert W. Franke, an »einem Wendepunkt in der Kunstgeschichte«.

Die Ästhetik der programmierten Kunst

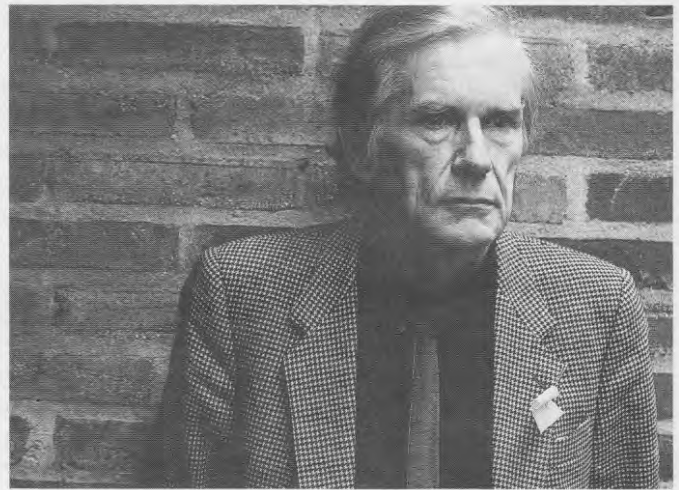
Herbert W. Franke

Computergraphiken bringen ästhetische Überraschungseffekte in Videos und Werbespots. Aber kunstfertige Künstlichkeit macht noch keine Kunst. Der Mitbegründer der »Ars Electronica« berichtet, was Künstler aus dieser Technik herausholen.

Computergraphik gibt es schon seit mehr als fünf- undzwanzig Jahren. Neben wissenschaftlichen, technischen und kommerziellen Anwendungen gab es von Anfang an auch Versuche, das neue Medium für künstlerische Zwecke einzusetzen. Mitte der sechziger Jahre experimentierten die beiden Deutschen Frieder Nake und Georg Nees sowie der Amerikaner A. Michael Noll mit Computerkunst. Ihnen folgten mehrere Einzelpersonlichkeiten und Gruppen, beispielsweise Kenneth Knowlton sowie das Team Charles Csurik und James Schaffer in den USA und die japanische Computer Technique Group. Sie alle waren 1968 in der großen Ausstellung »Cybernetic Serendipity« in London vertreten. In der Zeit danach eigneten sich neben Mathematikern und Programmierern mehr und mehr auch ausgebildete Künstler die Methode der Computergraphik an, die zu einer internationalen Aktivität wurde, doch in der Öffentlichkeit wenig Beachtung fand. Dies ändert sich erst seit einigen Jahren – weniger infolge eines Durchbruchs auf künstlerischem Gebiet als vielmehr aufgrund spektakulärer computergenerierter Tricksequenzen in Science-Fiction-Filmen und Werbespots.



Als technische Methode hat Computergraphik mit Kunst zunächst nicht mehr zu tun als der Gebrauch von Zeichenstift und Farbe. Aus der Perspektive der Kunst wird sie erst interessant, wenn sie für gestalterische Zwecke eingesetzt wird, und dann hängt es von dem schöpferischen Menschen selbst ab, ob ästhetisch hochwertige Resultate geschaffen werden. In Anbetracht dessen, daß die Mittel der Computergraphik erst seit kurzer Zeit zur Verfügung stehen, kann man noch kein einheitliches Oeuvre mit dieser Technik erwarten. Vielmehr kann man auch heute noch jede mit ästhetischer Zielsetzung entstandene Computergraphik als Erprobung des Mediums und seiner Eignung als künstlerisches Ausdrucksmittel ansehen. Dieser besondere Zustand entzieht die Computerkunst dem Zugriff des katalogisierenden Historikers, der Einheitlichkeit, Geschlossenheit und so weiter vermißt. Andererseits liegt hier der einzigartige Fall einer Kunst »in statu nascendi« vor. Der außerordentlich interessante Initialzustand einer Kunst, der sich in allen klassischen Sparten der Beobachtung entzieht, kann hier in der Entstehung beobachtet wer-



den – eine außerordentliche Gelegenheit, die seltsamerweise bisher von den einschlägigen Wissenschaftlern kaum genutzt wurde.

Einer der interessantesten Entwicklungszüge ist jener, der von ersten spielerischen Experimenten zur Kommerzialisierung führt. Ein anderer ist die Herausbildung von Stilrichtungen und Bewertungskriterien – eine Phase, die längst noch nicht abgeschlossen ist. In diesem Beitrag soll ein weiterer aufschlußreicher Aspekt aufgegriffen werden: die Wechselwirkungen zwischen technischem Instrumentarium und künstlerischem Ausdruck.

In den fünfziger Jahren war der mechanische Plotter das einzige graphische Ausgabegerät. Mit ihm ließen sich Tusche, Federn und Stifte programmgesteuert über eben aufgelegtes oder über eine Walze gespanntes Papier führen. Dieses Verfahren beschränkte die Computergraphik auf die für die Anfangszeit typische Strichzeichnung. Das entsprach auch der ursprünglich gestellten Aufgabe, der Erstellung von Blockdiagrammen, Schaltplänen, Landkarten und dergleichen.

Aber nicht nur die Bedingungen der Hardware, sondern auch die Qualitäten der Software wirkten sich auf die künstlerischen Gestaltungsversuche aus. Entsprechend dem angestammten Anwendungsbereich des Computers waren die Programmiersprachen besonders zur Beschreibung mathematischer und logischer Zusammenhänge geeignet. Verständlicherweise stützten sich die ersten Benutzer auf die verfügbaren Routinen, und so ist es nicht weiter verwunderlich, daß vieles, was damals entstand, seine Herkunft aus dem Formenschatz der Technik und Wissenschaft nicht verleugnen konnte.

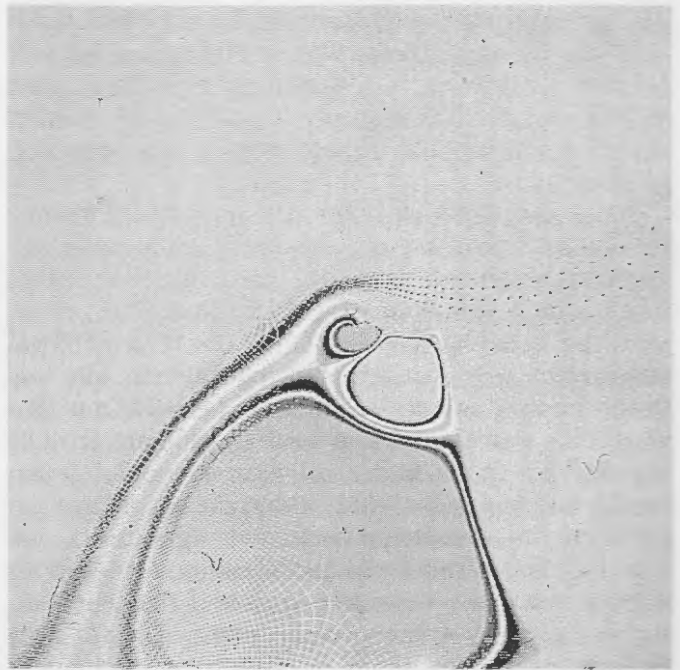
Diese Werke sind zumindest formal dem Konstruktivismus verwandt, insbesondere was die Präzision der Darstellung und die damals noch nötige Beschränkung auf einfache Formelemente betrifft. Während aber die Repräsentanten des klassischen Konstruktivismus mit Lineal und Zirkel auskommen mußten und dadurch auf Gerade und Kreis angewiesen waren, ist es für den Benutzer des Computers leicht, auch kompliziertere Kurven mit höchster Präzision einzusetzen. Das gelingt einerseits durch das Verfahren der Interpolation, andererseits durch den Einsatz mathematischer Formeln, die das Programm in graphische Darstellung umsetzt. Eine andere durch die Programmiersprachen gegebene formalstilistische Erweiterung betrifft den Übergang von der Ordnung zum Chaos. Mit Hilfe des sogenannten Random-Befehls lassen sich (scheinbar) ungeordnete Zahlenreihen ausgeben, die man als Kenngrößen graphischer Darstellungen verwenden kann. Die Einbeziehung des Zufalleffekts, die typisch war für die erste Periode der Computergraphik, hat sogar einen Niederschlag in manuell gefertigten konstruktivistischen Werken gefunden, beispielsweise in jenen von Herman de Vries. Einige Konstruktivisten, beispielsweise Peter Struycken, Zdeněk Šykora und Gerhard von Graevenitz, setzten den Computer ein, um ihre Bildideen zu verwirklichen.

Ganz andere Zielsetzungen hat die Methode des picture processing, die graphische Datenverarbeitung. Ursprünglich hatte sie dieselbe Aufgabe wie die mit fotografischen Mitteln arbeitende Bildverbesserung, doch ließ sich der Anwendungsbereich mit digitaler Elektronik beträchtlich erweitern – beispielsweise zur Entzerrung von Bildern oder zur Eliminierung von Störungen. Im Gegensatz zur Computergraphik arbeitet man beim picture processing mit Bildern von realen Objekten, die damit auch einer künstlerischen Verarbeitung offenstehen. Wieder bieten sich dazu die bereits verfügbaren Routinen an, die nun allerdings nicht zur Verbesserung der Bilder, sondern zu ihrer Verfremdung angewandt werden, wobei sich reizvolle graphische Effekte ergeben. Auf diese Weise vollzieht sich mit den Mitteln der Computergraphik der Übergang vom Realismus zur Abstraktion, was zu aufschlußreichen Vergleichen mit der entsprechenden kunsthistorischen Entwicklung herausfordert.

Die Anfänge des picture processing gehen noch in die Zeit der Drucker und Plotter zurück, doch der

Aufschwung ist an den Bildschirm gebunden. Der Einsatz der Bildröhre als Ausgabegerät der Computergraphik leitete eine entscheidende Wendung ein. Mit Color-Screen erweitert sich die Farbpalette über die Grenze der menschlichen Farbunterscheidung hinaus – auf Abstufungen von weitaus mehr als hundert Millionen Tönen. Andererseits vollzieht sich dabei der Übergang von der Strichzeichnung zum Tonbild, das auch den Verlauf zuläßt, insbesondere von Hell-dunkel-Werten. Im Gegensatz zu den Plotterdarstellungen, deren Aufbau oft über eine Stunde dauerte, wird das Bild nun im Bruchteil von Sekunden erzeugt. Das erlaubt den interaktiven Betrieb: Es gibt keine Wartezeiten mehr, der Benutzer kann das Resultat seiner graphischen Anwendungen unmittelbar sehen und seine Bilder so lange verändern, bis er das Ergebnis für optimal hält. Und schließlich wird damit auch die Erfassung von Bewegungsverläufen leicht. Bei größeren Systemen lassen sich Sequenzen von 25 Bildern pro Sekunde sogar in Echtzeit erzeugen. Zum ersten Mal hat der bildende Künstler ein Mittel in der Hand, das es ihm erlaubt, graphische Abläufe frei zu gestalten, ohne sich auf Vorbilder zu stützen.

Während die eingeschränkten Möglichkeiten des Plotters noch einen Trend zu mathematisch-konstruktivistischen Darstellungen ausübten, gibt das Monitorbild dem Künstler volle Freiheit: Computergraphik ist heute nicht mehr an einen bestimmten Stil gebunden, vielmehr hängt dieser uneingeschränkt von den Auffassungen des Künstlers ab. Und trotzdem ist er nicht völlig unabhängig vom Instrumentarium. Ihm bieten sich die aus Wissenschaft und Technik stammenden Methoden an – von denen er Gebrauch machen kann oder nicht. Es liegt bei ihm, ob er die sogenannten Paint-Systeme – mit denen sich die manuelle Arbeitsweise simulieren läßt



Serie »Komplex«. Unter dem Begriff Computer Scienc Art hat Herbert W. Franke gemeinsam mit Horst Helbig mathematische Methoden in Bilder umgesetzt.

– einsetzen will. Er erreicht dadurch eine vorher kaum vorstellbare Flexibilität: Er kann die Farben beliebig mischen und ändern, Teile des Bildes drehen, verschieben, vervielfältigen oder löschen, er kann Objekte herauszoomen und im vergrößerten Maßstab Details ausarbeiten und dieses Teil wieder ins Bild einfügen. Die Reihe dieser technischen Vorteile ließe sich beliebig fortsetzen. Die auf diese Weise entstandenen Bilder unterscheiden sich formal nicht von denen, die mit konventionellen Methoden zustande kommen.

Für manche Künstler sind dagegen gerade die in der klassischen Malerei nicht bekannten, erst in der Computergraphik gegebenen Erweiterungen in Stil und Ausdruck interessant. So ist beispielsweise der Einsatz mathematischer Formeln keineswegs überholt, sondern läßt sich mit den Mitteln, die seit der Pionierzeit der Computergraphik entwickelt wurden, ganz gehörig erweitern. Dabei ergibt sich ein wesentlicher Unterschied: Bei herkömmlicher Arbeitsweise arbeitet man punktuell – das heißt, daß in einem Bild genau die Stelle verändert wird, in die man gerade eingreift. Die Computergraphik erlaubt aber auch den integralen Eingriff, mit dem das Bild in seiner Gesamtheit verändert wird. Dazu gehören insbesondere die sogenannten Transformationen, mit deren Hilfe sich Bilder nicht nur verzerren, sondern in vielfacher Weise verändern lassen. In einfachsten Fällen kann man Farben austauschen, das Bild rastern oder Konturen herausheben. Bei komplizierteren Transformationen können auch völlig neue Bildstrukturen entstehen, denen man die Abstammung von den ursprünglichen Elementen nicht mehr ansieht. Durch mehrfache Transformationen kann man ein Bild ebenso gestalten wie durch die Verteilung von Form- und Farbelementen, ja, ein mathematisches Gesetz besagt sogar, daß auch auf diese Art jedes beliebige Bild entstehen kann. Beide Methoden sind also komplementär. Andererseits aber ist jede Methode jeweils anderen Aufgaben angemessen, und so bedeutet die Einbeziehung der »integralen Methode« auch von der Bildstruktur her eine beachtenswerte Innovation.

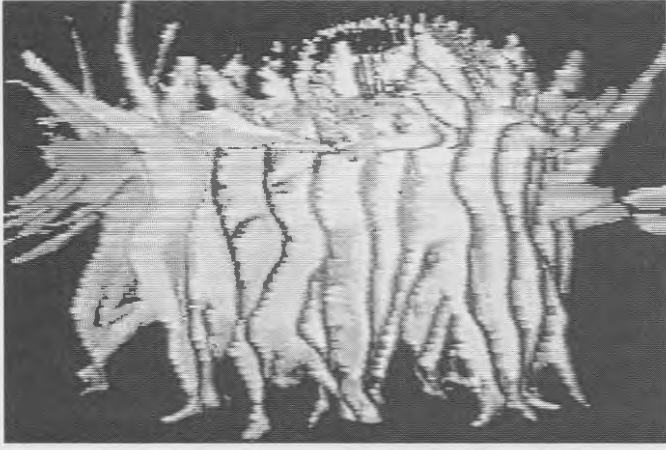
Noch deutlicher wird die Anlehnung der Entwicklung beim Übergang zur dreidimensional-perspektivischen Darstellung. Waren es zuerst die Strichzeichnungen von Plänen und Karten, so sind es seit neuestem die Darstellungen des CAD, des »computerunterstützten Designs«, die als Vorbild dienen. Die Methode stammt aus der Technik und betrifft den Entwurf von Maschinenteilen und Gebäuden. Anstelle der Modelle treten Bilder, die man von allen Seiten her betrachten kann. Vom Schaltpult aus kann man den Sichtwinkel ändern. Voraussetzung dazu ist, daß man das Objekt nicht nur als Projektion, sondern als dreidimensionales Gebilde gespeichert hat. Zur Software gehörige Routinen sorgen dafür, daß auf dem Bildschirm die gewünschten Ansichten erscheinen. Mit Hilfe von speziellen Programmen läßt sich die Wirklichkeitsnähe der Darstellungen noch steigern. Zunächst ging es um das Problem, hinten liegende

Teile durch die vorderen zu verdecken, später verlagerte sich die Entwicklung der Software auf Schattenschwurf und Helligkeit entsprechend der geometrischen Optik, wobei sich die Position der fiktiven Lichtquellen beliebig festlegen läßt. In der letzten Zeit richtete sich die Aufmerksamkeit der Systemanalytiker auf Fragen, die über die Modellierung von Maschinenteilen und Gebäuden weit hinausgehen, beispielsweise die realistisch scheinende Darstellung von Gebirgen, Wolken, Wasser, Lebewesen... Die Anwendungen liegen nicht mehr in der Technik, sondern im Bereich von Film und Video. Manche der Effekte sind so verblüffend, daß man sie für Kunstwerke hält. Bei den Ausstellungen der SIGGRAPH-Konferenz, der jährlich veranstalteten größten Computergraphiktagung, findet man neben künstlerischen Arbeiten immer wieder auch solche, die zweifellos Ergebnis kreativer Programmgestaltung, aber keineswegs Kunst sind.

Und dennoch wäre es verfehlt, dieser Methode deshalb künstlerische Potenz abzuspochen. Die Programmentwicklung erweist sich als unverzichtbare Basis, ohne die die künstlerische Leistung in diesem Bereich nicht möglich wäre. Es ist der Bereich des Fotorealismus, jenes in Kunstkreisen vor einiger Zeit aktuellen Stils, bei dem es darum ging, Szenen so realistisch wie eine Fotografie wiederzugeben.

Obwohl sich die Ergebnisse dieser Richtung nicht von gemalten Arbeiten unterscheiden, gibt es doch einen wesentlichen Unterschied. Mit dem Einsatz von 3D-Routinen bleibt man nämlich nicht an der Oberfläche der Dinge, sondern baut sie als räumliche Objekte auf – eine ganz andere Näherung als die Wiedergabe perspektivischer Projektionen. Daß es sich um eine echte Erweiterung in der Darstellung handelt, wird daraus ersichtlich, daß man auf diese Weise wiedergegebene Objekte von allen Seiten her betrachten kann – wenn es gewünscht wird, auch in einer simulierten Kamerafahrt. Hat man es mit beweglichen Dingen zu tun, beispielsweise mit einem Tier, dann geht die Auseinandersetzung des Künstlers mit seinem Objekt noch weiter. Er kann beispielsweise das Zusammenspiel von Skelett und Muskeln betrachten, die Freiheitsgrade der Bewegung studieren und sie schließlich wie in einem Film in Bewegung setzen. Auch hier genügt der Effekt allein noch nicht, um eine Darstellung zum Kunstwerk zu machen, doch allein die Verfügbarkeit der Methode bedeutet eine ungeheure Herausforderung an den Künstler, der mit ihrer Hilfe zu Ausdrucksmöglichkeiten findet, die ihm bisher verschlossen waren.

Die Erfahrungen mit den ersten Bildern oder Bildsequenzen, die auf diese Weise entstanden, zeigen, daß der Realismus relativ uninteressant ist. Es geht hier ja nicht um den Abklatsch der Wirklichkeit. Doch genau wie beim *picture processing*, das man nicht nur zur Verbesserung von Bildern, sondern auch für ihre Verfremdung anwenden kann, eröffnet sich für den Künstler eine ganz neue Dimension, wenn er vom Realismus zum Surrealismus übergeht. Zum ersten Mal ist es ihm möglich, in seiner Phantasie entstandene Szenarien dreidimensional aufzu-



Digitales Ballett. Hommage à E.M.

bauen, die Vision von Welten zu geben, die in Wirklichkeit nicht existieren, vielleicht auch gar nicht existieren können.

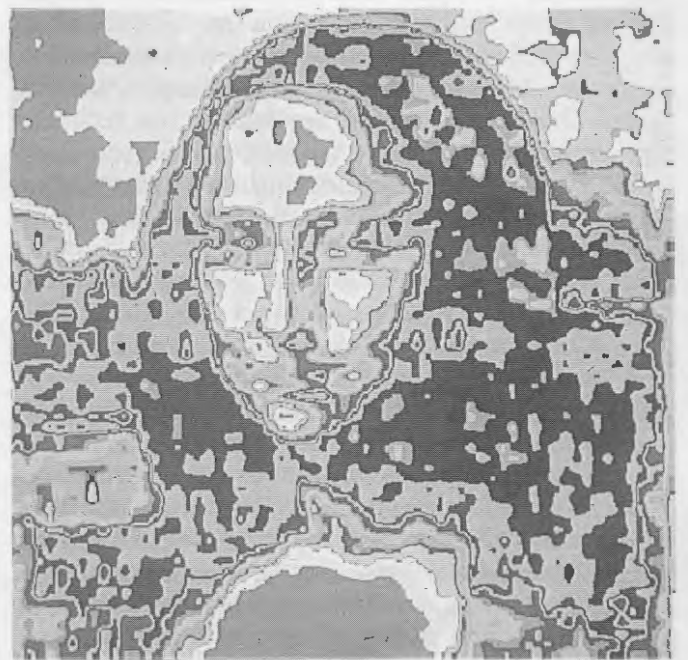
So ungewöhnlich Arbeiten dieser Art auch erscheinen mögen, so beruhen sie doch noch auf einem klassischen Kunstverständnis, das den Künstler als aktives und kreatives Element sieht, dem Publikum dagegen lediglich eine passive, betrachtende Rolle zuweist. Der Computer erlaubt aber auch die Interaktion, das Zusammenspiel zwischen dem Benutzer und dem Programm, wie man es für manche technisch-wissenschaftliche Zwecke einsetzt. Überträgt man dieses Prinzip in die Kunst, so ergeben sich weitere phantastische Möglichkeiten der Gestaltung und des Ausdrucks.

Erst vor kurzem wurde in die Computergraphik der Begriff der »virtuellen Wirklichkeit« (virtual reality) eingeführt. Dazu benötigt man heute noch recht komplizierte Apparaturen, beispielsweise einen Helm, der perspektivisch und stereoskopisch wiedergegebene Bilder vermittelt. Er gestattet es dem Benutzer, in die von einem Programmierer geschaffene Welt einzudringen und sich in ihr in beliebige Richtungen zu bewegen. Manche dieser Systeme bieten sogar die Möglichkeit des Eingriffs; mit einem »Datenhandschuh« kann man Gegenstände spüren, ergreifen und sie an eine andere Stelle versetzen.

Wie auch in anderen Bereichen der Computertechnik sind es keineswegs nur ästhetische Gründe, die zur Entwicklung solcher Produkte geführt haben. Die Technik der »künstlichen Wirklichkeit« läßt sich für didaktische Zwecke verwenden, um beispielsweise die Bewegung in schwierigem Gelände zu trainieren, medizinische Operationen zu üben oder unzugängliche Räume aus eigener Anschauung kennenzulernen. Heute steht die dazugehörige Hard- und Software erst in der Entwicklung, doch das Interesse daran ist groß. Vermutlich wird es neben der Wissenschaft und der Technik zuerst die Unterhaltungsindustrie sein, die davon Gebrauch macht, doch werden auch bereits rein künstlerische Anwendungen diskutiert. Wie es scheint, wird der Künstler der Zukunft sich nicht damit begnügen, Bilder seiner Phantasiewelten zu erstellen, sondern er kann sie dem Betrachter unmit-

telbar zugänglich machen, also eigene Erfahrung, eigenes Erlebnis vermitteln.

Zeitkritiker weisen darauf hin, daß sich der Mensch in den künstlichen Räumen verirren oder das Interesse an der Realität verlieren könnte. Sicher wird es nötig sein, Entwicklungen dieser Art kritisch zu verfolgen, doch die faszinierenden Möglichkeiten, die sie eröffnen, bleiben unbestritten. Man kann voraussehen, daß die neuen Medien tiefgreifenden Einfluß auf das Verhalten und Denken der Menschen ausüben werden, und mancher mag der Meinung sein, daß die Veränderungen im Raum der Kunst dagegen unwichtig sind. Allem Anschein nach dürften sich die Veränderungen im kulturellen Raum durch einen neuartigen und immens erweiterten Gebrauch von Bildern äußern. Manche Philosophen wie etwa Vilém Flusser sind der Meinung, daß die lange Epoche der Sprachkultur am Ende ist und einer neuen Bildkultur Platz macht. Wenn sich aber die Kommunikation tatsächlich von der Sprache auf das Bild verlagert, dann gewinnen die damit verbundenen ästhetischen Kriterien gegenüber den bisher im Vordergrund stehenden literarischen an Bedeutung. Es wäre also wünschenswert, daß sich auch Künstler daran beteiligen, wenn es darum geht, unsere künftigen Kommunikationsmittel zu entwickeln.



Künstliche Wunderwelten, natürliche Verständigungs- probleme

Diskussion mit
Marvin Minsky und
Joseph Weizenbaum
über »Virtual Reality«

Einer der Festival-Gurus wurde bis zur letzten Minute mit Spannung erwartet – und kam doch nicht: Jaron Lanier, der visionäre Westküsten-Produzent der »Virtual Reality«-Technik, bei der Auge, Ohr und Hand über Datenhelm und -handschuh mit dem Computer gekoppelt werden, so daß der Anwender als »Maus« eine dreidimensionale Computergraphik in Bewegung setzen kann. Das einzig Präsentierbare war ein Video, das Laniers Firma VPL den Festival-Organisatoren zugeschickt hatte. Doch Laniers Abwesenheit schuf einen »virtuellen Raum«, in dem sich Erstaunliches ereignete: Marvin Minsky und Joseph Weizenbaum, alte Kontrahenten am Massachusetts Institute of Technology (MIT), setzten sich auf die Bühne der Gasteig-Philharmonie und diskutierten gemeinsam mit Festivalbesuchern über »Virtual Reality« (VR).

Prof. Minsky, Sie haben doch mal den VR-Helm benutzt. Vielleicht können Sie uns erzählen, was das für ein Gefühl ist.

Marvin Minsky: Ich hatte mal Scott Fischers Helm an und den von Jaron Lanier. Sie haben ja in dem VPL-Video gesehen, daß die VR-Bilder ziemlich holprig sind. Dazu will ich erstmal eine Geschichte erzählen. Anfang der sechziger Jahre, als wir am Massachusetts Institute of Technology anfangen, künstliche Intelligenz zu erforschen, und damit auch Roboter bauten, sprach ich ab und zu mit einem Freund, der Detektiv-Roboter-Geschichten schreibt: Isaac Asimov. Jedesmal wenn ich ihn traf, sagte ich ihm, er soll doch mal ins MIT kommen und sich unsere Roboter anschauen. Er hat immer abgelehnt und irgendwelche Ausreden dafür gefunden. Erst vor einem Jahr, als ich ihn in New York wiedertraf und er sich inzwischen längst mal die MIT-Roboter angeschaut hatte, hat er mir verraten, warum er nie kommen wollte: »Ich wollte deine Roboter nie sehen, weil ich mir schon denken konnte, daß sie plump und ziemlich dumm sein würden.« Er arbeitete damals in Boston an einem Science-Fiction-Roman, in dem ein Roboter namens Roboter Daniel Oliver eine Rolle spielt. Jetzt erklärte er mir: »Ich wollte mir meine Vorstellungskraft nicht

verderben lassen. Ich dachte, ich könnte Probleme bekommen beim Schreiben.«

Er hatte recht: Wenn Sie sich für Virtual Reality interessieren, sollten Sie es im Moment lieber vermeiden, sie anzuschauen.

Bei Virtual Reality haben Sie irgendein Objekt, das räumlich vorhanden zu sein scheint. Scott Fischer hatte ein Flugzeug bei der Nasa, er arbeitete bei JPL (Jet Propulsion Laboratory). Man konnte daruntersteigen und sich die Unterseite ansehen, man konnte die Hand herausstrecken. Wenn man sich bewegte, veränderte sich die Sicht auf die Objekte. Es ist, als ob man an einem anderen Ort wäre. Sie gewöhnen sich schnell daran, daß Sie sich an einem Ort mit schlechter Sicht und geringer Auflösung befinden. Wenn Sie vorwärts gehen, weichen Sie aus, um nicht gegen das Flugzeug zu stoßen – auch wenn es überhaupt nicht da ist. Es ist also eine einigermaßen gute Illusion, und sie wird immer vollständiger werden in den nächsten Jahren, wenn die Bildgeneratoren besser werden.

Die Erfahrungen mit dem Datenhandschuh sind auch recht überzeugend. Sie können in den Monitor hineinlängen und Objekte herumschieben. Das geht einigermaßen. Es gibt noch das Problem, daß man die Objekte nicht fühlen kann, weil es noch keinen Datenhandschuh mit gutem taktilem Feedback gibt, der den Tastsinn wiedergeben würde. Das wäre die nächste wichtige Stufe. Wenn wir dann eine Art Kraft-Wiedergabe haben, könnte Virtual Reality ein sehr gemütlicher Ort werden. Er braucht noch etwas technische Entwicklung, aber es ist ganz klar: Die Rechenleistung wird jedes Jahr billiger, alle fünf Jahre bekommt man ungefähr zehnmal mehr Leistung, also müßten die VR-Maschinen in den nächsten fünf oder zehn Jahren recht gut werden. Für was auch immer Sie sie benutzen wollen – es müßte machbar sein.

In Deutschland ist Cyberspace etwas ganz Neues. Wie ist das in den USA?

Joseph Weizenbaum: Das hängt ganz davon ab, mit welchem Insiderkreis Sie reden. Für manche Leute begann es Anfang der sechziger Jahre, wie Marvin gesagt hat. Und das Militär hat diese Sachen schon viel früher für Piloten entwickelt. Aber für viele ist es auch in den USA etwas Neues.

Ich will jetzt erstmal etwas zum Begriff »Virtual Reality« sagen. Wenn man auf englisch sagt, daß eine Sache »virtually« etwas anderes sei, dann meint man, daß es einige »virtues« (Eigenschaften) einer anderen Sache hat, obwohl es dieses andere nicht ist. Wenn jemand zum Beispiel sagt, »Es war virtuell Nacht«, dann meint er: »Was auch immer es war, es war nicht Nacht.« Was man damit meint: »Für meine Zwecke war es in diesem Moment so, als ob es Nacht gewesen wäre.« Man sollte also festhalten, daß dieses ganze Konzept mit Absichten zu tun hat. Der Begriff »Virtuelle Realität« ist ein Widerspruch in sich.

Woody Allen, der hier als ein großer amerikanischer Philosoph bekannt sein dürfte, sagte mal, daß er Realität nicht so wahnsinnig mag, aber es sei eben



der einzige Ort, wo man ein gutes Steak bekommt. Bis Virtual Reality die Stufe erreicht, in der man nicht nur einen Datenhandschuh, sondern auch eine Daten-Zahnprothese benutzen kann, wird Woody Allen wohl bei der Realität bleiben, die er am besten kennt.

Kann man verhungern in Cyberspace oder sich darin verirren?

Minsky: Vor vielen Jahren hatte ich einen jungen Freund, er war in der dritten Klasse, also etwa neun Jahre alt. Es war 1964, und wir hatten einen der ersten Computerterminals. Das waren noch richtige Fernschreiber-Apparate. Die Studenten hatten ein Spiel programmiert, das »Adventure« hieß (Abenteuer) und sehr berühmt wurde. Ein typisches intellektuelles Computerspiel, bei dem die Maschine sagt: »Du bist in einem Zimmer, und es gibt eine Tür nach Norden und ein Fenster nach Osten und ein Treppenhaus, das in westliche Richtung nach oben führt.« Dann gibt man »nord«, »ost« oder »west« ein, und die Maschine gibt zurück: »Jetzt bist du in einem anderen Raum. Links sind ein Drachen und ein Schwert.« Diese Spiele sind ja meistens von Fabeln abgeleitet. Und wie Sie wissen, werden den Kindern seit Menschengedenken gleich als erstes diese Geschichten beigebracht über Kriege und Drachen und Waffen und Monster, die dich umbringen, wenn du sie nicht vorher tötet. Dieser Junge, Michael, war nicht sehr gut in der Schule, weil er sich weigerte,

lesen zu lernen. Aber er wollte »Adventure« kennenlernen, also lernte er die Worte, die von der Maschine geschrieben wurden, und er lernte, wie er Begriffe eingeben kann. Nach einer Weile wollte Michael in einen bestimmten Raum in »Adventure« zurück, wo es ihm gefiel, vielleicht wollte er auch raus, aber er verirrt sich, und er blieb den ganzen Tag lang im Spiel hängen. Er wäre fast verhungert, weil er sich verirrt hat in Cyberspace. Wir brachten ihm das Essen an den Terminal. Sein Vater rief an und fragte, wo Michael geblieben sei, und ich sagte, »Oh, er hat sich in Cyberspace verirrt«. Und sein Vater sagte: »Oh, gut, er hat noch nie irgendetwas Intellektuelles gemacht.« Nach einer Weile sagte Michael: »Hast du einen Bleistift und Papier?« Ich sagte »klar« und war neugierig, was er damit machen würde, dieser Junge, von dem seine Lehrer sagen, er sei ein Legastheniker. Er begann, eine Karte zu zeichnen, den Thron-Raum und den Drachen-Raum und so weiter, und er zeichnete die Verbindungen ein. Für ihn war es eine ganz große Aufgabe, es dauerte länger als einen Tag, er vergaß das Essen völlig, und es war das erste Mal, daß er eine Verwendung fand für Worte, Bleistifte und Papier. Er dachte: Wenn ich eine Karte zeichnen kann, kann ich die Struktur verstehen.

Dabei war das eine sehr schlechte Virtual Reality, 1964 gab es keine Bilder oder Diagramme, nur Texte, die in den Computer eingegeben werden. Und doch war es interessant, aufregend und ein bißchen beängstigend zu beobachten, wie schnell dieses Kind, das sich ja nicht wirklich verirrt hatte, die reale Umgebung vergaß. Er hätte ja jederzeit, wie Jo Weizenbaum gesagt hat, in die Küche gehen und ein echtes Steak essen können, aber er hatte vielleicht Angst, er könnte dann die Karte vergessen. Ich erinnere mich an diese Situation, weil ich Optimist bin. Weizenbaum ist da meiner Meinung nach pessimistisch. Ich dachte: Es ist ein großartiges Werkzeug, es bringt Kinder dazu, Intellektuelle zu werden.

Viele Jahre später konnte ich sehen, wie 40 Millionen Kinder Nintendo spielen und Atari-Games, und ich weiß immer noch nicht, ob das gut oder schlecht ist. Ich weiß nur, daß sie es mögen. Wenn man sie fragt, wie ihnen die Schule gefällt, dann sagen sie: »Ganz furchtbar. Wir lernen nie irgendetwas, und hier lerne ich, wie man ein Monster erlegt.« Es ist beängstigend und interessant.

Weizenbaum: Ich will noch etwas zum Thema Optimismus und Pessimismus sagen. Es gibt da einen Kalauer: Der Optimist sagt: »Dies ist die beste aller möglichen Welten.« Und der Pessimist antwortet: »Stimmt.« Viele meiner Kollegen, insbesondere Marvin, haben großes Vertrauen, daß bestimmte Entwicklungen machbar sind, und ich stimme darin mit ihnen überein, aber für mich ist das eine pessimistische Feststellung, keine optimistische. Das war lange Zeit sozusagen mein Schicksal.

Können Sie sich vorstellen, daß Virtual Reality uns allen eine neue Art des Denkens eröffnet, wenn wir durch abstrakte Produkte wie Software laufen können wie durch unsere Häuser?

Weizenbaum: Ich weiß es nicht. Ich hätte Angst davor, ausgehend von Geräten, technischen Spielereien und Ideen, die momentan noch in einer sehr primitiven Form existieren, auf solche monumentalen Sachen zu schließen wie »neue Denkformen«. Und angenommen, sie würden uns tatsächlich die Möglichkeit eröffnen, auf eine neue Art zu denken: Sind das die neuen Methoden, die wir jetzt brauchen? Aber mich stört vor allem diese Extrapolation. Heute ist es sicherlich viel zu früh, um das zu beurteilen. Und was diese Analogie angeht, wir könnten dann durch Programme laufen wie durch Häuser: Ich weiß nicht, was das bedeuten soll, und ich bin mir nicht sicher, ob derjenige, der die Frage gestellt hat, das weiß. Im Computerbereich ist es ziemlich wichtig, sich klar und präzise auszudrücken. Es hilft jetzt nicht viel, sich einzureden, daß einige primitive Geräte, die wir heute haben, irgendwann irgendwelche großen Probleme lösen könnten.

Minsky: Ich weiß auch nichts über »neues Denken«. Es gibt viele Ideen, die interessant wären auszuprobieren. Jeder Mathematiker denkt irgendwann mal: Wäre es nicht nett, wenn ich in einer vierdimensionalen Welt sein könnte? Wenn ich nur meine Finger bewegen müßte, um mich in den Koordinaten X, Y und Z zu bewegen? Ich könnte vielleicht noch mit meinem vierten Finger eine zusätzliche Dimension verbinden. Dann könnte man ein vierdimensionales Objekt von allen Seiten betrachten.

Keiner weiß, ob es einem dabei hilft, vier Dimensionen zu denken, wenn man so eine Einrichtung zur Manipulation von vier Dimensionen hat. Mir erscheint es als unwahrscheinlich, daß das Gehirn darin sehr gut sein sollte. Meiner Meinung nach ist das Gehirn nicht einmal dafür geeignet, in drei Dimensionen zu denken. Die meisten Leute sind eher ungeschickt darin. Und die Leute, die drei Dimensionen zu verstehen scheinen, lassen sich immer noch von einem einfachen chinesischen Puzzle verwirren. Die räumliche Vorstellungskraft des Menschen ist meistens zweidimensional, aber wir kommen ganz gut zurecht mit unseren rechnerischen Methoden, verschiedene Perspektiven zu verbinden. Ich wäre neugierig auf ein Werkzeug, das mir helfen könnte, vierdimensional zu denken, aber der große Vorteil von Virtual Reality von der Art, wie Lanier sie baut, besteht eher darin, daß sie ganz normalen Leuten dabei helfen könnte, Dinge dreidimensional zu visualisieren.

Eine andere Sache, die keine neue Art zu denken ist, aber eine neue Art des Handelns: Als ich ein Kind war, habe ich Sachen aus Bauklötzchen gebaut. Sie nehmen Bauklötze aus der Kiste heraus und bauen einen Turm, der ungefähr zehn Einheiten hoch ist, und es wird etwas wackelig. Wenn Sie sehr gut sind, dann können Sie ihn vielleicht dreißig Klötze hoch bauen, dann beginnt er zu schwanken. Ich habe nie etwas aus hundert Holzklötzchen bauen können, es fiel dann immer um.

Einer der Vorzüge von Virtual Reality ist, daß man damit einige Eigenschaften der Realität haben und



andere weglassen kann. Im Computer könnten die Klötze perfekt passen, und Sie könnten, wenn Sie wollen, einen Turm bauen, der eine Million Bauklötzchen hoch reicht. Sie könnten mit solchen spektakulären intellektuellen Sportarten vielleicht etwas lernen, was Sie im Klassenzimmer nicht tun konnten, weil Sie weder die Kraft noch das Geld oder die Zeit dazu hatten.

Weizenbaum: Ein anderes Beispiel dafür sind die Darstellungen, die wir heute von großen Molekülgruppen haben, die man drehen und wenden und von allen Seiten betrachten kann. Ich glaube, es war James D. Watson, der meinte, wenn er und sein Kollege Crick damals dieses Hilfsmittel gehabt hätten, um molekulare Modelle zu bauen, dann hätten sie die Doppelhelix viel früher entdeckt. Das Problem war, daß ihre mechanischen Modelle zu schwer wurden und umfielen.

Zum Thema »neues Denken«: Wir könnten fragen, ob es dafür ein Beispiel aus der Vergangenheit gibt. Tatsächlich bietet die Mathematik, besonders moderne Mathematik, neue Formen des Denkens an. Man denke nur an die Erfindung der analytischen Geometrie. Wir können jetzt den Raum auf eine andere Art visualisieren als früher. Und es gibt neue Notenschriften, durch die man Musik anders wahrnehmen kann als bisher. Genauso die Notation in der Chemie oder choreographische Aufzeichnungen.



Alle diese Sachen machen es möglich, anders zu denken über Dinge, die sehr schwer verständlich sind. Natürlich stellt sich heraus, daß diese Denkformen für die meisten Leute nicht verfügbar sind, oder sie machen sie sich nicht zunutze. Es werden also neue Methoden des Denkens angeboten, aber nur für Leute, die gelernt haben, in der betreffenden Disziplin zu denken. Ich glaube, daß es heute genauso schwierig ist wie in der Vergangenheit, in anderen Darstellungsformen denken zu lernen.

Kann es durch Cyberspace eine Veränderung in der menschlichen Kommunikation geben?

Minsky: Man kann eine Welt mit verzerrten Entfernungen machen. Ich kann Ihnen nichts erzählen über die Grenzen menschlicher Kommunikation, aber ich weiß, daß man Dinge manchmal besser verstehen kann, wenn sie in Form von wunderschönen Diagrammen dargestellt werden. Manche Leute können besser über Telepräsenz kommunizieren, wenn sie Bilder von Gesten austauschen können, manche Leute kommunizieren besser über Stimmen.

In der VPL-Werbung, die wir gerade gesehen haben, haben Sie bemerkt, daß die erste Aussage war: »Wir haben die erste virtuelle Realität gemacht, in die zwei Leute eintreten können.« Das ist süß. Zwei Leute können in Laniers Maschine hineingehen mit zwei verschiedenen Helmen und können darin Ping-Pong spielen. Sie können einen Tischtennisschläger

mit ihrem Datenhandschuh halten und ihren Arm bewegen. Es ist perfekt, außer daß der Bildaufbau der Risc-Maschine zu langsam ist für Ping-Pong, aber das müßte nächstes Jahr auch noch verfügbar sein. Ich nenne es »Telepräsenz«, was dasselbe bezeichnet wie »Virtual Reality« und bedeutet, daß zwei Leute, die in verschiedenen Städten sind, miteinander Ping-Pong spielen können. Ich weiß gar nicht, warum Sie lachen. Ich stelle nur fest, daß Menschen ungefähr zwanzig Prozent ihrer Zeit damit verbringen, doofe Sportveranstaltungen anzuschauen. Mit Virtual Reality könnten die Leute zum Beispiel mit simulierten Tennis-Profis spielen – sogar die Leute, die nicht das Geld haben, das ein menschlicher Profi verlangen würde, bevor er mit ihnen Tennis spielt, weil sie so schlecht spielen, daß es keiner aushalten würde.

Das Tolle an einfacher künstlicher Intelligenz ist, daß sie zu dumm ist, um sich zu langweilen, auch wenn sie mit Leuten redet, die nur geringfügig intelligenter sind. Entschuldigen Sie, daß ich das Wort »Intelligenz« benutze, ich bin nicht der Meinung, daß es wirklich etwas bedeutet. Aus meiner Sicht sind alle Leute, die reden können und eine natürliche Sprache vernünftig verwenden können, ungefähr gleich hinsichtlich ihrer Gehirn-Leistungen. Die gescheitesten Leute denken nur ein bißchen anders. Es gibt da keinen großen Unterschied. Wir sind alle extravagante Schimpansen. Zwei beliebige Individuen könnten – wenn sie nicht schwer geschädigt sind – sehr ähnliche Dinge erreichen, wenn sie dasselbe Schicksal hätten.

Früher haben Philosophen die Grenzen des Wissens erprobt. Sind die AI-Leute die Philosophen von heute?

Weizenbaum: Einige wenden sich der AI zu, ich denke da zum Beispiel an Daniel C. Dennett, ein wichtiger junger amerikanischer Philosoph.

Minsky: Ich denke, daß die computerwissenschaftlichen Werkzeuge für die Beschreibung von Prozessen besser geeignet sind als die traditionellen philosophischen Methoden. Ich habe ja schon in meinem Vortrag die zynische Sichtweise geäußert, daß Philosophen die Grenzen des Wissens mehr demonstrieren als erproben. Die Leute, die über das Wissen und das Denken denken, sind heute meistens bei den neuen kognitiven Wissenschaften zu finden, die mit Computerforschung zu tun haben. Einige von ihnen wird man eines Tages vielleicht als Philosophen bezeichnen. Aber der Begriff Philosoph ist heute eher eine akademische Bezeichnung. Wenn Sie in den USA in ein Philosophisches Institut gehen, werden Sie meistens Leute vorfinden, die damit beschäftigt sind, Thesen mittels mathematischer Logik zu beweisen; Leute, die nicht die Grenzen von überhaupt irgend etwas auf die Probe stellen.

Was ist mit menschlicher Sprache, Verständnis von natürlicher Sprache, Übersetzung?

Minsky: Sobald die Maschine Worte verstehen kann, wird es für sie einfach sein, sie zu übersetzen. Vorher aber ist sowas Zeitverschwendung.

Wie weit ist die KI bis jetzt auf diesem Gebiet gekommen?

Minsky: Die AI-Leute wurden bisher in einem solchen Ausmaß von Linguisten dominiert, die sich nur für Grammatik interessieren, daß wir nicht vorankamen. Wir brauchen Projekte, die sich mit Repräsentationen von Alltagswissen beschäftigen und das mit Worten verbinden. Es macht keinen Sinn, die Sprache als eine abgesonderte Wissenschaft zu erforschen.

Weizenbaum: Viel hängt davon ab, was man meint mit »Verstehen von natürlichen Sprachen«. Meine Sicht davon ist: Jedes Individuum hat eine einzigartige Lebensgeschichte und versteht Sprache auf eine einzigartige Weise. Es ist für mich nicht klar, was es heißen soll, daß der Computer ein Fragment der Sprache verstehen soll. Lassen Sie mich das mit einem Beispiel veranschaulichen. Ich werde eine kurze Geschichte erzählen, die mir mal passiert ist, obwohl die meisten Leute in diesem Raum sie wahrscheinlich nicht verstehen werden. Das hat nichts damit zu tun, ob ich sie auf deutsch oder auf englisch erzähle, das ist nicht der Punkt.

Die Geschichte: Ich war an einem hektischen Nachmittag in New York und wollte die 5th Avenue überqueren. Ich stand in einer Gruppe mit vielen anderen Leuten vor der Fußgängerampel und wartete auf grün. Da stieß mich der Mann neben mir sanft mit dem Ellbogen an, ich schaute ihn an, und er fragte mich leise: »Sind Sie jüdisch?« Ich sagte »ja«, und dann fragte er: »Wieviel Uhr ist es?«

Okay, diese Geschichte hat mich tief bewegt. Viele Leute werden sie überhaupt nicht verstehen. Was würde es jetzt bedeuten, wenn man sagen würde, daß ein Computer diese Geschichte verstehen kann? Was würde dann »Verständnis« bedeuten?

Minsky: Das hängt davon ab, ob der Computer jüdisch ist.

Weizenbaum: Das war ein netter Kommentar, aber was würde das für den Computer bedeuten?

Minsky: Oh, alles mögliche.

Weizenbaum: Was bedeutet es für ein Individuum, jüdisch zu sein? Das ist gar nicht so eindeutig. Worauf ich hinaus will: Es geht einem leicht über die Lippen zu sagen: Computer können heute Sprache bis zu einem bestimmten Grad verstehen. Ich glaube nicht, daß es zwei Individuen möglich ist – auch nicht Zwillingen, die im selben Haus aufgewachsen sind –, sich gegenseitig absolut zu verstehen. Absolutes Verstehen ist nicht möglich. Trotzdem verstehen wir uns. Wie ist das möglich? Das geht, weil wir im allgemeinen in einem relativ engen Kontext miteinander reden. Und innerhalb dieses Zusammenhangs können wir viele Voraussetzungen machen. Das ist die eine Seite.

Minsky: Wir nehmen beide teil an Millionen kleinen Wissensteilen, die sehr schwer einzufangen sind.

Weizenbaum: Auf der anderen Seite können wir signalisieren, daß wir uns nicht verstehen. Dann können wir uns weiterunterhalten. Bei einem Concierge eines teuren Hotels könnte man leicht glauben, daß er alle Sprachen der Welt versteht, denn wenn man ihm – egal in welcher Sprache – die Frage stellt, »Können Sie mir ein Ticket für die Oper heute abend besorgen«, wird er in der gleichen Sprache antworten.

In Wirklichkeit versteht er nur wenige Sprachen, und die anderen Sprachen versteht er nur innerhalb dieser sehr beschränkten Hotel-Welt der Hotel-Gäste. Wenn man die Sache nach diesem Gesichtspunkt betrachtet, dann ist es durchaus möglich, einen Computer dahin zu bringen, daß er innerhalb eines ganz bestimmten Kontextes »verstehet« – gar keine Frage.

Es geht doch darum, mehr über Sprache, Syntax und Semantik herauszubekommen, was nicht dasselbe ist wie Verstehen.

Minsky: Das Problem bei den Leuten, die sich mit formaler Sprache beschäftigen, mit linguistischer Theorie, ist zumindest in den USA, daß sie besessen sind von Syntax. Also haben sie vierzig Jahre lang wenig Gedanken auf die Frage verwandt, was ein Wort bedeutet. Was bedeutet das Wort Glas? Es schließt vieles ein: das Glas als Material, normalerweise transparent, auf Siliconbasis, das zu einer bestimmten Gestalt geformt wurde. Es kann Flüssigkeit aufnehmen. Man weiß auch, daß Dinge aus Glas leicht zerbrechen, wenn man sie fallen läßt. Und Sie können »fallen« nicht verstehen, außer Sie wissen etwas über Schwerkraft und andere Phänomene aus der erfahrbaren Physik.

In der Computerwissenschaft ist nun folgendes passiert: Leute meinten, es sei zu schwer, Bedeutungen zu verstehen und zu schwierig, die vielleicht Millionen von kleinen, miteinander verbundenen Erfahrungs- oder Repräsentationsfragmenten zu sammeln, auf die sich Jo Weizenbaum bezogen hat. Also haben sie einfach aufgegeben, und jetzt versucht ein Projekt nach dem anderen, dem Problem von Bedeutung oder annähernder Bedeutung auszuweichen und sucht nach etwas in der Struktur der Sprache selbst.

Ich finde das verrückt. Die formalen, sequentiellen und syntaktischen Aspekte der Sprache machen nur einen Bruchteil der Prozesse aus, die stattfinden, wenn einer etwas zu einem anderen sagt. Die meisten kurzen Sätze können Sie ganz ohne Grammatik verstehen, weil Sie diese Form von Satz schon einmal gehört haben und bestimmte Worte und ihre Reihenfolge kennen. Jedes Kind kennt mindestens zwanzigtausend vollständige kurze Sätze. Aber in Linguistik-Aufsätzen habe ich noch nie eine Auseinandersetzung mit Spracherkennungsprogrammen gefunden, die auf Basis von zwanzigtausend abgespeicherten Sätzen arbeiten.

Wenn Sie Fortschritte machen wollen im Bereich Spracherkennung, dann konzentrieren Sie sich am besten erstmal auf die privaten Kulturen, Alltagskulturen und Alltagserfahrungen der Leute, mit denen Sie reden. Und noch wichtiger: Sorgen Sie dafür, daß kein Linguist Ihr Labor betritt in den nächsten zehn Jahren, denn wir müssen uns erstmal von dem Schaden erholen, den sie in den letzten vierzig Jahren angerichtet haben. In Europa gibt es Leute, die Semiotik studieren – in den USA sind die ziemlich rar –, also dürften die Bedingungen hier besser sein. In den USA werden die Theoretiker dominiert von Leuten wie Noam Chomsky, die sagen: »Es bringt nichts, Bedeutung zu untersuchen, Sie müssen diese forma-

len Geschichten untersuchen.« Ich meine, daß das Gegenteil wahr ist. Roger Shank kam ganz gut aus ohne jede Grammatik, nicht sehr gut, aber besser als andere.

Wird Cyberspace von praktischer Bedeutung sein für Architekten?

Minsky: Viele professionelle Architekten benutzen jetzt schon diese Technik. Sie benutzen meistens nicht den Helm, sondern eine Maus oder einen Joystick, um die Grafik auf dem Monitor zu bewegen. Ich habe einen Freund in Buenos Aires, der die Technik schon seit zwanzig Jahren einsetzt, und ich wette, daß sie in Europa überall zu finden ist für Architektur.

Meinen Sie, daß dieses Medium Design und Kreativität anregen wird, oder wird dadurch alles immer uniformer?

Weizenbaum: Ich glaube, daß Kreativität abhängt von kreativen Leuten und nicht von Werkzeugen. Was Sie jetzt beobachten können in Städten wie München oder London: Leute laufen herum mit ziemlich extravaganten Videokameras und glauben, sie könnten damit gute Filme machen. Sie wissen offensichtlich nicht, daß dazu gute Ideen gehören. Und die fliegen einem nicht einfach zu. Es gibt Fotografen, die ganz einfache Kameras benutzen und damit sehr gute Ar-



beit leisten, und es gibt Millionen von Fotografen, die sehr teure Kameras benutzen und sehr schlechte Arbeit machen. Nicht das Instrument macht jemanden kreativ. Kreative Leute erfinden kreative Verwendungen für das Instrument.

Minsky: Ich bin da etwas anderer Meinung. Um kreativ zu sein, braucht man viele Talente. Man braucht eine Vorstellung von dem, was man machen will, man muß das Werkzeug beherrschen, man muß planen können, man braucht einen Sinn für die Komposition. Ich weiß das aus eigener schmerzhafter Erfahrung, denn ich bin sozusagen mit halbem Gehirn ein guter Musiker. Ich kann Harmonien machen, die sich wie Mozart oder Chopin oder Bach anhören, aber nach ein paar Takten stimmt etwas nicht, weil ich keinen Sinn dafür habe, wie man Geschichten erzählt oder einen Plan macht. Ich bin einfach kein Komponist. Ich kann also neun von zehn Dingen machen, die ein Komponist macht, aber ich brauche Hilfe mit dem Rest. Neulich habe ich Tod Machover um Hilfe gebeten, und wir werden sehen, ob wir et-

was zusammen machen können. Man kann sich in manchen Fällen beraten lassen von jemanden, der einige ungewöhnliche Talente hat, aber andere nicht. So könnte man etwas finden, was einem weiterhilft – wie ein künstliches Bein, eine Prothese. Aber der Computer wird da nicht viel tun können. Er wird einige Unzulänglichkeiten ausgleichen, aber er wird zumindest in den nächsten zwanzig Jahren weder die Vorstellungskraft noch den Plan oder die anderen Dinge zur Verfügung stellen, die Sie brauchen. Mein Freund John Cage zum Beispiel meinte, daß er eher schwach sei in Harmonie. Ihm gefielen die Rhythmen, die ich schrieb, und er sagte zu mir – vielleicht auch nur, um einen mangelhaften Musiker zu trösten: »Ich wünschte, ich wäre in Harmonie so gut wie du.« Und ich sagte: »Ich wünschte, ich könnte Stücke so gut planen, wie du es tust, aber nie zu tun scheinst.«

Haben nicht die Akzeptanzprobleme der künstlichen Intelligenz ihren Grund in einer Art Selbstrestriktion? Die meisten Leute schrauben ihre Erwartungen herunter, sobald sie mit Computern kommunizieren, wie wenn sie mit Babies sprechen, und erwarten einfach kein intelligentes Programm.

Minsky: Ich hoffe nicht.

Weizenbaum: Ich glaube, man könnte manchmal das Gegenteil sagen – das habe ich auch nachweisen können.

Minsky: Du hast die Probe gemacht.

Weizenbaum: Ja, ich habe die Erfahrung gemacht, daß ein ganz einfaches Programm, an das man eigentlich keine Erwartungen knüpfen sollte, die Leute dazu bringen kann zu denken, es sei extrem intelligent, so daß sie mit ihm in einer Art sprechen, die weit über die Fähigkeiten jedes Programms hinausgeht. Diese Fehleinschätzungen gehen also in beide Richtungen. Glücklicherweise, vielleicht auch unglücklicherweise, sind Leute sehr anpassungsfähig, und sie passen sich schlechten Sachen fast genauso leicht an wie guten.

Minsky: Wir haben einen gemeinsamen Freund, der ein Programm geschrieben hat, das einige Psychiater hereinlegte. Das Programm sollte einen paranoiden Menschen simulieren, also jemanden, der sehr mißtrauisch ist. Der Witz bei diesem Programm war: Wenn man etwas eintippte, dann antwortete es auf eine ziemlich befremdende, irrelevante Art und sagte Sachen wie: Ich traue dir nicht. Nach einer Weile sagte es: Ich rede einfach nicht mehr mit dir. Dann kam gar nichts mehr. Die Leute sagten dann immer: Ganz erstaunlich, es ist genau wie ein Paranoider. Ich habe mich ziemlich darüber geärgert, daß sie den Unterschied nicht verstehen wollten. Wenn ein Paranoider aufhört zu sprechen, dann steckt ein enorm komplexer Zusammenhang von Gründen dahinter.

Weizenbaum: Ich glaube, ich halte immer noch den Weltrekord auf das kürzeste Programm, das jemals veröffentlicht wurde: ein Programm, das infantilen Autismus simuliert. Sie geben irgendetwas ein, und das Programm antwortet überhaupt nichts. Darüber schrieb ich einen winzigkleinen Aufsatz, der tatsäch-



lich veröffentlicht wurde; ich fügte dem Programm bloß einen Satz bei: »Dieses Programm hat den Vorteil, daß es auf einer einfachen Schreibmaschine implementiert werden kann, die nicht mal mit dem Computer verbunden werden muß.«

Aber macht es keinen Unterschied, an welchem Ort und in welchem Zusammenhang ein Programm getestet wird?

Minsky: Ich verstehe nicht ganz, worauf Sie hinauswollen. Das Problem der Programme für Spracherkennung ist nicht das Testen, weil jeder weiß, daß sie nicht das Wissen haben, um sehr viel zu verstehen.

In den letzten dreißig, vierzig Jahren wurden Computer ziemlich forciert. Alle reden jetzt über Computer. Halten Sie es für notwendig, daß sich die Kommunikationsstrukturen durch die Kommunikation mit Computerprogrammen verändern?

Minsky: Bei einer neuen Wissenschaft, neuen Problemen oder einer neuen Technologie müssen Sie normalerweise eine neue Sprache lernen. Es gibt ein Computerprogramm namens Mathematica, das ganz nett ist für Ingenieur-Mathematik, aber unbrauchbar für Leute, die nichts von Algebra verstehen. Aber es gibt keinen Grund, warum Sie Ihre Sprache verändern sollten, nur weil Sie mit einem Computer reden. Wie Sie kommunizieren, hängt ganz davon ab, mit welchen Ideen und Kulturen Sie kommunizieren wollen und welches Wissen und welche Systemprozeduren im Computer stecken. Da kann man keine Generalisierungen machen.

Was können wir machen, damit Computer natürliche Sprachen verstehen können? Und warum können Menschen einfache Operationen wie Rechnen weniger gut und korrekt ausführen als der Computer?

Minsky: Ich glaube, daß bei Menschen nur etwa zehn Register einer bestimmten Art von Kurzzeitgedächtnis mit der Sprachmaschine verbunden sind. Das macht es für sie sehr schwer, zehnstellige Zahlen zu multiplizieren. Es ist eher bemerkenswert, daß wir so viele Dinge ohne Maschine machen können. Das ist so etwas wie ein Mysterium, man müßte es noch erklären.

Ich wollte auf etwas anderes hinaus. Angenommen, wir hätten mehrere Systeme mit jeweils verschiedenen Stärken. Könnten wir ein neues System konstruieren, indem wir immer das Beste aus jedem Einzelsystem zusammenfügen?

Minsky: Das ist das Thema meines Buchs. Die wichtigen Teile von »Society of Mind« handeln davon, daß man nicht nach einer universellen oder einheitlichen Theorie vom Denken suchen sollte, sondern genauso wie die Evolution verschiedene Lösungen sammeln sollte. Wenn Sie Tiere anschauen, finden Sie Tausende verschiedener Erfindungen, wie Knochen oder Muskeln gebaut werden könnten. Jedes Körpergewebe hat ungefähr dreitausend Gene, außer dem Gewebe des Nervensystems, das zehnmal soviel hat, also sind Gehirnzellen offensichtlich zehnmal komplizierter als andere. Wenn man also intelligente Maschinen entwickeln will, besteht das Hauptproblem darin, viele verschiedene Wege zu finden, wie Dinge gemacht werden können und die dann in irgendeiner Weise zu verbinden. Sigmund Freud hat zum Beispiel gesagt, daß ein wichtiger Teil des Geistes dieser Zensor sei, der Dinge unterdrückt, die als schlecht beurteilt werden. Das Problem mit modernen Expertensystemen ist, daß sie keine Sensoren haben. Man gibt Gesetze ein: »wenn dies passiert, mache das«, aber es gibt keine Gesetze, die sagen würden: »wenn dies passiert, mache das nicht.« Manchmal ist es besser, wenn man das schlechte Verhalten explizit entfernt.

Weizenbaum: Sie haben vorgeschlagen, das jeweils Beste von jedem System zusammenzufügen. Mit diesem Begriff »das Beste« setzen Sie aber bestimmte Absichten voraus. Wenn Sie von den besten Eigenschaften sprechen, zum Beispiel des Menschen, dann haben Sie eine bestimmte Sichtweise, ob Ihnen das bewußt ist oder nicht. Es könnte sein, daß diese Sichtweise nicht von allen geteilt wird, genauso wie der deutsche Versuch, einen Meisterweg für die »besten« Eigenschaften des Menschen zu finden. Diese Fragestellung ist nicht durchführbar und führt zu ganz großen Schwierigkeiten.

Minsky: Dazu ein Beispiel: Einer der größten Vorteile des Macintosh ist seine graphische Benutzeroberfläche mit Lanier-artiger Virtual Reality. Und die beste Eigenschaft des PC oder von MS-DOS ist, daß es Batch-Dateien gibt. Alle anderen Eigenschaften von MS-DOS sind meiner Meinung nach furchtbar, aber wie nimmt man die großartigen Eigenschaften der Batch-Dateien und verbindet sie mit der Macintosh-Benutzeroberfläche?

Der Begriff des »Besten« steht in Relation zur Absicht, wie Jo gesagt hat. Aber das schlimme ist, daß er auch in Relation zum System steht: Der Grund, warum der Macintosh gut ist, ist gleichzeitig auch der Grund dafür, warum er keine Batch-Datei hat, weil es sehr schwierig ist, der Batch-Datei zu sagen, wohin der Pointer bewegt werden soll. Natürlich können Sie Quickies kaufen oder Macro-Maker und all diese teuflischen Programme, die normalerweise funktionieren, um Ihre grafischen Eingaben in Batch-Datei-Eingaben zu verwandeln, aber nach sechs Wochen Arbeit damit entdecken Sie, daß es dem nicht ganz entspricht und – PENG! Sie wissen nicht, was falsch gelaufen ist. Furchtbar.

Übersetzung: Eva Weber

Biographien

Izumi Aizu

Mit japanischer Bescheidenheit bezeichnet sich Izumi Aizu als »ganz gewöhnlichen Anwender«, doch der 39jährige Japaner hat sich intensiv mit den Auswirkungen der Computernetze auf die Gesellschaft beschäftigt. Als Fachübersetzer zahlreicher EDV-Titel vom Amerikanischen ins Japanische stolperte Aizu »eher zufällig« in die Netzwerk-Szene. Ein Besuch in den USA machte ihn »süchtig«, scherzt der gelernte Drucker. Besonders der Kontakt zum kalifornischen Schriftsteller Howard Rheingold und dem San Francisco Netzwerk »WELL« prägten den Kommunikationsforscher Izumi Aizu. Mit Unterstützung des japanischen Handelsministeriums MITI gründete er das Institut für Netzwerk-Design (IND); in diesem »Think Tank« untersuchen Aizu und seine Mitarbeiter, wie in den nächsten Jahren die Vernetzung der Gesellschaft via PC und Telefonleitung das tägliche Leben verändern wird. Izumi Aizu ist Mitglied von Forschungsgruppen des MITI sowie des Ministeriums für Post und Telekommunikation in Japan.

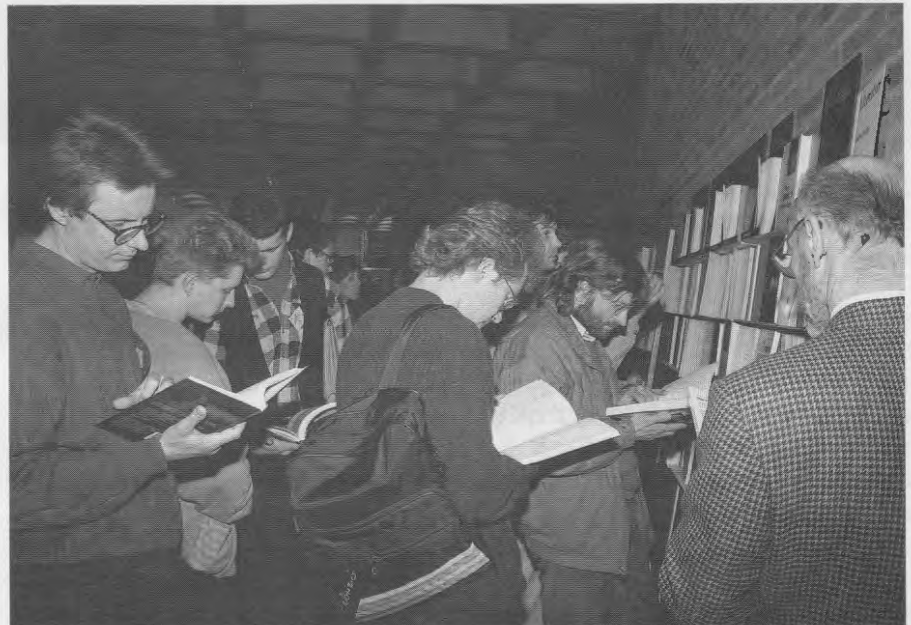
Literatur: The Personal Computer Network Revolution; 1986

Klaus Brunnstein

Klaus Brunnstein hat sich als Virenjäger einen Namen gemacht. Der 1937 in Köln geborene Diplom-Physiker war von 1964 bis 1973 Mitarbeiter am Rechenzentrum des Deutschen Elektronen-Synchrotrons (DESY) in Hamburg. Ab 1969 arbeitete er am Aufbau des Fachbereichs Informatik der Universität Hamburg, seit 1973 als Professor für Anwendungen der Informatik. Seit 1988 betreut Klaus Brunnstein das Virus Test Center in Hamburg. Er verfaßte zahlreiche wissenschaftliche Veröffentlichungen, zum Beispiel: Der Computer-Viren-Report, WRS-Verlag Freiburg 1989

Herbert W. Franke

Der Österreicher Herbert W. Franke gilt als Pionier der Computerkunst in Deutschland. 1927 in Wien geboren, promovierte er zum Dr. phil. an der Universität Wien, Technische Hochschule. Nach Forschungstätigkeit und einem kurzen Ausflug in die Wirtschaft arbeitete Franke ab 1956 als freier Fachpublizist. 1968 übernahm er die Leitung des Seminars »Kybernetik und Informationstheorie« an der Universität Frankfurt. Ein Jahr später folgten erste Arbeiten mit Computergraphik, und 1970 erhielt Herbert Franke von der Universität München einen Lehrauftrag für »kybernetische Ästhetik«. Der Computerkünstler und Science-Fiction-



Autor Franke rief 1979 als Mitbegründer das Festival Ars Electronica in Linz ins Leben. Noch heute gilt die »Ars« als wichtigstes Computerkunst-Forum. 1980 erhielt Franke den Berufstitel Professor vom Österreichischen Ministerium für Unterricht und Kunst. Herbert W. Franke ist berufenes Mitglied des deutschen PEN-Clubs und beratender Redakteur der Zeitschrift »Leonardo«. Weiterführende Literaturempfehlungen in Herbert W. Franke und Horst Helbig: Die Welt der Mathematik. Computergrafik zwischen Wissenschaft und Kunst; VDI-Verlag 1988

David Galloway

David Galloway, 54, ist seit 1960 als Kunstberater, Herausgeber von Kunstveröffentlichungen und freier Ausstellungsmacher tätig. 1972 wurde Prof. Dr. Galloway zum Ordinarius für Amerikanistik an der Ruhr-Universität Bochum berufen, 1977 bis 1978 war er Chefkustos des Teheran Museums of Contemporary Art. David Galloway, Fellow der Royal Society of Arts in London, begründete 1985 in Deutschland die Ausstellungsreihe »artware - Kunst und Elektronik«, und er organisierte das »Sommeratelier« Hannover. Neben zahlreichen Artikeln verfaßte Galloway wissenschaftliche Studien und schrieb einige Romane.

Herbert Heckmann

Herbert Heckmann, 61, hat sich als Literaturwissenschaftler mit dem Thema »Automaten in der Literatur« befaßt. Der gebürtige Frankfurter studierte in seiner Heimatstadt Philosophie, Geschichte und Literaturwissenschaft. Mit 26 promovierte Heckmann mit dem Thema »Elemente des barocken Trauerspiels«. Es folgten Lehraufträge an den Universitäten Münster und Heidelberg; zwischen 1963 und 1983 zeichnete Herbert Heckmann als Mitherausgeber der »Neuen Rundschau« verantwortlich. Seit 1980 lehrt er als Professor für Sprache und Literatur an der

Hochschule für Gestaltung in Offenbach. Außerdem ist Herbert Heckmann seit 1982 Präsident der Deutschen Akademie für Sprache und Dichtung in Darmstadt.

Literatur: Herbert Heckmann, Die andere Schöpfung. Geschichte der frühen Automaten in Wirklichkeit und Dichtung; Umschau Verlag 1982



Philippe Kahn

Starre Regelungen beeindruckten Philippe Kahn, 38, noch nie: Der französische Selfmade-Man begann seine Karriere, indem er ohne Arbeitserlaubnis eine Software-Firma in den USA gründete; berühmt wurde er, als er mit seinem Unternehmen Borland ein Übersetzungsprogramm für die Sprache Pascal so billig in den damaligen Hochpreis-Markt drückte, daß viele Konkurrenten glaubten: »dem geht in zwölf Monaten die Puste aus«. Kahn, der Anfang der siebziger Jahre bei dem Pascal-Vater Niklaus Wirth an der Eidgenössischen Technischen Hochschule studierte, spielte nachts in den Züricher Jazz-Clubs Flöte und Saxophon. Damals glaubte er noch, es könnte ein Science-Fiction-Schriftsteller in ihm stecken. Doch als er nach einem Jahr noch immer kein Buch fertig hatte, beendete er sein Mathematik-Studium. Nach einem kurzen Ausflug als Mathelehrer zog er ins gelobte Land der Computer: ins Silicon Valley. Im Chip-Tal südlich von San Francisco gründete Philippe Kahn 1983 die Firma Borland. Der erste große Er-

folg: Turbo Pascal, ein Übersetzer für die Wirth-Sprache Pascal, geschrieben von einem 19jährigen Kopenhagener Gymnasiasten. Heute ist Borland ein weltweit agierendes PC-Softwarehaus. Und Philippe Kahn liebt immer noch den Jazz: Unlängst hat er zusammen mit einigen Jazzberühmtheiten eine CD mit Titel »Pacific High« eingespielt.

Werner Künzel

Mit dem geschultem Auge des Programmierers hat Werner Künzel, Philosoph mit Informatik-Weiterbildung, die mittelalterliche Tradition einer technisch aufgeklärten Philosophie wiederentdeckt. Geboren wurde Werner Künzel 1951 in Duisburg; in Freiburg studierte er Philosophie, Kunstgeschichte und Archäologie. Nach der Promotion in Philosophie Stipendiat der Heinrich-Heine-Stiftung in Freiburg. Seit 1981 lebt Werner Künzel in Berlin. Momentan beschäftigt er sich mit der Geschichte

der Computertheorie in philosophischer Hinsicht.

Literatur:

Werner Künzel, Heiko Cornelius: Die Ars Generalis Ultima des Raymundus Lullus. Studien zu einem geheimen Ursprung der Computertheorie; Berlin 1986
Der Oedipus Aegyptiacus der Athanasius Kircher. Das ägyptische Rätsel in der Simulation eines barocken Zeichensystems; Berlin 1989; Neuauflage 1990
Werner Künzel, Peter Bexte: Gottfried Wilhelm Leibniz, Barock-Projekte. Maschinenwelt und Netzwerk im 17. Jahrhundert; Berlin 1990

Tod Machover

Er gilt als der profilierteste Musiker, der mit elektronischen Instrumenten spielt: Tod Machover, 37, gehört zu den bedeutendsten modernen Komponisten. »Niemand«, befand das Feuilleton der New York Times, »hat so erfolgreich daran gearbeitet, die elektronische Technologie zu humanisieren, wie Tod Machover.« Am New Yorker Juilliard Konservatorium lernte Machover bei Elliott Carter und Roger Sessions Cello und Komposition. Mit 27 ging er nach Paris an Pierre Boulez' »Instiut de Recherche et Coordination Acoustique/Musique« (IRCAM) im Centre Pompidou. Der europäische Ausflug endete 1985 mit einem Lehrauftrag des Massachusetts Institute of Technology als Leiter des dortigen »Experimental Media Facility«. Der Computer-Kom-





ponist nahm am Media Lab des MIT das »Hyperinstrument-Projekt« in Angriff: Mit »künstlich intelligenter« Software will Machover den Rechner zum »intelligenten Assistenten« machen, »der seine eigene Musik entwickeln kann«.

»MailBox-Frauen«

Katharina Baumann, 25, lebt und arbeitet in Hamburg. Studium der Geschichte und Kunstgeschichte. Vorträge am Förderungswerk Königstein zum Thema künstlicherischer Serienschmuck; zur Zeit Aufbau einer Mailbox für das Hamburger »Ponton European Media Art Lab«.

Rena Tangens lebt und arbeitet in Bielefeld: Super8, Video, Radio und Buchbeiträge. 1984 gründete sie mit Padaluun das Projekt und die Galerie Art d'Ameublement. 1984 Rahmenbau für Erik Saties »Pages Mystiques«, 1985 »Interregionale MEHRWERT Vorstellung« bei Art d'Ameublement, ab 1986 ernsthafter Umgang mit Computern, 1986 Performance mit ASA bei »d'art room« in Bologna, 1987 mit ASA auf der documenta,

»Chaos Communication Congress« in Hamburg, seit 1987 »Public Domain«, eine eigene Veranstaltungsreihe zu Randbereichen des Computers, 1988 auf Einladung des Canada Council »artist in residence« mit mehreren Einzelausstellungen in Kanada, 1989 Gründung der //BIONIC MailBox, 1990 Software-Lektorat für das Zerberus MailBox Programm, 1991 globale Dorfbrunnengestaltung.

Literatur zum Thema Frauen und Computer:

Micro Sisters: Digitalisierung des Alltags - Frauen und Computer; Ingrid Schöll, Ina Küller (Hrsg.); Elefant Press 1988

Frauenwelt - Computerräume. GI-Fachtagung Bremen, September 1989; Heidi Schelhowe (Hrsg.); Springer Verlag Berlin, Heidelberg 1989

Computerkultur und weiblicher Lebenszusammenhang. Zugangsweisen von Frauen und Mädchen zu neuen Technologien; Christiane Schiersmann, Hrsg. Bundesminister für Bildung und Wissenschaft; K.H. Bock Verlag 1987

Mensch und Computer, Zeitschrift für Bildung, Erziehung,

Kultur und Soziales, 2. Jahrgang 1989, Heft 1/89 (Frauen und Computer)

Miteinander reden - Störungen und Klärungen. Allgemeine Psychologie der Kommunikation; Friedemann Schulz von Thun; rororo Sachbuch Oktober 1981 Band 1

Designing society; Marianne Brun and correspondents; Princeton Editions, London 1985

Die Wunschmaschine - Der Computer als zweites Ich; Sherry Turkle; rororo Computer 8135, 1986 Politisch Arbeiten mit Computern; Gabriele Hooffacker; Rowohlt rororo computer 8184, 1991

Z-NETZ - jeden Tag neu - DATA +49-521-171188

Marvin Minsky

Früher als andere Informatiker hat sich Marvin Minsky, 63, der Idee verschrieben, Computer zu intelligenten Wesen zu machen. Schon Mitte der fünfziger Jahre reihte sich Minsky in den kleinen Zirkel von Computerwissenschaftlern ein, die über »künstliche Intelligenz« (KI) forschten. Schon als Harvard Student beschäftigte sich Minsky mit Theorien über Lernprozesse und setzte sich zum Ziel, eine Maschine zu bauen, die lernen kann. Damals gab es nur an drei Universitäten ernsthafte KI-Projekte: in Stanford, an der Carnegie Mellon Universität in Pittsburgh und am Massachusetts Institute of Technology. Marvin Minsky arbeitete in Massachusetts mit Seymour Papert zusammen. Mit John McCarthy gründete das Duo die MIT Artificial Intelligence Group, die Mitte der sechziger Jahre Roboter für industrielle Automation entwickelte. Bald fand die Gruppe heraus, daß die bekannten Programmier Techniken für die KI unzureichend sind. So entwarfen die Forscher einen dezentralen Programmier-Ansatz. Aus diesen Ansätzen, verbunden mit den Theorien der kognitiven Psychologie und der Verhaltensforschung, entwickelte Marvin Minsky eine Erkenntnistheorie, die er 1987 in seinem



Buch »The Society of Mind« veröffentlichte (deutsch: Mentopolis, Klett Cotta Verlag 1990).

Zum Weiterlesen: von Marvin Minsky und Joseph Weizenbaum erwähnte Autoren:

Noam Chomsky (u.a. Aspekte der Syntax-Theorie, 1983; Reflexionen über die Sprache, 1977; Regeln und Repräsentationen, 1981; Sprache und Geist, 1973; alle Suhrkamp TB)

Daniel C. Dennett (und Douglas R. Hofstadter: Einsicht ins Ich. Fantasien und Reflexionen über Selbst und Seele; Klett-Cotta 1986)

Hubert L. Dreyfus (Die Grenzen künstlicher Intelligenz. Was Computer nicht können; Athenäum Verlag 1985)

Hans Moravec (Mind Children. Der Wettlauf zwischen menschlicher und künstlicher Intelligenz; Hoffmann und Campe 1990)

Allen Newell (und Herbert Simon: Human Problem Solving, Pentice Hall 1972)

Seymour Papert (Gedankenblitze. Kinder, Computer und Neues Lernen, Rowohlt TB 1985)

Roger Penrose (The Emperor's New Mind. Concerning computers, minds, and the laws of physics, Vintage 1989)

Roger C. Schank (Die Zukunft der künstlichen Intelligenz. Chancen und Risiken, DuMont 1986)

John R. Searle (u.a. Ausdruck und Bedeutung. Untersuchungen zur Sprechaktttheorie, 1982; Geist, Hirn und Wissenschaft,

1986; Intentionalität. Eine Abhandlung zur Philosophie des Geistes, 1987; alle Suhrkamp)

Thomas von Randow

Thomas von Randow, Jahrgang 1921, hat die erste Hälfte seines Lebens als Mathematiker verbracht, unter anderem am Massachusetts Institute of Technology, wo er sich hauptsächlich der statistischen Informationstheorie und dabei auch den mathematischen Methoden der Kryptologie widmete. Nach seiner Rückkehr in die Bundesrepublik Anfang 1961 schrieb er auf Einladung der Wochenzeitung »Die Zeit« einen Artikel - und wurde dort Redakteur für Wissenschaft. Das blieb er 28 Jahre lang. Jetzt ist er im Ruhestand als freier Journalist und Buchautor tätig.

David Rokeby

Multitalent David Rokeby als Kommunikationskünstler zu bezeichnen, wäre eine Untertreibung. Der 31jährige Kanadierer ist nicht nur Musiker, sondern auch Hardware-Tüftler und Programmierer. In Kanada gewann er mit seinen Ausstellungen, Soundtracks und Installationen zahlreiche Preise und Ehrungen. David Rokeby will sein Publikum in die Performance ziehen, seine Klanginstallationen funktionieren nur mit und durch die Mitspieler. Zahlreiche Installationen in Kanada, den USA, Japan und Europa bestimmen die Karriere des Musikers.

Bjarne Stroustrup

Bjarne Stroustrup hat die Programmiersprache C++ entwickelt, die wesentliche Erweiterungen gegenüber dem Vorgänger C bietet. Als objektorientierte Sprache erlaubt C++ eine ganz neue Art des Programmdesigns. Der dänische Forscher hat in Aarhus Informatik und Mathematik studiert, 1979 promovierte er an der Cambridge Universität, England, in Informatik. Sein Schwerpunkt sind verteilte Systeme, Betriebssysteme, Simulationen, Programmiersprachen und die Methodik des Programmierens. Seit 1979 arbeitet er im technischen Forschungszentrum der AT&T Bell Laboratories in Murry Hill, USA.

Literatur:

Stroustrup, Bjarne: The C++ Programming Language; Addison-Wesley 1986, 2nd Edition 1991

Hansen, Tony: The C++ Answer Book; Addison-Wesley 1989

Lippman, Stanley B.: C++ Einführung und Leitfaden; Addison-Wesley 1990

Dewhurst, Stephen und Stark, Kathy: Programming in C++; Prentice-Hall, Englewood Cliffs New Jersey, 1989

Eckel, Bruce: Using C++; Osborne McGraw-Hill 1989

Im Beitrag »Objektorientiertes Programmieren« wird auf folgende Quellen verwiesen:

1) Birtwistle, Graham et.al.: SIMULA BEGIN. Studentlitteratur, Lund, Sweden. 1971 Chartwell-Bratt Ltd, UK. 1980.

2) Dahl, O.-J. and Hoare, C.A.R.: Hierarchical Program Structures. In Structured Programming. Academic Press 1972.

3) Cargill, Tom A.: PI: A Case Study in Object-Oriented Programming. SIGPLAN Notices, November 1986, pp 350-360.

4) C.C.I.T.T Study Group XI: CHILL User's Manual. CHILL Bulletin no.1 vol.4 March 1984.

5) Ellis, M.A. and Stroustrup, B.: The Annotated C++ Reference Manual. Addison-Wesley 1990.

6) Goldberg, A. and Robson, D.: Smalltalk-80: The Language and its Implementation. Addison-Wesley 1983.

7) Ichbiah, J.D. et.al.: Rationale for the Design of the Ada Programming Language. SIGPLAN Notices, June 1979.

8) Kernighan, B.W. and Ritchie, D.M.: The C Programming Language. Prentice-Hall 1978. 2nd Edition 1988.

9) Keene, Sonya E.: Object-Oriented Programming in COMMON LISP. Addison-Wesley 1988.

10) Kerr, Ron: Object-Based Programming: A Foundation for Reliable Software. Proceedings of the 14th SIMULA User's Conference. August 1986, pp 159-165. Eine kürzere Fassung dieses Aufsatzes findet man unter dem Titel: A Materialistic View of Software »Engineering« Analogy in SIGPLAN Notices, March 1987, pp 123-125.

11) Liskov, Barbara et.al.: Clu Reference Manual. MIT/LCS/TR-225, October 1979.

12) Liskov, Barbara et.al.: Abstraction Mechanisms in Clu. CACM vol 20, no 8, August 1977, pp 564-576.

13) Milner, Robert: A Proposal for Standard ML. ACM Symposium on Lisp and Functional Programming. 1984, pp 184-197.

14) Nygaard, Kristen: Basic Concepts in Object Oriented Programming. SIGPLAN Notices, October 1986, pp 128-132.

15) Rovner, Paul: Extending Modula-2 to Build Large, Integrated Systems. IEEE Software, Vol. 3, No. 6, November 1986, pp 46-57.



16) Shopiro, Jonathan: Extending the C++ Task System for RealTime Applications. Proc. USENIX C++ Workshop, Santa Fe, November 1987.

17) SIMULA Standards Group, 1984: SIMULA Standard. ASU Secretariat, Simula a.s. Post Box 150 Refstad, 0513 Oslo 5, Norway.

18) Snyder, Alan: Encapsulation and Inheritance in Object-Oriented Programming Languages. SIGPLAN Notices, November 1986, pp 38-45.

19) Stroustrup, Bjarne: The C++ Programming Language. Addison-Wesley, 1986. 2nd Edition 1991.

20) Stroustrup, Bjarne: Multiple Inheritance for C++. Proceedings

of the Spring '87 EUUG Conference. Helsinki, May 1987.

21) Stroustrup, Bjarne: The Evolution of C++: 1985-1987. Proc. USENIX C++ Workshop, Santa Fe, November 1987.

22) Stroustrup, Bjarne: Possible Directions for C++: 1985-1987. Proc. USENIX C++ Workshop, Santa Fe, November 1987.

23) Weinreb, D. and Moon, D.: Lisp Machine Manual. Symbolics, Inc. 1981.

24) Wirth, Niklaus: Programming in Modula-2. Springer Verlag, 1982.

25) Woodward, P.M. and Bond, S.G.: Algol 68-R Users Guide. Her Majesty's Stationery Office, London. 1974.



Joseph Weizenbaum

Joseph Weizenbaum, 68, hat sich wie kaum ein anderer Informatikforscher mit den sozialen Folgen der Computertechnik beschäftigt. 1923 in Berlin geboren, emigrierte er 1936 mit seiner Familie in die USA.

Nach dem Ende des zweiten Weltkriegs beendete er sein Mathematik-Studium in der Automobil-Stadt Detroit und verschrieb sich der Informatik.

1963 erhielt er am Massachusetts Institute of Technology in Cambridge einen Lehrauftrag als Professor für Informatik. Am MIT, dem »größten Spielfeld der Welt« (Weizenbaum), der er-



hebliche Forschungsmittel vom amerikanischen Militär erhält, stellte Weizenbaum die Frage nach der Verantwortung der Wissenschaft gegenüber der Gesellschaft. Der Mitbegründer des deutschen »Forums der Informatiker für Frieden und gesellschaftliche Verantwortung« und der amerikanischen Gruppe »Computer Professionals for Social Responsibility« wurde zum personifizierten Gewissen der Computerszene, und er wehrte sich zugleich gegen die Rolle des einsamen Alibikritikers.

Literatur in deutscher Sprache: Die Macht der Computer und die Ohnmacht der Vernunft; Suhrkamp Verlag 1977) (englisch: Computer Power and Human Reason)

Kurs auf den Eisberg - Oder das Wunder wird uns retten, sagt der Computerexperte; Pendo Verlag Zürich 1984

Kurs auf den Eisberg - Die Verantwortung des Einzelnen und

die Diktatur der Technik; Piper Verlag München 1987
Computer im Alltag. Joseph Weizenbaum, Klaus Haefner; herausgegeben von Michael Haller; Pendo Verlag Zürich 1987

Sind Computer die besseren Menschen? Ein Streitgespräch zwischen Joseph Weizenbaum und Klaus Haefner. Herausgegeben von Michael Haller, Pendo Verlag Zürich 1990

Naturwissenschaft oder Esoterik. H. Pitschmann, F. Paschke, K. Apel, J. Weizenbaum, Verlag Leuschner und Lubensky 1990

Niklaus Wirth

Niklaus Wirth ist einer der bedeutendsten europäischen Informatiker. Die von ihm entwickelte Computersprache Pascal prägte entscheidend den Arbeitstil moderner Programmierer. Der Schweizer beendete 1958 sein Studium an der Eidgenössischen Technischen Hochschule in Zürich als Diplom-Elektroingenieur

und erwarb in Kanada zwei Jahre später seinen »Master of Science.« 1963 erhielt Wirth an der renommierten kalifornischen Universität Berkeley seinen Dr. phil. Vier weitere Jahre lehrte er als Assistenzprofessor an der Stanford Universität in Californien. Dort entwickelte er die Programmiersprachen PL360 und Algol W. 1968 wechselte Professor Wirth zur ETH Zürich, wo er Pascal und Modula entwarf. In den letzten fünf Jahren entwickelte Niklaus Wirth zusammen mit seinem ETH-Kollegen Jürg Gutknecht das Oberon-System.

Literatur:

Martin Reiser, The Oberon System. User Guide and Programmer's Manual; Addison-Wesley Publishing Company 1991

Hans-Jürgen Zimmermann

Kaum ein Name ist in Deutschland mit der Zukunftswissenschaft »Fuzzy Logic« so verknüpft wie Hans-Jürgen Zimmermann. Nachdem der heute 57jährige Wirtschaftsingenieur einige Jahre lang als Professor in den USA arbeitete, übernahm er 1967 den Lehrstuhl für Operations Research an der Rheinisch-Westfälischen Technischen Hochschule in Aachen. Dort ist er auch Direktor des Instituts für Wirtschaftswissenschaften. Zimmermann war Gründungspräsident der deutschen und europäischen Gesellschaften für Operations Research wie auch der weltweiten »Fuzzy Systems Association«:

Hans-Jürgen Zimmermann hat zahlreiche Bücher und Aufsätze zum Thema der »Fuzzy Sets« veröffentlicht und zeichnet als Herausgeber der größten internationalen Zeitschrift über dieses Fachgebiet verantwortlich. Seine derzeitigen Forschungsschwerpunkte: Anwendungen von OR-Techniken auf technischen und wirtschaftswissenschaftlichen Gebieten, rechnergestützte Informationssysteme, unscharfe Entscheidungen, empirische Entscheidungstheorie, Stichproben-Inventuren, Expertensysteme, künstliche Intelligenz.

Erstes European Software Festival

Marvin Minsky	Die Geistesmaschine
Marvin Minsky	Die Zoologie des Gehirns
Joseph Weizenbaum	Gegen die Prostitution des Geistes
Tod Machover	Balanceakt im Klangjungle
Tod Machover	Konzert mit sensibler Software
Bjarne Stroustrup	Der produktive Weg zu C++
Bjarne Stroustrup	Objektorientiertes Programmieren
Niklaus Wirth	»Die Dinge beim Namen nennen«
Niklaus Wirth	Das Oberon System
Philippe Kahn	Turbo-Kommunikation im Jazzrhythmus
Das Bonner Klangkollektiv	Oekophon und Café Hypermedia
Klaus Brunnstein	Computer-Unfälle
Hans-Jürgen Zimmermann	Fuzzy-Logik für unsichere Inhalte
David Rokeby	Tanz in der Feedback-Schleife
Izumi Aizu	Auf dem Weg zur Hypernetzwerk-Gesellschaft
Rena Tangens, Katharina Baumann	Mailboxen – keine Männerkiste
Herbert Heckmann	Literarische Automaten
Thomas von Randow	Maria Stuart hätte ihren Kopf behalten können, wenn ...
Werner Künzel	Zur Geschichte der Logik-Maschine
David Galloway	Die Muse in der Steckdose
Herbert W. Franke	Die Ästhetik der programmierten Kunst
Diskussion mit Marvin Minsky und Joseph Weizenbaum	Künstliche Wunderwelten, natürliche Verständigungsprobleme

ISBN N 3-8023-1146-9 DM+028.00

DM 28,-

SFR 28,-

ÖS 230,-

1.Auflage



VOGEL



9 783802 311468

02800

