

Der Computer als Werkzeug und Medium

Michael Friedewald

**Der Computer als
Werkzeug und Medium**
**Die geistigen und technischen Wurzeln
des Personalcomputers**

Berlin Diepholz 1999

Verlag für Geschichte
der Naturwissenschaften und der Technik

Deutsche Bibliothek – CIP-Einheitsaufnahme:

Friedewald, Michael:

Der Computer als Werkzeug und Medium : Die geistigen und technischen Wurzeln des Personal Computers / Michael Friedewald. – Berlin ; Diepholz : Verl. für Geschichte der Naturwiss. und der Technik, 1999

(Aachener Beiträge zur Wissenschafts- und Technikgeschichte des 20. Jahrhunderts ; Bd. 3)

Zugl.: Aachen, Techn. Hochsch., Diss., 1999

ISBN 3-928186-47-7

ISBN 3-928186-47-7

Printed in Germany. Alle Rechte vorbehalten

Inhaltsverzeichnis

Wolfgang Coy: Zum Geleit	11
1. Einleitung	15
1.1 Motivation und Fragestellung	15
1.2 Stand der Forschung	18
1.3 Akteure und Leitbilder	20
1.4 Vom Automaten zum Werkzeug und Medium	25
2. Vision und Kontinuität: Vannevar Bushs Memex	35
2.1 Vannevar Bush: Ingenieur, Manager und Politiker	35
2.2 Der Ingenieur zwischen Theorie und Praxis	39
2.2.1 Ein Vertreter der alten Ordnung	39
2.2.2 Der Technokrat	42
2.3 Das Informationsproblem	43
2.4 ... und erste Lösungsversuche	46
2.4.1 Der Navy Comparator	46
2.4.2 Der Rapid Selector	47
2.5 »As we may think« und der Memex	51
2.5.1 Das fiktive Gerät	52
2.5.2 Der Mikrofilmspeicher	56
2.5.3 Eingabetechnik	57
2.5.4 Assoziative Datenspeicherung	61
2.5.5 Verwendung des Memex	64
2.6 »The day is not yet here«: Das Schicksal des Memex	67
2.7 Das Vermächtnis des Memex	69

3.	Vom Analogrechner zum Minicomputer	73
3.1	Die kybernetische Revolution	74
3.1.1	Information und Rückkopplung	75
3.1.2	Kybernetik und Computer	77
3.2	Whirlwind: Der erste Echtzeitcomputer	78
3.2.1	Vom Analog- zum Digitalcomputer	79
3.2.2	Der Whirlwind-I-Computer	81
3.2.3	Ein Computer auf der Suche nach Anwendungen	85
3.2.4	Wie Ingenieure Computer entwerfen	90
3.3	SAGE: Ein komplexes Mensch-Maschine-System	95
3.3.1	Mensch-Computer-Kommunikation	96
3.3.2	Systematische Programmentwicklung	105
3.4	Whirlwinds Nachfahren	110
3.4.1	Memory Test Computer	110
3.4.2	Lincoln TX-0 und TX-2	111
3.4.3	Digital Equipment und der Minicomputer	114
4.	Mensch-Computer-Symbiose: Joseph Licklider und die Denkmaschine	119
4.1	Der Computer als Partner des Menschen	119
4.2	Interaktive Computer für das Pentagon	124
4.3	Time-Sharing	127
5.	Douglas Engelbart und der Computer als Intelligenzverstärker	139
5.1	Memex, Radar und eine Vision	139
5.2	Die Quellen der Inspiration: Neue Theorien und altbekannte Maschinen	143
5.3	»Augmenting Human Intellect: A Conceptual Framework«	149
5.3.1	Der Computer als Medium	152
5.3.2	Eine universelle Sprache für die Mensch-Computer-Kommunikation	154
5.3.3	Bootstrapping: Eine evolutionäre Entwicklungsmethodik	159
5.3.4	Kurzfristige und langfristige Ziele	162
5.4	Geld von ARPA und NASA	163
5.5	Die Gestaltung der Mensch-Computer-Schnittstelle	168
5.5.1	Dateneingabe: Möglichkeiten und Widerstände	171
5.5.2	Revolution oder Verschwendung: Arbeiten am Bildschirm	175
5.5.3	Grafische Eingabegeräte: Von Mäusen und Käfern	176
5.6	Servicesystem und Benutzersystem	185
5.7	Das Offline-Text-System	186

5.8	Das Online-System: Prototyp des Intelligenzverstärkers	191
5.8.1	Ein Time-Sharing-Computer für das Online-System	191
5.8.2	Konvergenz von Fernsehen und Computer	192
5.8.3	Bootstrapping in der Praxis: Das NLS-Softwaresystem	200
5.8.4	Mehr als Textverarbeitung	206
5.8.5	Arbeiten mit dem NLS	211
5.8.6	San Francisco, 9. Dezember 1968: Douglas Engelbarts große Show	214
5.8.7	Erste Anwendungen	218
5.9	Vom Individuum zur Gruppe: Die <i>Online Community</i> und das ARPANET	220
5.9.1	Das Journal	220
5.9.2	ARPANET und Network Information Center	223
5.9.3	Der Exodus	230
5.9.4	Epilog	234
6.	Xerox PARC und die Architektur der Information	237
6.1	Das Kopiererimperium im Computerzeitalter	237
6.1.1	Diversifizierung und das Büro der Zukunft	237
6.1.2	Die Elite der Computerwissenschaftler	240
6.2	Time-Sharing: Kontinuität am PARC	245
6.3	Das Dynabook: Ein Computer für Kinder in jedem Alter	249
6.4	Der Alto: Ein Personal Computer	261
6.4.1	Der Prozessor	263
6.4.2	Der Grafikbildschirm	269
6.4.3	Eingabegeräte: Tastatur und Maus	273
6.4.4	Die Weiterentwicklung des Alto	275
6.4.5	Systemsoftware	276
6.5	Von ARPANET zu Ethernet und wieder zurück	280
6.5.1	ALOHAnet und Paketvermittlung	280
6.5.2	Ethernet: Ein experimentelles lokales Netzwerk	282
6.5.3	Vernetzte Netze	286
6.5.4	Clients und Server: Ein verteiltes System	288
6.6	Bausteine für das Büro der Zukunft	292
6.6.1	Drucken mit Licht und Schatten: der Laserdrucker	292
6.6.2	Benutzerfreundliche Textverarbeitung	297
6.6.3	Grafikprogramme von Konstruktion bis Kunst	305
6.7	Smalltalk und die grafische Benutzungsoberfläche	311
6.7.1	Eine Programmiersprache für wen?	311
6.7.2	Programmieren mit Objekten	313

6.7.3	Smalltalk-72 und die Kinder	316
6.7.4	Programmentwicklung in Smalltalk	321
6.7.5	Die Evolution von Smalltalk	323
6.7.6	Benutzerillusion und grafische Benutzungsoberfläche	326
6.8	Der lange Weg zum Produkt	336
6.8.1	Das Ende von <i>Xerox Data Systems</i>	336
6.8.2	Die Ablehnung des Alto	337
6.8.3	Der NoteTaker	338
6.8.4	Der Dorado	341
6.8.5	Der Xerox Star: Das Dokument ist das Herz der Welt	341
6.9	Das Labor, das Xerox davonlief	351
7.	Die Computer für den Rest von uns	355
7.1	Mikroprozessoren, Bastler und Homecomputer	355
7.1.1	Der Volkscomputer	355
7.1.2	Taschenrechner	360
7.1.3	Der Mikroprozessor	361
7.1.4	Die ersten Mikrocomputer	364
7.1.5	Software für Homecomputer: Basic und CP/M	367
7.1.6	Der Aufstieg von Apple	370
7.1.7	Der späte Start von IBM	374
7.2	Die Lisa – Apples Computer der Zukunft	377
7.2.1	Die technischen Herausforderungen der Hardware	381
7.2.2	Die Schwierigkeiten einfacher Software	384
7.2.3	Enthusiasmus, Enttäuschung und Hoffnung	390
7.3	Small is beautiful: Der Apple Macintosh	393
7.3.1	Lisas kleiner Bruder	393
7.3.2	Apples Marketing-Blitzkrieg und die Killerapplikation	398
7.3.3	Konkurrenten und Imitatoren	405
8.	Das Ende der Entwicklung?	409
Anhang	413
A.	Übersicht über die finanzielle Förderung der SRI-ARC Projekte, 1961–1967	414
B.	Übersicht der NLS Befehle, Stand 1967	417
C.	Nutzung von NLS und Augment	421
D.	Butler Lampson: »Why Alto«, Dezember 1972	424
E.	Die Entwicklung des Macintosh, 1979–1984	428
Über den Autor und dieses Buch	431

Literaturverzeichnis	433
Quellenverzeichnis	477
Abkürzungsverzeichnis	485
Tabellenverzeichnis	486
Abbildungsverzeichnis	489
Sach- und Namensregister	493

The world has arrived at an age of cheap complex devices of great reliability; and something is bound to come of it.

VANNEVAR BUSH, 1945.

Computers today are smaller, faster, cheaper, and more reliable, but they are essentially the same machines we had 30 years ago. The important question has been – and continues to be – how we use them.

JAY W. FORRESTER, 1995.

Zum Geleit

Computer umgeben uns überall, die meisten unsichtbar – als eingebettete Systeme in Autos, Aufzügen, Telefonen oder Kameras. Gleichzeitig werden sie überall sichtbar – im Arbeitszimmer, im Wohnzimmer, im Klassenzimmer, im Fernsehen, im IKEA-Katalog. Zuerst standen sie in den klimatisierten Kellern großer Unternehmen, zugänglich nur für die Eingeweihten nach Überwindung komplexer Zugangssperren. Der Rest konnte sie im Film bewundern. In den Achtzigern standen sie plötzlich auf allen Schreibtischen. »Wow, my own IBM computer! Imagine that!« verhiess die Reklame. Inzwischen sind sie tragbar: »Computers for the rest of us!« Dieses Vorwort schreibe ich in einem ICE, computertauglich gemacht durch eine Steckdose am Sitzplatz – obwohl die Batterie meines *Powerbooks* dies nicht wirklich braucht.

Sind dies alles Erscheinungsformen des gleichen Geräts? Das wird man nicht ohne Weiteres behaupten können. Ein *embedded controller* ist anders gebaut als ein *Mainframe*, und beide unterscheiden sich von einer *Workstation* oder einem PC. Sicher gehören sie der gleichen digitalen Technologie an, doch Mainframes sind tief im Innern Rechenautomaten und Datenverarbeitungsmaschinen geblieben, gewandelt zu großen Archivmaschinen oder zu *Servern* in internen und externen Netzen. Die eingebetteten Varianten des Mikroprozessors heißen zu recht *Mikrocontroller*; sie haben mechanische oder elektromechanische Steuerbausteine durch allgemeinere, programmierbare Chips und Boards abgelöst. Workstations und PCs sind dagegen multimediale Endknoten in einem neuen Medium geworden, dem Internet. Sie besetzen nun den Begriff Computer so sehr, daß Erstsemester gelegentlich Bill Gates für den Erfinder des Computers halten – wie mir amerikanische Kollegen glaubhaft versichern.

Aber wer hat den PC erfunden? Wo kommen diese allgegenwärtigen, alltäglichen Medienmaschinen her? Michael Friedewald geht dieser Frage als Technikhistoriker nach – und er findet Antworten, die weit über noch vorhandene *Oral History* oder die gängigen journalistischen Aufbereitungen hinaus führen. Der PC ist *nicht* die bruchlose Fortsetzung der Mainframes als mikroelektronische Kopie, wie es selbst Informatiker sehen möchten, sondern der technische und kulturelle

Schnittpunkt unterschiedlicher Linien der Software- und der Geräteentwicklung aber auch unterschiedlicher Anwendungslinien.

In der vorliegenden Darstellung wird klar, daß der PC eine eigenständige Geschichte hat, die ebenso wie die Geschichte der Großrechner in den Zweiten Weltkrieg zurückweist. Nur: Es waren nicht John v. Neumann und Howard Aiken, die den PC erahnt haben, sondern Vannevar Bush (und, wenn ich das hinzufügen darf, auch Konrad Zuse mit seinem früh angedachten *Graphomaten*). Der PC kommt also eher vom MIT, als aus Princeton oder aus Harvard. Es ist vielleicht kein Zufall, daß Vannevar Bush tief in der Analogrechenstechnik verwurzelt war und Digitalrechnern zutiefst mißtraute. Weshalb er sein *Memex* als digital gesteuertes Archiv analog repräsentierter Texte und Bilder konzipierte — bis schließlich Forresters *Whirlwind*, als digitale Basis des radargesteuerten Raketenfrühwarnnetzes SAGE ebenfalls am MIT entwickelt, im Monat mehr Geld verplanen durfte als Bush in einem ganzen Jahr zur Verfügung stand.

Bush konnte seine Ideen eines persönlichen Archivierungssystems nicht umsetzen. Seine Vorstellungen von persönlicher Software und persönlichen Geräten fanden aber andere Wege zu ihrer Realisierung, ironischerweise auch über die militärische Forschungsförderung. Joseph C. R. Licklider war sowohl Psychologe wie Mathematiker und Physiker, mit einem Arbeitsplatz am MIT. Dort lernte er nicht nur Digitalcomputer aus erster Hand kennen, er erkannte auch die Möglichkeit, oder soll man sagen: die Notwendigkeit, diese Maschinen als persönliche Arbeitshilfen weiter zu entwickeln. Die »Man-Computer Symbiosis« wurde zu seiner Leitidee. Die Chance, diese Erkenntnis umzusetzen, bot sich ihm durch den Wechsel zur *Advanced Research Projects Agency* (ARPA), wo er 1962 zum Direktor des *Information Processing Techniques Office* ernannt wurde. Unter seiner Leitung wurden grafik- und interaktionsfähige Maschinen entwickelt, aus denen mit Hilfe des 1971 erfundenen Mikroprozessors diejenigen Kleinstrechner wurden, die man heute als PCs bezeichnet. Andere Entwicklungen haben dies verstärkt. Ivan Sutherlands grafisches Eingabesystem *Sketchpad*, Doug Engelbarts *Mouse*, das *Online Text System*, der speicherfressende Rasterbildschirm und vor allem die Arbeiten Alan Kays und anderer bei *Xerox PARC*, die im Apple Macintosh einen ersten Massenmarkt fanden, der wiederum zum Vorbild der Windows-Varianten wurde. Als WIMP-Machine (*Windows, Icons, Menus, and Pointers*) wurde der Computer doch noch zum Archivsystem für Text und Bild – und vielem mehr: für Töne, Animationen, Filme.

Natürlich ging es nicht nur um die multimediale Geräteausstattung, es mußten auch geeignete Programmierhilfen gefunden werden. Der Durchbruch vom Hobbygerät zur Alltagstechnik gelang den frühen PCs (bevor IBM sie PCs nannte) mittels Texteditoren wie *Wordstar*, aber vor allem mittels *VisiCalc*, dem ersten vermarkteten Tabellenkalkulationsprogramm – einem Programmtyp, den es

für Mainframes und Minicomputer vorher nicht gab, einer Killerapplikation also. Nicht: »My own IBM computer!«, sondern »Meine selbst programmierbare Buchhaltung!«

Um die wilden Ideen einer Multimedia-Maschine umsetzen zu können, wurde *Smalltalk* als Sprache der Wahl bei Xerox PARC entwickelt, andere, wie Apples *HyperTalk*, führten diesen Einstieg fort, bis das WWW und in seinem Gefolge *Java* das Internet als zwingende Erweiterung des Tischmediums PC charakterisierte. Vernetzung, Interaktion, multimediale Datenspeicherung und -verarbeitung kennzeichnen den Stand des Mediums Computer heute – ein Medium, das alle andere simulieren kann, sie aber auch auflöst. Ein weltweit verbreitetes Medium, das doch das Ende der Massenmedien andeutet – natürlich nicht im Rückgriff auf die individuelle Handarbeit, sondern in Form eines *mass-customized medium*, eines Mediums, daß für jeden Nutzer und jede Nutzerin in gefälliger Weise zurechtgestutzt werden kann. *The medium for the rest of us!* Sie wollen Infos über Perserkatzen: Bitte hier ist www.perserkatzen.de; sie wollen mehr Nachrichten über Leguane: Bitte hier ist alt.pets.reptiles.lizards.iguana (leider nur in Englisch) – und Sie können Fragen stellen oder Kommentare schicken. Sie wollen *keine* Sex- oder Gewaltdarstellungen aus dem Internet: Bitte Microsofts *Internet Explorer* enthält ein passendes Filterprogramm. Sie wollen *nur* Sexbilder: Drehen Sie doch einfach den Filter um ;-)

Es ist Michael Friedewalds Verdienst, diese Entwicklung in ihrer Kontinuität, aber vor allem in ihren Brüchen klar und deutlich zu zeigen. Sein Text beweist, daß die Geschichte des PCs bereits zu einem Stück Geschichte geworden ist. Er legt die relevanten Fakten offen – jenseits aller Spekulationen über den Charakter einer kommenden »Informationsgesellschaft«.

Prof. Wolfgang Coy
Humboldt-Universität zu Berlin
Institut für Informatik, Informatik in Bildung und Gesellschaft

1. Einleitung

The Serpent [to Eve]:

When you and Adam talk, I hear you say »Why?« Always »Why?« You see things; and say »Why?« But I dream things that never were; and I say »Why not?«

George Bernard Shaw, *Back to Methuselah*¹

1.1 Motivation und Fragestellung

Wenn heute vom Computer gesprochen wird, ist meist der kleine Computer mit Tastatur, Bildschirm und Maus gemeint, der seit 1981 den Weg in Millionen Büros und Haushalte gefunden hat. In den allermeisten Fällen wird dieses Gerät zur Textverarbeitung, für Kalkulationen und nicht zuletzt zum Spielen verwendet. Mit dem Siegeszug des Internets ist es seit Mitte der 1990er Jahre auch zunehmend zum Medium für Kommunikation und Information geworden. Diese Geräte werden in der Regel als *Personal Computer* (PC) bezeichnet.²

Zuvor war für fast 40 Jahre ein ganz anderes Bild vom Computer vorherrschend. Computer waren riesige technische Geräte, mit deren Betrieb ein ganzes Team von Technikern, Operateuren und Programmierern beschäftigt war. Sie waren so groß und teuer, daß sich nur große Unternehmen, Behörden und das Militär Computer leisten konnten, die damit komplizierte Berechnungen anstellten oder

¹ Shaw 1990, S. 67.

² »Personal Computer« war zunächst lediglich der Name eines erfolgreichen Produkts, das die *International Business Machines Corp.* (IBM) 1981 auf den Markt brachte. Die Bezeichnung Personal Computer bürgerte sich aber schnell als Synonym für eine bestimmte Klasse von preiswerten Computern aller Hersteller ein, zu denen man auch die leistungsfähigeren *Workstations* rechnen kann. Vgl. Bell 1988, S. 7ff.

Da der Personal Computer hier als das vorläufige Endprodukt eine vierzigjährigen Entwicklung verstanden wird, macht es keinen Sinn, unter Hinweis auf die Größe oder verwendete Technologie von einem bestimmten Computer als dem »ersten PC« zu sprechen, wie man dies immer wieder in der Literatur findet, z. B. bei Olsen (1988), Lehmann (1988), Alpar (1993) oder Wood (1994).

1. Einleitung

große Datenmengen verwalteten. Obwohl im Lauf der Jahre immer mehr Menschen z. B. in Form von Lohn- und Gehaltsabrechnungen mit der elektronischen Datenverarbeitung in Berührung kamen, blieb der Computer für die allermeisten eine Technik weit jenseits des eigenen Erfahrungs- und Verständnishorizonts. Für sie handelte es sich gleichzeitig um eine magisch-faszinierende als auch als bedrohlich empfundene neue Technologie.¹

Diese Großcomputer oder *Mainframes* haben zwischen 1945 und 1975 eine beispiellose Karriere gemacht. Nach einer Zeit erster Entwürfe und Prototypen wurden sie ab 1950 industriell hergestellt. 1955 waren in den Vereinigten Staaten immerhin schon 240 Großcomputer in Betrieb. 1974, noch vor der Markteinführung der ersten Personal Computer, wurde der Bestand allein in den USA auf 165 000 geschätzt.²

Obwohl Großcomputer und Personal Computer im wesentlichen die gleiche logische Grundstruktur besitzen, aus den gleichen elektronischen Bauelementen aufgebaut sind und auf ähnliche Weise programmiert werden, handelt es sich doch um Artefakte, die nichts miteinander zu tun haben. Trotz aller Ähnlichkeiten ist der Personal Computer aus einer ganz anderen Kultur hervorgegangen.³ Dies betrifft sowohl den Entwurf und die Konstruktion des eigentlichen elektronischen Geräts, der sogenannten *Hardware*, als auch die Herstellung der *Software*, also der immateriellen Programme zur Steuerung der Hardware. Schließlich unterscheiden sich auch die Anwendungsgebiete des Großcomputers und des Personal Computers ganz erheblich.

Trotz der skizzierten historischen Abfolge – zunächst Dominanz der Großcomputer bis etwa 1975, danach Aufkommen der Personal Computer – haben sich beide Stränge der Computerentwicklung seit den Anfängen des elektronischen Digitalcomputers um 1940 stets parallel entwickelt. Auch heute noch werden für bestimmte Aufgaben Großcomputer verwendet, wirtschaftlich sind sie weiterhin von erheblicher Bedeutung. Während diese Tatsache noch weithin bekannt ist, finden sich in der bisherigen Computergeschichtsschreibung nur wenige Hinweise auf die Vorgeschichte des Personal Computers, seine geistigen und technischen Wurzeln.⁴

Im Gegensatz zur traditionellen Historiographie, die vor allem die Geräteentwicklung und die Entstehung der Computerindustrie behandelt, bietet die Vorgeschichte des Personal Computers eine besondere Gelegenheit, die Verflechtung

¹ Die Einschätzung spiegelt sich sowohl in der Berichterstattung der Presse wieder, die den Begriff des »Elektronengehirns« prägte, als auch in der zeitgenössischen (Science Fiction) Literatur. Vgl. Martin 1993; Mowshowitz 1977.

² Norberg 1984, S. 198; Phister et al. 1983, S. 337ff.

³ Campbell-Kelly and Aspray 1996, S. 207; Bell and Gray 1997, S. 21.

⁴ Eine positive Ausnahme stellen die empfehlenswerten Bücher von Campbell-Kelly und Aspray (1996) bzw. Ceruzzi (1998) dar, die diesem Thema mehr Aufmerksamkeit widmen.

mit anderen Wissenschafts- und Technikbereichen aufzuzeigen. Diese vielfach übersehene Interdisziplinarität war eine wichtige Voraussetzung für die Entwicklung des Computers zu einem einfach zu bedienenden Werkzeug und Medium für jedermann und weist eindringlich auf die zentrale Rolle des Computers für die Technikentwicklung nach 1945 hin.

Die Darstellung der Vorgeschichte des interaktiven persönlichen Computers orientiert sich folglich an drei zentralen Fragestellungen:

1. Vor welchem kulturellen Hintergrund ist die Idee der persönlichen Informationsverarbeitungsmaschine entstanden, wie hat sie sich im Laufe der Zeit unter dem Einfluß anderer Wissenschafts- und Technikentwicklungen verändert?
2. In welchen Entwicklungsschritten wurde der Digitalcomputer bei der Umsetzung dieser Ideen von einer reinen Rechenmaschine zu einem universellen Medium für Informations- und Kommunikationsaufgaben umgestaltet?
3. Wo gibt es Berührungspunkte zwischen den beiden Linien der Computerentwicklung, und welches sind die eigenständigen Problemstellungen, Lösungsansätze und Entwicklungsmethoden bei der Entwicklung interaktiver persönlicher Computer?

Dabei soll auch gezeigt werden, daß sich die eigentliche Computerrevolution – wenn der Begriff der Revolution überhaupt verwendet werden sollte – nicht im Bereich der Hardware abgespielt hat, wie viele bisherige Untersuchungen suggerieren. Es sind vielmehr die in den Programmen realisierten Ideen und Vorstellungen, die den Computer zu einer besonderen Maschine gemacht haben. Da Soft- und Hardware nicht sinnvoll getrennt voneinander betrachtet werden können, wird auch deren Wechselwirkungen zu analysieren sein. Schließlich haben auch die Zeitläufe ihre Spuren hinterlassen: Das beginnende Wettrüsten nach Ende des Zweiten Weltkrieges, die verstärkte amerikanische Technologieförderung während des Vietnamkonflikts und das Aufkommen der sogenannten Gegenkultur in den späten sechziger Jahren, sie alle haben die Computerentwicklung beeinflusst und sind damit mittelbar ein Teil der uns heute umgebenden Technik. Den Charakter dieser Entwicklung hat Pierre Lévy pointiert so formuliert: Der Computer bilde »den Schlußpunkt einer zufälligen Folge lokaler Gegebenheiten und Umstände, die sich zahlreiche Akteure – so gut es eben ging – zunutze machten.«¹

¹ Lévy 1995, S. 904.

1.2 Stand der Forschung

Die Computergeschichtsschreibung hat sich bislang nur wenig mit dem skizzierten Themenbereich befaßt. Tatsächlich beginnt sich die Computergeschichte erst seit wenigen Jahren methodisch innerhalb der Wissenschafts- und Technikgeschichte zu verorten.¹ Obwohl sich die Technikgeschichte mittlerweile auch der jüngeren und jüngsten Computergeschichte angenommen hat, ist der Bereich der Softwareentwicklung und der Wechselwirkung zwischen Software- und Hardwareentwicklung in der Forschung kaum präsent.² Zur Entwicklung des *Personal Computing* sind neben den Beiträgen zur Tagung über die Geschichte der Personal Workstations³ vor allem die Bücher »Tools for Thought« (1985) von Howard Rheingold sowie »Hackers« (1984) und »Insanely Great« (1995) von Steven Levy zu nennen. Sie befassen sich mit der veränderten Beziehung zwischen dem Computer und seinem Anwender seit Ende der fünfziger Jahre zunächst am Massachusetts Institute of Technology (MIT) und seit Anfang der sechziger Jahre im Umfeld der Stanford University. Dabei deutet bereits der Untertitel von Rheingolds Buch (»The People and Ideas behind the *Next* Computer Revolution«) an, daß es dem Autor – wie in seinen anderen Büchern⁴ – vor allem darum geht, die Verheißungen des technischen Fortschritts zu verkünden.⁵ Steven Levy hat zwar ein ausgeprägteres historisches Interesse als Rheingold, erzählt aber in seinen beiden solide recherchierten Büchern recht unreflektierte Heldengeschichten. In »Hackers« befaßt er sich mit Gruppen junger idealistischer Computerenthusiasten am MIT und im Großraum San Francisco, die der Softwareentwicklung seit 1960 manchen Impuls gegeben haben und für Levy die eigentlichen »Helden der Computerrevolution« sind.⁶ Levys Buch »Insanely Great« kommt der hier verfolgten Argumentation am nächsten, setzt aber andere Schwerpunkte, indem

¹ Norberg 1984; Mahoney 1988; Aspray 1994.

² Mahoney 1990, S. 325ff.; Cortada 1990, S. 396. Seit dem Erscheinen von Herman Goldstines Monographie im Jahre 1972 ist eine ganze Reihe von Studien zu Teilbereichen der Computergeschichte veröffentlicht worden. Sie befassen sich aber überwiegend mit der Hardwareentwicklung bis etwa 1960, der Entstehung der Computerindustrie oder mit der Rolle des Computers als mathematischem Instrument.

Mit der Entwicklung der Computertechnik beschäftigen sich beispielsweise Beauclair (1968), Goldstine (1980), Williams (1986) und Croarken (1990). Gute Darstellungen zur Wirtschaftsgeschichte des Computers stammen von Stern (1982), Flamm (1988), Petzold (1992) und Campbell-Kelly (1989). Zur Geschichte des Computers als mathematischem Instrument wird auf Nash (1990) und Heintz (1993) verwiesen. Weitere Hinweise auf Literatur zur Computergeschichte finden sich in den umfangreichen Bibliographien von Randell (1979) und Cortada (1990, 1996a, 1996b).

³ Goldberg 1988.

⁴ Rheingold 1994; Rheingold 1995.

⁵ Rheingold 1985, S. 13.

⁶ Levy 1984, S. 7f.

versucht wird, den Apple Macintosh als zwangsläufigen Endpunkt aller Computerentwicklungen nach 1945 darzustellen.¹ Sowohl Levy als auch Rheingold arbeiten als Journalisten, so daß bei ihren populärwissenschaftlich geschriebenen Büchern trotz fachkundiger Recherche viele Details und Zusammenhänge zu Gunsten eines pointierten Stils auf der Strecke bleiben.²

Ansonsten dominieren (immer noch) ausgeprägt technik- oder unternehmenszentrierte Darstellungen das Feld. Dabei betont die technisch orientierte Geschichtsschreibung des Personal Computers die Fortschritte in der Mikroelektronik und die dadurch ermöglichten Fortschritte beim Bau kleiner Computer. Solche Darstellungen stellen meist die Erfindung des Mikroprozessors durch die Intel Corporation im Jahre 1971 an den Beginn der Entwicklung, um dann die raschen Fortschritte bei der Leistung und Speicherausstattung der mikroprozessorbasierten Computer aufzulisten, an deren Ende in der Regel die heutigen Hochleistungs-PCs stehen.³ Solche Darstellungen lassen freilich außer acht, daß die Entwickler und Nutzer der ersten Mikroprozessoren Erfahrungen bei der Konstruktion und/oder Programmierung von Groß- oder Minicomputern besaßen, die in die ersten Computer auf Mikroprozessorbasis einfließen. Andere Autoren sehen hingegen die Subkultur der sogenannten Hacker mit ihren sozialutopischen Ideen als den Ausgangspunkt für die Entwicklung und Nutzung des Personal Computers.⁴ Beide Sichtweisen übersehen in der Regel die Kontinuität, die man beim Übergang vom gemeinsam genutzten Großcomputer zum einzeln verwendeten Personal Computer feststellen kann.

Auf diese Weise wird die Entstehung des Mikrocomputers entweder als Übergang einer technischen Innovation in ein Produkt oder aber als Resultat einer utopischen Vision betrachtet, deren Ziel die Demokratisierung der Technik war. Beide Sichtweisen führen allerdings zu Argumenten, die den Mikrocomputer als revolutionär ausweisen, sei es auf technologischer Seite oder in bezug auf die Diffusion der Computernutzung und das Versprechen eines Computers für den »Rest von uns«. Sie wiederholen damit lediglich die Ideologie zweier wichtiger Gruppen, die an der Entstehung des Personal Computers beteiligt waren; deshalb kann auch keine der beiden Sichtweisen eine angemessene Rekonstruktion der Entwicklungsgeschichte und der Interaktion zwischen den beteiligten Akteuren leisten.

Erst seit kurzem setzt sich die Technikgeschichtsschreibung auch mit den politischen, kulturellen, gesellschaftlichen Einflüssen bei der Entwicklung von

¹ Levy 1995.

² Raskin 1994a.

³ Zum Beispiel Noyce and Hoff 1981; Bischoff 1983; McMullen and McMullen 1993; Tredennick 1996 oder Miller 1997.

⁴ Levy 1984; Nelson 1987; Pfaffenberger 1988; Raskin 1994c.

1. Einleitung

Computern und deren Nutzung auseinander. Neben den Überblickswerken von Campbell-Kelly und Aspray (1996) bzw. Ceruzzi (1998) sind hier vor allem die Bücher von Edwards (1995) und Norberg und O’Neill (1996) zu nennen. Edwards erklärtes Ziel ist, anhand der Computergeschichte zu zeigen, wie Ideen und technische Artefakte durch Politik und Kultur miteinander verbunden sind, und er konzentriert sich dabei auf die parallele Entwicklung von Kognitions- und Computerwissenschaft während der fünfziger und sechziger Jahre. Während Edwards den Blick von außen auf die Computerentwicklung wirft, haben Norberg und O’Neill in ihrem Buch die Mechanismen der amerikanischen Forschungsförderung durch die *Advanced Research Projects Agency*, eine Abteilung des amerikanischen Verteidigungsministeriums, analysiert und dabei festgestellt, daß ein Großteil der heutigen Computertechnologie direkt oder indirekt auf solche militärisch finanzierten Entwicklungsprojekte zurückgeht.¹

1.3 Akteure und Leitbilder

Da es sich um ein völlig neues Gebiet von Wissenschaft und Technik handelte, wurde die Entwicklung der ersten Digitalcomputer während der vierziger Jahre von einer kleinen Gruppe von Mathematikern und Ingenieuren vorangetrieben, die sich – zumindest im angloamerikanischen Raum – persönlich kannten. Ihr Wissens- und Erfahrungsaustausch erfolgte in der Regel auf informelle Weise, durch persönliche Kontakte, Sommerschulen und in kleinen Auflagen vervielfältigte Forschungsberichte.² Auch während der fünfziger Jahre hatte man es, trotz der Anfänge einer Computerindustrie, immer noch mit einem überschaubaren Kreis von Akteuren zu tun, die an wenigen renommierten Institutionen arbeiteten.³ Erst mit dem wirtschaftlichen Erfolg der Computerbranche und der Anerkennung der *Computer Science* bzw. Informatik als eigenständige wissenschaftliche Disziplin entwickelte sich ein formalisierter, weitgehend anonymer

¹ Edwards 1998. In näherer Zukunft werden schließlich drei weitere Studien fertiggestellt, die sich auch inhaltlich mit dem gleichen Themenkreis wie dieses Buch befassen. Der kanadische Technikhistoriker Thierry Bardini hat soeben sein Buch über Douglas C. Engelbart fertiggestellt, der Pulitzer-Preisträger Michael Hiltzik schreibt momentan an einer Geschichte des Xerox Palo Alto Research Center und Ted Friedman arbeitet an der Duke University an einer Dissertation über die Kulturgeschichte des Personal Computers.

² Wolfe 1992; Flamm 1988, S. 217ff.

³ Wichtige Zentren der frühen Computerentwicklung in der USA waren die University of Pennsylvania in Philadelphia, das Institute of Advanced Study (IAS) in Princeton, das Massachusetts Institute of Technology in Cambridge, Mass. sowie das amerikanische Bureau of Standards. In England wurden Computer an den Universitäten in Manchester und Cambridge sowie beim National Physical Laboratory in Teddington entwickelt.

Forschungsbetrieb, in dem der einzelne keine detaillierte Übersicht über die Gesamtheit der Forschungsaktivität mehr besitzen konnte.¹

Trotz dieser Tendenz blieben die wenigen seit Beginn der Entwicklung beteiligten Institutionen auch nach 1960 führend in der Computerforschung.² Diese Entwicklung wurde auch durch die Forschungsförderung des US-Verteidigungsministeriums begünstigt, das statt einer flächendeckenden Unterstützung der Computerforschung sogenannte *Centers of Excellence* unterstützte, die in der Regel an den renommierten privaten Eliteuniversitäten angesiedelt waren. Ihnen gelang es wegen ihrer guten finanziellen Ausstattung weiterhin, die besten und kreativsten jungen Wissenschaftler zu verpflichten. Innerhalb dieser Gruppe spielten persönliche Kontakte nach wie vor eine bedeutende Rolle, wenn es um den Austausch von Wissen und Erfahrung, aber auch um die Besetzung von Stellen ging.

Da die in diesem Buch geschilderten Entwicklungen vorwiegend innerhalb solcher militärisch geförderten Projekte stattgefunden haben, scheint es sinnvoll, die Akteure in das Zentrum der Untersuchung zu stellen, zumal die französische Technikoziologie in den vergangenen Jahren ein umfangreiches Instrumentarium zur Analyse von Akteur-Netzwerken bereitgestellt hat.³

Dieser Ansatz geht davon aus, daß sich die Entstehung von Technik als ein Prozeß von Netzwerkbildungen vollzieht. Dabei besteht das Netzwerk aus heterogenen Elementen, die in Beziehung zueinander treten. Personen und Institutionen sind aus dieser Perspektive ebenso Akteure wie die technische Artefakte oder nichtmaterielle, abstrakte Ideen. In Analogie zu biologischen Vorgängen wird davon ausgegangen, daß sich die Beziehungen zwischen den Akteuren durch ständig stattfindende Transformations- und Übersetzungsprozesse kontinuierlich verändern. Stabilität – die sogenannte Punktualisierung des Netzwerks – ist demnach immer nur ein vorübergehender Zustand.⁴

¹ Ceruzzi 1989.

² William Aspray (1998) hat allerdings darauf hingewiesen, daß gerade das IAS bzw. die Princeton University und die University of Pennsylvania ihre Spitzenposition nicht wahren konnten.

³ Dieser Forschungsansatz wurde zuerst von Michel Callon Anfang der achtziger Jahre am Centre de Sociologie de l'Innovation in Paris entwickelt und wird heute – neben Callon (1986a, 1987, 1991) – insbesondere von John Law (1992) weitergeführt. In der Bundesrepublik wurde die Theorie der Akteur-Netze bisher noch kaum rezipiert. Vgl. Heimer 1993; Joerges 1995; Schmidt and Werlé 1998.

Die Theorie der Akteur-Netzwerke steht auch in enger inhaltlicher und methodischer Nähe zur Theorie großer technischer Systeme, die von T. P. Hughes in seinem Buch »Networks of Power« (1983) beispielhaft formuliert wurde und heute als wertvolle Methode zur Beschreibung der technischen Entwicklung anerkannt ist. Vgl. auch Hughes 1987; Joerges 1988.

⁴ Callon 1991; Law 1992, S. 386 – Law schreibt dort: »(The theory explains) how actors and organizations mobilize, juxtapose, and hold together the bits and pieces out of which they are composed; how they are sometimes able to prevent those bits and pieces from following their own inclinations and making off; and how they manage, as a result, to conceal for a time the process of transla-

1. Einleitung

Für die Analyse der (Inter-)Aktionen während der Entwicklung interaktiver persönlicher Computer sollen vor allem folgende Konzepte der Akteur-Netzwerk-Theorie genutzt werden:

- Die Rekrutierung von assoziierungsfähigen Elementen in ein Akteur-Netzwerk (*enrolment*)¹,
- der Verhandlungsrahmen (*negotiation space*), innerhalb dessen Akteure ihre Projekte verstetigen,
- die Taktiken der Übersetzung in Handlungsprogramme (*translation*)², die sie dabei anwenden,
- sowie die obligatorischen Kontrollstellen bzw. Verbindungsglieder (*obligatory point of passage*) im Kommunikationsfluß.³

Mit Hilfe solcher Konzepte soll beschreibbar gemacht werden, wie sich in einem wenig formalisierten und institutionalisierten Umfeld bestimmte Forschungs- und Entwicklungsströmungen ausbilden konnten.

Nun sind Computer keine gewöhnlichen technische Artefakte, weil sie neben materiellen Komponenten in zunehmendem Maße auch immaterielle Bestandteil in Form von Software umfassen. Erst die Software macht aus dem elektronischen Gerät »Computer« eine Universalmaschine. Jeder Sachverhalt, der sich in Form eines Algorithmus formulieren läßt, ist auch einer computergestützten Bearbeitung zugänglich. Da auf diese Weise der Verwendung des Computers grundsätzlich kaum Grenzen gesetzt sind, ist es besonders wichtig, welche Vorstellungen die Entwickler über die Eigenschaften des Computers, über mögliche Einsatzgebiete und Beschränkungen sowie über die Rolle des Computers in unterschiedlichen gesellschaftlichen Bereichen besessen haben. Nur durch die Software kann sich der Computer wie eine Rechen- oder Schreibmaschine verhalten oder wie im Fall der sogenannten »Virtuellen Realität« völlig artifizielle Welten simulieren, in denen die Gesetze der realen Welt keine Gültigkeit haben müssen. Deshalb kann man die Softwareentwicklung mit einigem Recht als eine Form von Realitätskonstruktion betrachten, deren Endprodukt materialisierte Ideen sind.⁴

Beispielsweise haben Entwickler einer neuen Computerarchitektur, d. h. dessen logischer Funktionsweise, bestimmte Vorstellungen von der Arbeitsweise der neuen Maschine. Diese Vorstellung verändert sich, sobald der abstrakte Entwurf mit konkreten Bauelementen unter bestimmten finanziellen und technologischen

tion itself and so turn the network from a heterogenous set of bits and pieces each with its own inclinations, into something that passes as a punctualized actor«.

¹ Law 1983.

² Callon 1991, S. 142ff.; Law 1992, S. 386ff.

³ Callon 1986b, S. 205f.

⁴ Floyd 1989; Hörning 1989, S. 117f.; Snelting 1998.

Randbedingungen in die Realität umgesetzt wird. Gleichzeitig hat der Entwickler eine bestimmte Art der Nutzung im Auge, die sich ganz erheblich von der tatsächlichen späteren Verwendung des Computers unterscheiden kann. Solche Veränderung stehen im Zentrum der folgenden Ausführungen.

Für die Rolle von Ideen, Metaphern und Analogien wurde in den vergangenen Jahren von der sozialwissenschaftlichen Technikforschung der Begriff des Leitbildes entwickelt.¹ Dabei konnten Dierkes et al. (1992) zeigen, daß es in Wissenschaft und Technik tatsächlich einen Mechanismus des »Leitens« durch »Bilder« zu geben scheint. Leitbilder bündeln demnach die Intentionen, das Wissen und die Erfahrung der Menschen darüber, was ihnen einerseits als machbar und andererseits als wünschenswert erscheint. Darüber hinaus können sie die individuelle Wahrnehmung und das Wertesystem der an der Produktion von technischem Wissen beteiligten Akteure in eine gemeinsame Richtung lenken. Schließlich vereinfacht ein gemeinsames Leitbild gerade in einem noch nicht ausgereiften Wissenschafts- und Technikbereich die Kommunikation zwischen den Beteiligten. Inwieweit ein Leitbild wirklich eine leitende Funktion wahrnehmen kann, hängt allerdings von der Attraktivität und Stärke der verwendeten Bilder, Metaphern und Analogien ab.²

Bereits Leibniz notierte 1705, alles menschliche Denken vollziehe sich mittels gewisser Zeichen und Charaktere.³ Bei diesen Zeichen und Symbolen kann es sich um unterschiedliche Repräsentationsformen handeln, um eine begriffliche oder um eine bildliche Repräsentation. Vieles spricht dafür, daß manche Probleme gedanklich leichter in einer begrifflichen, andere wiederum leichter in einer bildlichen Repräsentationsform zu bewältigen sind. Die Originalität des Menschen – und eines erfolgreichen Leitbildes – besteht hauptsächlich darin, für jeden Wissensbereich und jedes Problem die adäquate, d. h. am leichtesten zu durchschauende Repräsentationsform zu finden. Für die Repräsentation neuen technischen Wissens, für die noch keine eigenen sprachlichen Begriffe existieren, werden deshalb häufig Metaphern oder Analogien verwendet.⁴

Ein Idee muß aber nach Dierkes et al. (1992) auch eine genügend große Zahl von Anhängern besitzen und bestimmten Erfordernissen gerecht werden, um als

¹ Die Leitbildforschung stellt ein zentrales, aber auch umstrittenes Thema der deutschen Technikgeneseforschung dar. Sieht man vom Anspruch der Techniksoziologie ab, damit ein angemessenes Instrumentarium zur »weichen« Techniksteuerung entwickelt zu haben, so wird ihm doch ein erhebliches Potential als Analyseinstrument zugesprochen. Vgl. Hellige 1993, König 1993 und Hellige 1996b.

² Dierkes et al. 1992, S. 41–52. Insbesondere der letzte Aspekt wurde bereits von Thomas Kuhn als wichtiges Element bei der Entwicklung einer »normalen Wissenschaft« herausgestellt. Vgl. Kuhn 1992, S. 392f., 398.

³ Leibniz 1996, S. 110.

⁴ Dierkes et al. 1992, S. 52ff.; Mambrey und Tepper 1992; Busch 1995.

1. Einleitung

Leitbild bezeichnet werden zu können. Es muß evident sein (wie selbstverständlich immer vorhanden sein), es muß in verschiedenen Wissenskulturen zu Hause sein und die Kulturen damit überbrücken, es muß kreative Impulse auslösen, muß richtungweisend, stabilisierend und auch korrigierend sein.¹

Solche Leitbilder oder Orientierungsmuster, wie Heike Stach und Ralf Bohnsack sie genannt haben, können eine lang-, mittel- oder kurzfristigen Wirkungsdauer besitzen. Langfristige Orientierungsmuster beinhalten Gewohnheiten, Wissensstrukturen und Wertsysteme, die über lange Zeiträume in der Gesellschaft geteilt und durch Erziehung und Ausbildung in einem bestimmten kulturellen Umfeld vermittelt werden. Wegen ihrer Allgegenwart werden sie von den Beteiligten einer Wissenschaft- und Technikentwicklung nur selten explizit thematisiert. Dennoch ist das tief verwurzelte Verständnis von Wissenschaftlern und Technikern über die eigene gesellschaftliche Rolle von erheblichem Einfluß für ihre Ziele und Methoden. Mittelfristige Orientierungsmuster entsprechen am ehesten der geschilderten Form des Leitbildes mit Verwendung bestimmter Bilder, begrifflich faßbarer Ideen und Normen, die von den beteiligten Personen reflektiert und oft mit sehr viel Engagement propagiert werden.² Kurzfristige Orientierungen sind hingegen oft das Ergebnis eines mehr oder weniger zufälligen Aufeinandertreffens von Akteuren, durch die neue Ideen entstehen, aber auch zeitlichen und finanziellen Drucks oder der generellen Arbeitsatmosphäre. Insbesondere mittelfristige Leitbilder lassen sich besonders gut durch narrative Interviews und die Analyse von im Rahmen der Technikentwicklung entstandenen programmatischen Texten erschließen.³

Im Bereich der Informatik versteht man unter einem Leitbild häufig ein Verständnismodell einer Computeranwendung oder eines Computersystems, an dem sich die Gestalter bei ihrer Entwicklungsarbeit orientieren. Dabei handelt es sich im wesentlichen um die metaphorische Übertragung von Bedeutungen und ihre technische Nachbildung. Systeme werden vielfach in Analogie zu bekannten und vertrauten Kategorien gestaltet. Diese Analogie kann struktureller Natur sein wie bei der elektronischen Post. Sie sich aber auch an qualitativen Aspekten orientieren wie im Falle der sogenannten Künstlichen Intelligenz.

Bei der Frage nach grundsätzlichen Eigenschaften des Computers und der daraus resultierenden Verwendungsmöglichkeiten kann man drei Leitbilder identifizieren, die keine zeitliche Abfolge darstellen, wie dies z. B. bei einer Periodisierung nach Computergenerationen suggeriert wird. Es handelt sich dabei um die Sichtweisen des Computers als *Automat*, als *Werkzeug* und als *Medium*. Tat-

¹ Dierkes et al. 1992, S. 39ff.; Tepper 1996, S. 150f.

² Stach 1996, S. 51ff.; Bohnsack 1998; Bijker et al. 1987, S. 5.

³ Stach 1996, S. 51.

sächlich ist jedes dieser Leitbilder mit einer bestimmte Gruppe von Akteuren, bestimmten Anwendungen und Systemarchitekturen verbunden.

1.4 Vom Automaten zum Werkzeug und Medium

Was und wie

Dies ist keine Geschichte der *Computertechnik*, auch keine Geschichte der *Computeranwendungen*, sondern in erster Linie eine Geschichte der *Computernutzung*. In ihrem Zentrum steht die Frage, *was* man mit einem Computer alles tun kann und *wie* der Mensch dabei mit dem Computer umgeht. Beide Aspekte unterliegen der Programmierbarkeit und betonen den »weichen« Charakter des Computers.¹

In einem ersten Kapitel wird zunächst eine erste einflußreiche Vorstellung davon präsentiert, *was* ein Wissenschaftler mit einem Computer machen und *wie* er mit der Maschine in Interaktion treten soll. Diese Vorstellung stammt von Vannevar Bush, einem der wichtigsten Computerpioniere in der Zeit vor 1940 und gleichzeitig einem der einflußreichsten Organisatoren von Wissenschaft und Technik seiner Zeit. An seinem Beispiel wird gleichzeitig verdeutlicht, in welcher Weise die Vorstellungen einer Maschine zur Verarbeitung von *Wissen* ebenso durch das (amerikanische) Selbstverständnis der Ingenieure geprägt wird, wie durch die Möglichkeiten des technischen Fortschritts und die Situation von Wissenschaft und Technik während des Zweiten Weltkrieges. Am Ende steht eine Vision, die in den nachfolgenden Jahren direkt oder indirekt eine Vielzahl von Ingenieuren beeinflußt hat, die mit der Entwicklung von Digitalcomputern beschäftigt waren.

Die ersten, zwischen 1945 und 1950 gebauten, Computer waren riesige elektronische Maschinen, die zusammen mit der zugehörigen Peripherie, den Ein- und Ausgabegeräten, ganze Räume ausfüllten. Von ihrem Charakter her handelte es sich dabei um Rechenmaschinen, die nach einem festgelegten Schema umfangreiche numerische Operationen durchführten.² In dieser Zeit waren die Benutzer und die Bediener des Computers noch identisch, vielfach waren sie gleichzeitig auch Programmierer. Sie befanden sich im gleichen Raum mit dem Computer, richteten Schalt- und Stecktafeln ein, legten Magnetbänder ein, bedienten die Schalter auf der Bedientafel des Computers und kontrollierten mit Meßgeräten den Zustand

¹ Coy 1992; Nake 1994, S. 314.

² Aus dieser Entwicklung stammt auch die Bezeichnung »Computer«. Vgl. Ceruzzi 1991. Auch in Deutschland, wo der adäquatere Begriff »Datenverarbeitungsanlage« lange Zeit vorherrschend war, wird heute weitgehend vom Computer gesprochen.

1. Einleitung

der Hardware. Sie richteten die Maschine für ihre Berechnungen ein und setzten sie in Gang. Sie waren mit der Maschine vertraut und wußten das Flackern der Anzeigen und die Geräusche der Maschine zu deuten. Für sie war der Computer nichts als eine Maschine, mit deren Hilfe sich aufwendige und fehleranfällige Berechnungen automatisieren ließen. Dabei konnte und mußte der Benutzer die Maschine bis ins letzte Detail kontrollieren. Der Grad an Interaktion zwischen Mensch und Computer war in dieser Phase sehr hoch, die Art der Interaktion so unmittelbar, wie sie nur sein kann. Dabei sprachen die Benutzenden allerdings nicht von Interaktion, dieser Begriff kommt erst im historischen Rückblick in den Sinn.

Menschen und Systeme

Parallel zur Entwicklung der ersten Digitalcomputer kam die Erkenntnis auf, daß der Mensch in komplexen technischen Systemen eine entscheidende Rolle spielt. Aus dem Wunsch, soziotechnische Systeme in Hinblick auf Beherrschbarkeit, Sicherheit und Wirtschaftlichkeit optimal zu gestalten, entstand Mitte der vierziger Jahre die multidisziplinäre Disziplin der Ergonomie, deren Ziel es war, physiologisches, medizinisches und psychologisches Grundlagenwissen in den Entwurf komplexer technischer Systeme einfließen zu lassen.¹ Neben der Gestaltung von Flugzeugcockpits und Leitwarten, spielte auch die Gestaltung von (militärischen) Computersystemen von Beginn an eine entscheidende Rolle.² Dabei wurde das bisher vorherrschende lineare Modell der Datenverarbeitung zu einem geschlossenen Regelkreis erweitert, der den Menschen nicht nur als Anhängsel der Maschine, sondern als gleichwertigen Bestandteil des soziotechnischen Systems anerkannte.

Mit dem Übergang zu dieser systemtheoretischen Sichtweise wurde man sich auch der Schnittstelle (engl. *interface*) zwischen Mensch und Maschine bewußt. Da soziotechnische Systeme konstruiert sind, wurde auch die Schnittstelle zwischen Mensch und Maschine zum Gegenstand einer bewußten, wissenschaftlich fundierten Gestaltung. Notwendigerweise muß die Rolle des Menschen für den Systementwurf auf wenige Funktionen reduziert werden. Implizit oder explizit wird das Modell, das sich der Entwickler vom potentiellen Benutzer macht, zum Schlüsselement für die Eigenschaften des von ihm konstruierten Systems.³

Als Beispiel für diese Entwicklungen wird in Kapitel 3 die Entwicklung von Computern und Computersystemen am Massachusetts Institute of Technology (MIT) während der vierziger und fünfziger Jahre dargestellt, deren zentraler

¹ Die Etablierung der neuen Disziplin läßt sich mit dem Erscheinen des ersten Lehrbuchs von Chanis et al. 1949 datieren.

² Christensen 1976, S. 288f.; Sheridan 1986, S. 1f.

³ Ropohl 1979; Nake 1984, S. 113.

Akteur Jay W. Forrester ist, der zunächst in der von Vannevar Bush verkörperten Tradition des Ingenieurberufs stand. Gleichzeitig war er auch mit der Regelungstechnik bzw. Kybernetik in Kontakt gekommen, die während der dreißiger und vierziger Jahre von Gordon Brown und Norbert Wiener am MIT als neue Denkrichtung etabliert wurde und die *systemische* und *ganzheitliche* Betrachtung von biologischen, sozialen und technischen Phänomenen betonte. Bei der Entwicklung des Whirlwind-Computers durch Forrester und seine Mitarbeiter fand die kybernetische Denkweise ihren Niederschlag sowohl in der verwendeten Entwurfssystematik als auch bei den Überlegungen, den Computer für Steuerungsaufgaben statt nur für Berechnungen einzusetzen. Nicht zuletzt bedeutete die Rolle, die dem Menschen innerhalb eines solchen Systems zugedacht ist, einen erheblichen Unterschied zu anderen Computerentwicklungsprojekten der damaligen Zeit.

Aus diesem Projekt ging in den fünfziger Jahren das erste computergestützte Luftverteidigungssystem SAGE hervor, für das erstmals eine komplexe Mensch-Computer-Schnittstelle entwickelt wurde, weil die typischen Bediener von SAGE keine Computerspezialisten, sondern einfache Soldaten waren. Zu diesem Zweck wurden Verfahren zur Ein- und Ausgabe von Daten in unmittelbarer Interaktion mit dem Computer entwickelt, die den hohen Anforderungen eines Luftverteidigungssystems entsprachen. Dabei entstanden Bildschirmsysteme für Texte und Grafiken und der Lichtgriffel als Eingabegerät für grafische Daten.

Mensch, Computer, Kommunikation

Die Zahl der Benutzer wurde während der fünfziger Jahre allmählich größer, und diese kannten sich nicht mehr mit den Details der Maschine aus. Deshalb wurden sie nur noch als Beobachter und Experten ihrer eigenen Anwendung an die Maschine gelassen. Die Funktionen von Programmierer, Benutzer und Bediener des Computers traten auseinander. Insbesondere bei den beiden ersten Gruppen war dies mit einer zunehmend abstrakteren Vorstellung des Computers verbunden. Für sie bestand der Computer nicht mehr in erster Linie aus elektronischen Schaltkreisen, Registern und Speicherzellen, sondern war ein Gerät mit einem bestimmten Funktionsumfang, für das sie Programme zur Bearbeitung bestimmter Aufgaben schrieben. Solche Programme wurden in Form von Lochkartenstapeln gespeichert und wanderten zwischen dem Arbeitsplatz des Benutzenden und dem Computerzentrum hin und her. Die Aufträge der Benutzer wurden im Zentrum von Maschinenbedienern nacheinander in die Maschine zur Verarbeitung eingegeben. Die Maschine und ihre Bediener waren damit relativ autonom von den Benutzern und Programmierern.¹

¹ Ceruzzi 1994; Nake 1995, S. 35.

1. Einleitung

Kapitel 4 zeigt, wie aus Unzufriedenheit mit dieser Trennung seit Beginn der sechziger Jahre der Versuch unternommen wurde, den Benutzer wieder in engeren Kontakt mit dem Computer zu bringen. Joseph C. R. Licklider, Professor für experimentelle Psychologie am MIT und während der fünfziger Jahre Mitarbeiter bei der Entwicklung des SAGE-Systems, war der Mann, dessen Vision einer interaktiven Computernutzung während der sechziger Jahre Hunderte von gleichgesinnten Wissenschaftlern motivierte, sich mit neuen Wegen bei der Hard- und Softwareentwicklung zu beschäftigen. Als Licklider Anfang der sechziger Jahre einen wichtigen Posten innerhalb der militärischen Forschungsförderung übernahm, konnten diese Wissenschaftler mit der Realisierung des als »Time-Sharing« bezeichneten Ansatzes beginnen.

Die kostspieligste Ressource des Computers, der Zentralprozessor, war zu diesem Zeitpunkt so leistungsfähig geworden, daß er immer wieder auf periphere Prozesse, d. h. Ein- und Ausgaben, warten mußte. Deshalb wurde beim Time-Sharing mit dem Betriebssystem eine Ordnungsinstanz geschaffen, die die Hardware des Computers mit dem Ziel einer möglichst hohen Auslastung verwaltete. Mit dem Time-Sharing konnte die Rechenzeit des Computers automatisch auf viele gleichzeitig am Rechner Arbeitende aufgeteilt werden. An den Fernschreibern, die in der Nähe ihrer Arbeitsplätze installiert wurden, wurde den Benutzern näherungsweise der Eindruck vermittelt, die Maschine stünde ihnen allein ständig zur Verfügung. So entstand zu Beginn der sechziger Jahre der Begriff der »Mensch-Computer-Kommunikation«, der bereits auf die Ablösung der Automaten-sichtweise hindeutet.¹

Interaktive Programme, bei denen der Benutzer während der Ausführung Eingaben machen kann, gaben das zusätzliche Gefühl, mit dem Computer zu kommunizieren und partnerschaftlich zusammenzuarbeiten. Obwohl auch bei interaktiven Programmen der Ablauf im voraus festgelegt ist, vermittelten solche Programme ihren Benutzern das scheinbare Gefühl der Kontrolle über die Maschine. Man erkannte bald, daß der Umfang dieser Illusion von der Gestaltung der Kommunikation über die Mensch-Computer-Schnittstelle abhängt. Die facettenreichen und vieldeutigen Möglichkeiten der menschlichen Ausdrucksweise werden bei Eingaben in den Computer auf ihren nackten Datenkern reduziert. Um »Mißverständnisse« zwischen dem Computer und seinem menschlichen Benutzer zu vermeiden, muß der Programmierer eines interaktiven Programms dafür sorgen, daß bei dieser kommunikativen Einengung keine Fehlinterpretationen auftreten können. Letztlich findet die Kommunikation also nicht zwischen dem Benutzer und dem Computer, sondern zwischen dem Benutzer und dem Programmierer eines bestimmten Programms statt. Der Computer übernimmt bei dieser

¹ Friedewald 1998; Hellige 1996a, S. 209ff.

Kommunikation nur die Rolle eines *Mediums*, obwohl bei entsprechend fortgeschrittener Programmierung der Eindruck entsteht, der Computer selbst sei der Partner der Kommunikation.¹ Wie bei vielen anderen Medien droht also das Bewußtsein dafür verlorenzugehen, daß die Kommunikation nur vermittelt ist und daß jedes Medium eigene Ausdrucksformen besitzt. Der kanadische Medientheoretiker Marshall McLuhan brachte dies bereits Mitte der sechziger Jahre in seinem bekannten Slogan »Das Medium ist die Botschaft« zum Ausdruck.²

Computerwerkzeuge

Die interaktive Benutzung eröffnete schließlich völlig neue Anwendungsgebiete, bei denen nicht mehr die Vorstellung des Rechenautomaten, sondern die des Computers als Werkzeug im Vordergrund stand. Vom Computer als *Werkzeug* kann man sprechen, wenn ein Programm so gestaltet ist, daß das Ergebnis seiner Ausführung überwiegend von den Eingaben des Benutzers abhängig ist. Ein solches Programm stellt eine Reihe von Grundfunktionen zur Verfügung, die der Benutzer so kombinieren kann, wie es die Lösung einer bestimmten Aufgabe erfordert. So stellt beispielsweise ein Textverarbeitungsprogramm diejenigen Funktionen zur Verfügung, die man zum Erstellen, Formatieren und Drucken eines Textes benötigt. Inhalt und äußere Form des Textes werden allerdings nicht durch das Programm selbst, sondern durch die Aktionen des Benutzers mit dem Programm festgelegt. Dadurch erhalten die Benutzer des Computers eine relativ hohe Autonomie gegenüber ihrem Arbeitsmittel. Sie behalten stets das bearbeitete Material im Auge, an dem die vorgenommenen Änderungen sofort sichtbar sind. Der große Unterschied zu traditionellen Werkzeugen liegt allerdings darin, daß beim klassischen Werkzeug Benutzung und Funktion eins sind, während der Computer durch die Programmierung in jedes beliebige Werkzeug (und jeden beliebigen Automaten) verwandelt werden kann. Ob der Computer nun ein Automat oder ein Werkzeug ist, ist deshalb weniger eine Frage der verwendeten Hardware, als vielmehr der geeigneten Programmierung.³

Als Beispiel für die Konsequenzen dieser Sichtweise werden in Kapitel 5 die Arbeiten von Douglas C. Engelbart während der sechziger und siebziger Jahre vorgestellt. Engelbart hatte bereits kurz nach dem Zweiten Weltkrieg angefangen, sich Gedanken über ein Gerät zu machen, mit dessen Hilfe man die geistige Arbeit von Wissenschaftlern und Managern effektiver gestalten konnte und sprach von der Verstärkung der menschlichen Intelligenz mit Hilfe des Computers. Zu diesem Zweck untersuchte er – zunächst noch ganz im Sinne eines kybernetischen

¹ Licklider et al. 1968; Nake 1984, S. 115f.; Maaß 1994, S. 331f.

² McLuhan 1992, Kapitel 1, passim.

³ Evans 1969, S. 114ff.; Nake 1994, S. 314f.; Coy 1995, S. 35. – Das Bild des Computers als Automat, Maschine und Werkzeug ist von Budde und Züllinghoven (1990) sowie von Winograd und Flores (1992) in aller Breite behandelt worden.

1. Einleitung

Systemansatzes – seit Anfang der sechziger Jahre die technischen Möglichkeiten für die Gestaltung der Mensch-Computer-Schnittstelle. Gleichzeitig entwickelten Engelbart und seine Mitarbeiter leistungsfähige Werkzeuge für die Dokumentenerstellung und -verwaltung sowie für Aufgaben des Projektmanagements. Im Laufe dieser Arbeiten traten auch die kommunikativen Aspekte der Computertechnik, die Möglichkeit zur zwischenmenschlichen Kooperation und Kommunikation mit Hilfe des Computers zutage. Bei Engelbarts Computersystem wurde auch erstmals die bis heute gültige Form der Computerarbeit praktiziert: Der Benutzer arbeitete an einem Terminal mit einem alphanumerischen oder Grafikmonitor, einer Tastatur und einem grafischen Eingabeinstrument, der Maus, und befand sich dabei in ständiger Interaktion mit dem Computer.

Obwohl Engelbarts fortschrittliches Computersystem bei seiner ersten öffentlichen Vorführung Ende der sechziger Jahre sehr viel Bewunderung auslöste, blieb er damit erfolglos. Die veränderte Forschungsförderung des Pentagon, die Fortschritte bei der Computerhardware und nicht zuletzt Engelbarts überambitionierte Ziele stellten sich letztlich als Hindernis für eine größere Verbreitung seiner Forschungsergebnisse heraus.

Computer als Medium

Während bis Ende der sechziger Jahre lediglich die kommunikativen Aspekte des Computers in Form der Mensch-Computer-Interaktion beachtet wurden, wurde der mediale Charakterzug Ende der sechziger Jahre erheblich erweitert. Diese Entwicklung hat ihre Ursache in der Weiterentwicklung der Speicher- und der Übermittlungsfähigkeit des Computers. In dem Maße, wie die Speicherkapazität vergrößert wurde, trat auch der Automatencharakter des Computers hinter den des Mediums zurück. Ohne die lokale, regionale und globale Vernetzung wäre diese Entwicklung allerdings begrenzt geblieben. Erst durch die Möglichkeit zur gemeinsamen Nutzung von Daten und zur kooperativen Nutzung der miteinander verbundenen Maschinen wurde der Computer bzw. das Netz zu einem neuen digitalen Medium, das traditionelle Medien wie die Presse, Telefon und Fernsehen beerbt.¹

Die Attribute dieses neuen digitalen Metamediums wurden während der siebziger Jahre entscheidend im *Palo Alto Research Center* (PARC) von Xerox herausgearbeitet, das im Zentrum von Kapitel 6 steht. Nach der Einschätzung von Alan Kay, einem der Chefwissenschaftler des PARC, ist der Computer selbst »kein Werkzeug, obwohl er sich wie viele Werkzeuge verhalten kann«. Vielmehr sei er »ein Medium, das alle Einzelheiten anderer Medien dynamisch nachahmen

¹ Bolz 1994; Coy 1995, S. 36; Lang 1996, S. 242ff.

kann, selbst Medien, die in der wirklichen Welt gar nicht möglich sind«. Folglich, so Alan Kay, besitze der Computer Freiheitsgrade der Darstellung und des Ausdrucks, die es bisher noch nie gab.¹

Es zeigt sich, daß sich die Sichtweisen nicht widersprechen müssen. Im Laufe der Entwicklung wurde ein Großteil der Grundfunktionen automatisiert. Die Entwicklung von Programmen gestattet die Verwendung des Computers als Werkzeug, während die Vernetzung den kooperativen Charakter des Mediums hervortreten läßt. Beide Aspekte sind nur möglich, da Computer weiterhin sehr leistungsfähige Automaten für Grundfunktionen wie das Sortieren von und das Suchen in Zeichenketten und das Abbilden und Transformieren von Rasterbildern sind. Während also die (abstrakten) Leitbilder durchaus getrennt voneinander existieren können, greift die reale Umsetzung stets auf ältere technische Entwicklungen zurück, selbst wenn sie aus einer anderen Sichtweise heraus entwickelt wurden.

Mit den Forschungsarbeiten, die zwischen 1970 und 1981 in Xerox' Forschungslabor durchgeführten wurden, wurden beiden Aspekte des Computers, sowohl der des Werkzeugs als auch der des (Ausdrucks-)Mediums, entscheidende Impulse gegeben. Unter der Leitung von Robert Taylor, einem erfahrenen Forschungsorganisator und Talentsucher, versuchte eine Gruppe von jungen Computerwissenschaftlern in der Nachfolge von Douglas Engelbart, die Computerwerkzeuge für das »Büro der Zukunft« zu entwickeln. Dabei entstanden nicht nur der erste Arbeitsplatzcomputer und ein erstes lokales Computernetzwerk, sondern auch eine Vielzahl von Programmen zur Unterstützung der Arbeit von Wissenschaftlern, Ingenieuren und Managern, darunter zukunftsweisende und einflußreiche Textverarbeitungs-, Layout- und Grafikprogramme und neue Peripheriegeräte, wie der erste Laserdrucker. Es wurde auch deutlich, daß das Wissen über das Arbeiten in einem Büro nicht nur informell und intuitiv ist, sondern sich außerdem auch auf eine Vielzahl von Personen verteilt. Weil solches Wissen nur schwer formalisiert und automatisiert werden konnte, legten die Xerox-Wissenschaftler besonderen Wert auf die Anpassung der Computersysteme an individuelle Bedürfnisse und auf die Möglichkeiten der computervermittelten Kommunikation, etwa durch elektronische Post.²

Gleichzeitig arbeitete eine Gruppe unter Leitung von Alan C. Kay an der leistungsfähigen und leicht erlernbaren Programmiersprache Smalltalk, die nicht nur Spezialisten, sondern auch Computerlaien und sogar Kinder zu Programmierern

¹ Vgl. Kay 1984, S. 47 (Zitate); Kay and Goldberg 1977, S. 31. Alan Kay war freilich nicht der erste, der den medialen Charakter des Computers thematisierte. Bereits 1967 schrieb A. Michael Noll, ein Pionier der Computerkunst: »In the computer, man has created not just an inanimate tool, but an intellectual and active partner that, when fully exploited, could be used to produce wholly new art forms and possibly new aesthetic experiences.« Vgl. Noll 1967, S. 89.

² Nake 1984, S. 112.

1. Einleitung

machen sollte. Diese Fähigkeit sollte den Computer, ganz im Sinne von Marshall McLuhan, von dessen Theorien Kay stark beeinflusst war, zu dem bereits erwähnten neuen dynamischen Medium für jedermann machen. Um diesem Personenkreis den Zugang zum Computer noch weiter zu erleichtern, entstand auch die erste graphische Benutzungsoberfläche mit den heute so selbstverständlichen Bestandteilen (Fenster, Icons, Bildschirmmenüs). Nach langwierigen internen Auseinandersetzungen versuchte Xerox Ende der siebziger Jahre, die Ergebnisse beider Entwicklungslinien in einen Arbeitsplatzrechner für Büroanwendungen zu integrieren.

Bis zu diesem Zeitpunkt hatte keines der erwähnten Forschungs- und Entwicklungsprojekte einen über die Grenzen der Wissenschaft hinausreichenden Erfolg gehabt. Das Luftverteidigungssystem SAGE war bereits bei seiner Inbetriebnahme obsolet geworden, Douglas Engelbarts Projekt blutete innerhalb kurzer Zeit nach der öffentlichen Präsentation seines Computersystems personell und finanziell aus und wurde schließlich eingestellt. Und auch Xerox konnte seine fortschrittliche Technologie nicht in ein wirtschaftlich erfolgreiches Produkt umsetzen. Die Vorgeschichte des Personal Computers ist also bis Anfang der achtziger Jahre eine Geschichte von ambitionierten *Forschungsprojekten*, die letztlich alle scheiterten.¹ Kapitel 7 zeigt schließlich, wie der Personal Computer mit dem Auftauchen von jungen, risikobereiten Unternehmen, die an den wirtschaftlichen Erfolg von preiswerten Computern für jedermann glaubten, auch zu einem kommerziell erfolgreichen *Produkt* wurde.

Dabei zeigt sich, daß der Erfolg seine Wurzeln nicht nur in den technischen Eigenschaften des Personal Computers hat, sondern auch in der veränderten Einstellung breiter Bevölkerungsschichten gegenüber dem Computer und den massiven Veränderungen in der Halbleiter- und Computerindustrie seit 1970.

Die hier skizzierte Entwicklung der Mensch-Computer-Interaktion und Computernutzung ist freilich episodisch und bruchstückhaft.² Spätestens seit Mitte der sechziger Jahre ist die Zahl der Forschungs- und Entwicklungsprojekte, der mehr oder weniger innovativen industriellen Produkte zu groß, die Verflechtung zu stark, als daß eine umfassende Darstellung möglich wäre. Die hier getroffene Auswahl an Beispielen zeigt in besonders prägnanter Weise, wie bestimmte

¹ Lee 1994.

² Insbesondere bleibt die Darstellung auf den amerikanischen Kontext beschränkt. Der interaktive Computerbetrieb und die Entwicklung kleiner und kleinster Computer fand in der ersten Hälfte der siebziger Jahre noch nicht den Weg in die Forschung und Entwicklung europäischer Universitäten und Unternehmen. Vereinzelt Ausnahmen wie die Arbeiten von Wolfgang Händler (TU Erlangen) im Bereich der Time-Sharing-Betriebssysteme oder Niklaus Wirths (ETH Zürich) Workstation Lith entstanden meist mit mehreren Jahren Verspätung gegenüber den amerikanischen Entwicklungen und hatten vor allem keine nennenswerte *Rückwirkung* auf die Hauptlinie der amerikanischen Forschung. Vgl. dazu Händler 1968; Wirth 1986b.

Ideen zu Leitbildern der Technikentwicklung wurden, wie sich diese Leitbilder durch die Kommunikation zwischen den Mitgliedern einer überschaubaren Wissenschaftlergruppe verbreiteten und wie sich die Vorstellungen mit der tatsächlich entstehenden Technik, mit den wechselnden Finanzierungsquellen der Forschung und nicht zuletzt mit den veränderten politischen und gesellschaftlichen Bedingungen zwischen 1940 und 1985 weiterentwickelt haben.

Damit soll letztlich versucht werden, die Frage nach den Entstehungsbedingungen der Computertechnik zu beantworten, die Richard Hamming bereits 1976 auf der ersten Tagung zur Computergeschichte aufgeworfen hatte und schon lange zu den »großen« Fragen der Technikgeschichtsschreibung gehört.¹

¹ Daniels 1970; Hamming 1980.

2. Vision und Kontinuität: Vannevar Bushs Memex

The purpose of computing is insight, not numbers.

Richard W. Hamming¹

Das *Personal Computing* ist entgegen der weitverbreiteten Auffassung nicht erst im Zuge der Mikrocomputerentwicklung während der siebziger Jahre entstanden. Die zugrundeliegenden Ideen, die ab 1960 von unterschiedlichen Forschergruppen aufgegriffen und weiterentwickelt wurden, sind sogar älter als der elektronische Digitalcomputer selbst. Sie gehen auf die amerikanische Ingenieurtradition des späten 19. und frühen 20. Jahrhunderts zurück, in der die »Begreifbarkeit« eines technischen Artefakts und der visuelle Entwurf noch eine zentrale Rolle einnahmen.

Der amerikanische Ingenieur und Analogrechnerpionier Vannevar E. Bush (1890–1974, Abb. 1), dessen Arbeiten in den dreißiger und vierziger Jahren als Ausgangspunkt des *Personal Computing* im folgenden näher untersucht werden, war ein typischer Vertreter dieser Tradition. Aber bereits die nur wenig jüngeren ersten Computerpioniere, die ihre Ausbildung nach dem Ersten Weltkrieg absolviert hatten, waren nicht mehr in dieser Tradition verwurzelt. Sie waren die ersten, die sich bei ihrer Tätigkeit in immer stärkerem Maße auf (natur-)wissenschaftliche Methoden und abstrakte Modelle stützten. Aus dieser Konstellation erklären sich die deutlichen Unterschiede, die sich zwischen den Entwürfen für die frühen Digitalcomputer und Bushs Entwurf für eine zukünftige informationsverarbeitende Maschine, der im Zentrum dieses Kapitels stehen soll, feststellen lassen.

2.1 Vannevar Bush: Ingenieur, Manager und Politiker

Bush stammte aus einer alten Familie aus Everett, Mass. und entwickelte schon sehr früh ein ausgeprägtes Interesse für mechanische Basteleien. Als Abschlußarbeit seines Ingenieurstudiums am Tufts College in Medford, Mass., konstruierte

¹ Hamming 1962, S. 3.

2. Vision und Kontinuität: Vannevar Bushs Memex



Abbildung 1. Vannevar Bush um 1945

Bush ein Gerät zur automatischen Aufzeichnung der Geländetopographie, für das er 1914 sein erstes Patent erhielt.¹ Gleichzeitig stellte der sogenannte *Profile Tracer* Bushs ersten Beitrag zur Entwicklung der analogen Rechentechnik dar. Im Jahre 1916 wurde ihm von der Harvard University und dem Massachusetts Institute of Technology (MIT) gemeinsam der Doktorgrad verliehen. Ab 1919 war Bush als Wissenschaftler am MIT beschäftigt, ab 1923 schließlich als Professor für Elektrotechnik.²

Ab 1923 entwickelte sich bei Bush aus den Arbeiten zur Stabilität von über-regionalen Elektrizitätsnetzen ein verstärktes Interesse an einer leistungsfähigen Rechentechnik. Das Ergebnis der Entwicklungsarbeiten, die Bush zusammen mit seinen Schülern Herbert R. Stewart und Harold L. Hazen durchführte, waren 1925 der mechanisch arbeitende sogenannte *Product Integrator* zur Lösung von Differentialgleichungen erster Ordnung und 1927 ein erweiterter *Product Integrator* zur Lösung von Differentialgleichungen zweiter Ordnung.³

Zwischen 1928 und 1931 entstand schließlich unter der Leitung von Bush und Hazen am *Center of Analysis* des MIT der *Differential Analyzer*. Dabei handelte es sich um ein relativ klobiges, aber präzises elektromechanisches Gerät, das sechs Integratoren umfaßte, also sechs Differentialgleichungen erster oder drei

¹ Bush 1912.

² Gleiser 1980, S. 141f.; Wildes and Lindgren 1985, S. 82ff. – Für eine ausführlichere biographische Darstellung siehe Wiesner 1979 und Zachary 1997.

³ Wildes and Lindgren 1985, S. 87ff.; Bush and Hazen 1927; Bush et al. 1927.

Differentialgleichungen zweiter Ordnung handhaben konnte.¹ Dazu hatten Hazen und Bush die Funktionselemente des *Product Integrator* weiter verbessert und neue Elemente hinzugefügt. Die Maschine erwies sich als so nützlich, daß sie von verschiedenen Institutionen nachgebaut wurde.²

Während der dreißiger Jahre arbeiteten Bush und Samuel H. Caldwell an einer Weiterentwicklung des *Differential Analyzer*, die von der Rockefeller-Stiftung finanziell unterstützt wurde. Dabei sollten die mechanischen Bauelemente durch elektronische ersetzt werden; ab 1935 wurde auch an einer Dateneingabe mittels Lochstreifen gearbeitet. Die als *Rockefeller Differential Analyzer* bezeichnete Maschine (Abb. 2) war Mitte 1942 einsatzbereit, wurde aber wegen des Krieges erst nach 1945 publik gemacht. Der *Differential Analyzer* blieb bis zum Ende des Zweiten Weltkrieges die leistungsfähigste im Einsatz befindliche Rechentechnik.³

Zu dieser Zeit begannen die ersten Digitalcomputer bereits machtvoll, die Analogrechner zu verdrängen.⁴ Während das *Center of Analysis* am MIT von der zivilen Rockefeller-Stiftung mit etwa 100 000 \$ für ihre Computerentwicklung gefördert worden war, wurde Jay W. Forresters »Project Whirlwind« am MIT allein in den Jahren 1945/46 von der U.S. Navy mit einem Betrag von 875 000 \$ unterstützt (vgl. Kapitel 3). So ging die große Zeit der Analogrechner am MIT bis etwa 1950 sehr schnell zu Ende.⁵

Bereits im Dezember 1938 hatte Bush das MIT offiziell verlassen, um Präsident der *Carnegie Institution* in Washington zu werden, nachdem er schon am MIT während der dreißiger Jahre administrative Aufgaben übernommen hatte. 1939 wurde er von Präsident Franklin D. Roosevelt auch zum Vorsitzenden des *National Defense Research Committee* (NDRC) und 1941 schließlich zum Direktor des neugegründeten *Office of Scientific Research and Development* (O.S.R.D.) ernannt.⁶ Das O.S.R.D. koordinierte während des Zweiten Weltkrieges sämtliche militärische Forschungs- und Entwicklungsprogramme, darunter auch das Manhattan Project zur Entwicklung der Atombombe. Während dieser Zeit hat sich

¹ Bush 1931.

² Unter anderem waren dies das Ballistics Research Laboratory of the United States Ordnance Department in Aberdeen, MD., die University of Pennsylvania und ein Werk von General Electric in Schenectady. In England wurde der *Differential Analyzer* u. a. von Douglas Hartree in Manchester nachgebaut, in Deutschland an der Technischen Hochschule München.

³ Bush and Caldwell 1945; Wildes and Lindgren 1985, S. 90ff. Zu den von Bush entwickelten Analogrechnern vergleiche Hartree 1949, Ch. 2 und Owens 1986.

⁴ Zur Unterscheidung von digitaler und analoger Technik siehe Seite 55.

⁵ Redmond and Smith 1980; Owens 1996, S. 84f.

⁶ Trotz des Ausscheidens am MIT blieb Bushs Einfluß auf die Forschung am MIT aufgrund seiner persönlichen Kontakte und seines wissenschaftlichen Rufes weitgehend erhalten. Während seiner Zeit beim NDRC und O.S.R.D. war Bush für das MIT der wichtigste Vermittler von staatlichen Forschungsaufträgen. Vgl. Zachary 1997, S. 89–188, passim.

2. Vision und Kontinuität: Vannevar Bushs Memex



Abbildung 2. Rockefeller Differential Analyzer um 1942

Bush nicht nur als Standesvertreter einen Namen gemacht, sondern auch mit seinem Bericht »Science – The Endless Frontier« (1945) das Fundament für die politisch-organisatorische Struktur der amerikanischen Nachkriegsforschung gelegt.¹

Im Jahre 1945 veröffentlichte Bush schließlich einen Aufsatz, der das Resultat sowohl seiner Tätigkeit als Entwickler von elektronischen Analogrechnern als auch seiner Tätigkeit als Wissenschaftsmanager war. Dabei handelte es sich um den Aufsatz »As we may think«, der im Sommer und Herbst 1945 im *Atlantic Monthly* und im *Life*-Magazin erschien.² Bush schilderte in diesem Aufsatz eine fiktive Maschine, von ihm Memex genannt, die ein hochentwickeltes Gerät zur Informationsverarbeitung sein sollte. Obwohl der Aufsatz zu seiner Entstehungszeit sehr viel publizistische Aufmerksamkeit erregte³, wurden in der Folgezeit keine Versuche zur Realisierung des Memex unternommen. Dies hatte seine Ursache unter anderem in der ab 1946 verstärkt betriebenen Entwicklung elektronischer Digitalcomputer, die die analoge Rechentechnik rasch verdrängte.

Erst seit Beginn der sechziger Jahre wurde Bushs Aufsatz als Ideenlieferant diverser technischer und wissenschaftlicher Entwicklungslinien aufgegriffen. Unter anderem kann diese Veröffentlichung auch als erste geschlossene Formulierung des Konzepts der interaktiven Mensch-Maschine-Kommunikation bzw. einer informationsverarbeitenden Maschine als persönlichem Werkzeug gelten. Aus die-

¹ Wildes and Lindgren 1985, S. 69ff.; Kevles 1990; Zachary 1992 .

² Bush 1945b; Bush 1945a.

³ Zum Beispiel Anonymous 1946, Besterman 1945.

sem Grund ist es von Interesse, nicht nur einen Blick auf die Technik des Memex zu werfen. Die grundlegenden Ideen Bushs über die Organisation von Information, den Gebrauch des Memex und dessen Rolle im Leben der potentiellen Benutzer haben einen sehr viel größeren Einfluß auf die Entwicklung von Hard- und Software seit 1945 gehabt als seine konkreten Realisierungsvorschläge. Für ein tieferes Verständnis dieser Ideen muß zunächst Bushs Vorstellung über die Tätigkeit und Bedeutung des Wissenschaftlers und Ingenieurs erläutert werden, die sich auch im Entwurf des Memex widerspiegelt.

2.2 Der Ingenieur zwischen Theorie und Praxis

Die Verwissenschaftlichung der Technik führte zu Beginn der dreißiger Jahre zu einer tiefgreifenden Veränderung der Ingenieurausbildung und -tätigkeit. Der lange Zeit vorherrschende Typus des selbständigen Erfinders, dessen bekanntester Vertreter Thomas A. Edison war, hatte sich zu dieser Zeit als Folge der immer komplexer werdenden technischen Problemstellungen überlebt.¹ Neben der Gründung naturwissenschaftlich arbeitender Forschungslaboratorien durch große Industrieunternehmen (z. B. durch die Bell Telephone Comp. und die General Electric Comp.) war auch das veränderte Curriculum an den Technischen Hochschulen Ausdruck dieser Entwicklung.² Vannevar Bush hatte als Hochschullehrer und Standesvertreter einen bedeutenden Anteil an dieser Entwicklung. Dabei konzentrierte er sich stärker auf politische und organisatorische Fragen von Wissenschaft und Technik: 1937 schrieb er über den Ingenieur und seine Beziehung zur Regierung, 1939 über den Geist des Ingenieurberufs und 1943 über den Beitrag der Forschung zu den Kriegsanstrengungen. 1945 wurde schließlich seine bekannteste forschungspolitische Schrift »Science – The Endless Frontier« als Bericht an den Präsidenten über die Organisation der Forschung nach dem Zweiten Weltkrieg veröffentlicht.³

2.2.1 Ein Vertreter der alten Ordnung

Von seiner Herkunft und Ausbildung her ist Bush ein typischer Vertreter traditioneller Ingenieuritätigkeit, bei der Herstellung von Maschinen, Gebäuden und neuen technischen Geräten, also der Entwurf technischer Artefakte, im Mittelpunkt steht. Wenn der Entwurf und die Ausführung eines technischen Geräts nicht mehr in einer Hand liegt, wie es seit der Industriellen Revolution üblich war, muß die

¹ Hughes 1991, Kapitel 4.

² Braun 1994, Kapitel 1; Wildes and Lindgren 1985, S. 33.

³ Bush 1937; Bush 1939; Bush 1943; Bush 1990.

2. Vision und Kontinuität: Vannevar Bushs Memex

Konstruktionszeichnung zum Mittler zwischen der Idee und der Ausführung werden. Diese beiden Aspekte bestimmten seit dem 19. Jahrhundert die Ausbildung und Tätigkeit von Ingenieuren. Neben der häufig spielerisch wirkenden Freude am Erfinden von bislang nicht erdachten Mechanismen und Geräten¹ steht die Methodik des visuellen Entwurfs, also die Fähigkeit, sich die Funktionsweise eines technischen Artefakts vorstellen und diese Vorstellung dann in einen Konstruktionsplan umsetzen zu können.²

Neben diesen konstruktiven Fähigkeiten mußte der Ingenieur aber auch wirtschaftlich-organisatorische Kompetenzen besitzen, die ihn letztlich vom reinen Erfinder unterscheiden. Neben die Funktionsfähigkeit einer Erfindung mußte auch ihr praktischer und wirtschaftlicher Nutzen treten, der aus der Invention erst eine Innovation macht. Der Produktionsprozeß mußte geplant und umgesetzt werden; nicht zuletzt mußte ein Ingenieur auch ein gewisses Gefühl fürs Geschäft und eine Befähigung zur Menschenführung besitzen. Diese Qualitäten machten nach Bushs Meinung einen naturwissenschaftlich-technisch Geschulten erst zum wirklichen Ingenieur.³

Obwohl Naturwissenschaft und Technik seit dem Beginn des »Baconschen Zeitalters« eine bemerkenswerte Symbiose eingegangen sind, haben die Naturwissenschaften ihre Wurzeln in der Philosophie und sind deswegen ihrem Wesen nach eher eine zweckfreie Kultur. Nicht die Lösung realweltlicher Probleme, also die Schaffung des Machbaren, sondern die Entdeckung von Wahrheiten und Tatsachen steht in ihrem Mittelpunkt.⁴ Dieser Gegensatz von konkreter Ingenieurkunst und Naturwissenschaft hat sich erst in den vergangenen Jahrzehnten in Form der sogenannten Techno-Science verwischt.⁵

Insofern hatte der klassische Ingenieur sehr viel mehr mit einem Künstler als mit dem Naturwissenschaftler gemein und bezeichnenderweise waren ja auch viele Künstler seit dem Beginn der Neuzeit gleichzeitig technische Innovatoren oder Visionäre. Insofern bestand lange Zeit tatsächlich ein tiefer und häufig unüberbrückbarer Gegensatz zwischen Wissenschaft und Technik, die der amerikanische Technikhistoriker Edwin T. Layton treffend als unterschiedliche Ideologien bezeichnet hat.⁶

¹ Bush 1972, S. 157.

² Ferguson 1993, S. 15ff., 49ff.; König 1999, S. 180ff.

³ Bush 1972, S. 157.

⁴ Segal 1996; Böhme 1993, insbesondere S. 9–19.

⁵ Man denke hier an die kaum mehr voneinander trennbaren Disziplinen z. B. der Biotechnologie und der Biowissenschaft oder der Teilchenphysik und der Beschleunigertechnik. Vgl. Kreibich 1986, S. 133.

⁶ Segal 1996; Layton 1976, S. 695. – Wie bei politischen Ideologien ist auch die Ideologie in der Wissenschaft und Technik des 19. und frühen 20. Jahrhundert durch die Ausbildung einer Orthodoxie und dadurch verursachte Unbeweglichkeit gekennzeichnet.

Bushs gesamte berufliche Laufbahn gibt ein Zeugnis über seine Herkunft aus der althergebrachten Ingenieurtradition ab: Er erhielt zwischen 1914 und 1974 fast 50 Patente und wußte diese mit unterschiedlichen Partnern auch in wirtschaftlich erfolgreiche Produkte umzusetzen.¹ Dennoch erkannte er, daß die klassischen Fähigkeiten des Ingenieurs in einer Zeit immer komplizierterer Hochtechnologieprodukte nicht mehr ausreichten. Unter diesem Aspekt ist auch das vielkritisierete Zitat aus der Zusammenfassung von Bushs Bericht »Science – The Endless Frontier« zu verstehen, in dem er schreibt:

»New products, new industries, and more jobs require continuous additions to knowledge of the laws of nature, and the application of that knowledge to practical purposes. Similarly, our defense against aggression demands new knowledge so that we can develop new and improved weapons. This essential, new knowledge can be obtained only through basic scientific research.«²

Dieses knappe Fazit wird häufig dahingehend interpretiert, daß Grundlagenforschung die alleinige Wurzel aller Innovationen sei und dadurch automatisch zum technischen und wirtschaftlichen Fortschritt beitrage. Dies ist später als *Assembly-Line-Modell* bekannt geworden und hat sich mittlerweile als unrichtig herausgestellt.³ Darüber hinaus ist dies auch eine unangemessene Vereinfachung von Bushs Vorstellungen über das Verhältnis von Wissenschaft und Technik. Bush drückt in der Zusammenfassung von »Science – The Endless Frontier« nämlich lediglich seine Überzeugung aus, daß der Ingenieur in der Zukunft mit den klassischen Ingenieurfähigkeiten allein nicht mehr ausreichend ausgebildet sei und wichtige naturwissenschaftliche Kenntnisse und Methoden den Ingenieurberuf ergänzen müßten.⁴

Deutlicher hatte Bush dieser Überzeugung bereits einige Jahre früher vor einem Kongreßausschuß Ausdruck gegeben. Dort sagte Bush unter anderem, der Ingenieur (oder unabhängige Erfinder) habe »... einen sehr viel weiteren Ideenhorizont und ... bringt oft aus dem Nichts ein verblüffend funktionstüchtiges neues Gerät oder eine Kombination hervor, die sich als nützlich erweist und ohne seinen Scharfsinn vielleicht niemals erfunden worden wäre ... Heute werden ebenso

¹ Während der zwanziger Jahre brachte Bush sein technisches Wissen als Berater in eine Reihe von Unternehmen ein, darunter die American Research and Development Corp. (später Raytheon) und die Metals and Control Corp. (später eine Abteilung von Texas Instruments). Vgl. Krim 1993.

² Bush 1990, S. 5.

³ Greenberg 1966; Abelson 1966; Kaiser 1995, S. 367ff. – Eine ausführliche Diskussion von Bushs Intentionen und die aus seinem Bericht abgeleiteten Folgerungen kann an dieser Stelle nicht geleistet werden. Beispielhaft sei auf die Aufsätze von Layton (1971) und Wise (1985) hingewiesen.

⁴ Bush 1939.

2. Vision und Kontinuität: Vannevar Bushs Memex

viele neue Ideen geboren wie je zuvor, und während ein großes Forschungslaboratorium in diesem Lande ein sehr bedeutender Faktor für den Fortschritt der Wissenschaft und das Entstehen neuer industrieller Kombinationen ist, kann es keineswegs alle Bedürfnisse erfüllen. Der selbständige Erfinder, die kleine Gruppe, die Persönlichkeit, die eine Situation erfährt, weil sie fähig ist, unabhängig zu denken, ist oft ein unerhört wichtiger Faktor beim Hervorbringen von Dingen, auf die man andernfalls noch lange würde warten müssen.«¹

2.2.2 Der Technokrat

Die hochgesteckten Ansprüche an den Ingenieurberuf und die konservative Grundhaltung im Verbund mit seiner Rolle als wichtigem Mitglied der US-Regierungsadministration machen Bush auch zu einem typischen Vertreter der Technokratie, einer hauptsächlich amerikanischen Bewegung, die ihren Höhepunkt zwischen dem Ende des 19. Jahrhunderts und dem Zweiten Weltkrieg erlebte. Ziel der Technokratie war die Organisation des wirtschaftlichen und gesellschaftlichen Lebens nach den Erfordernissen und dem Vorbild der Technik, um auf diese Weise die Effektivität der Produktion und damit den allgemeinen Lebensstandard zu erhöhen.² Die Grundidee und das Selbstverständnis lassen sich treffend durch einen Ausspruch von Arthur D. Little kennzeichnen, den Bush in einem Vortrag von 1937 zitierte:

»The late Arthur D. Little in an essay . . . defined the ›fifth estate‹ as: ›those having the simplicity to wonder, the ability to question, the power to generalize, the capacity to apply. It is, in short, the company of thinkers, workers, expounders, and practioners upon which the world is absolutely dependent for the preservation and advancement of that organized knowledge which we call science. (. . .) It is they who bring the power and the fruits of knowledge to the multitude who are content to go through life without thinking and without questioning . . . ‹.«³

Nach einer solchen Definition haben Ingenieure und Wissenschaftler als technokratische Elite grundsätzliche Verpflichtungen und Rechte. Ihre Qualifikation verpflichtet sie zum »Dienst« an der Gemeinschaft und zur Mitgestaltung sozialer Prozesse.⁴ Aus dieser gesellschaftlichen Verpflichtung leitete Bush nicht nur den

¹ U. S. Congress, Temporary National Economic Committee 1939, S. 871f.

² Hughes 1991, S. 445f. Zur Geschichte der Technokratiebewegung zwischen den Weltkriegen vergleiche Willeke 1995.

³ Bush 1937, S. 930.

⁴ Bush verwendete in diesem Zusammenhang explizit den Begriff *ministry*, der auch einen Hinweis auf Bushs religiöse Prägung durch seinen Vater, einen Laienprediger, gibt. Vgl. Bush 1939, S. 195 und Gleiser 1980, S. 141.

natürlichen Primat der Technik über Wirtschaft und Politik, sondern auch einen besonderen hohen Anspruch an die Verantwortung der technokratischen Elite für die Auswirkungen der von ihr geschaffenen Technik ab. Dies war bereits während der dreißiger und vierziger Jahre eine klare Minderheitsmeinung, die Bush mit sehr viel Vehemenz vertrat. Dennoch konnte er nicht verhindern, daß der Status des Ingenieurs damals wie heute geringer ist als der des Naturwissenschaftlers.¹ Auch die Auflösung des von ihm verteidigten *American Engineering Council* als gesellschaftspolitischem Organ der Ingenieure im Jahre 1940 konnte er nicht aufhalten.²

Diese elitäre Grundhaltung, die an Platons Philosophenkönige³ erinnert, wurde durch eine extrem antizentralistische Haltung ergänzt, die Bush trotz seiner Beratertätigkeit für Präsident Franklin D. Roosevelt in Gegensatz zu den politischen Maximen des *New Deal* geraten ließ.⁴ Er lehnte – außer in wichtigen Gebieten wie der nationalen Sicherheit – jede Zentralisierung der Forschung z. B. in Form einer *National Science Foundation* vehement ab. Es sollte aber eine kollektive, dezentrale Koordinierung erfolgen, die Bush am MIT und vor allem beim O.S.R.D. als besonders produktiv kennengelernt hatte. Die Initiative sollte allerdings in Form eines *Grass-Root-Movements* organisiert sein, also von der wissenschaftlichen Basis ausgehen, um die Forschung nicht unnötig einzuengen.⁵

Um diese Form der Wissenschaft zu unterstützen, hatte sich Bush bereits vor dem Zweiten Weltkrieg mit der Frage auseinandergesetzt, welche technischen Mittel man der wissenschaftlich-technischen Elite – also letztlich ihm selbst – zur Verfügung stellen konnte. Ein solches Werkzeug sollte der Memex sein.

2.3 Das Informationsproblem ...

Der Ursprung des Memex läßt sich bis in die frühen dreißiger Jahren zurückverfolgen.⁶ Bush war zu dieser Zeit einerseits gerade Vizepräsident des MIT gewor-

¹ Da sich das Ansehen der Ingenieure auch während des Krieges innerhalb der Forschungsbehörden nicht besserte, wurde das komplette O.S.R.D. Personal »augenblicklich zu Wissenschaftlern«. Vgl. Bush 1972, S. 53f.

² Layton 1986, S. 242.

³ Platon 1958, 6. Buch. Dort argumentiert Plato, der Hauptzweck eines Staates sei die Erziehung der Bürger zur Tugend, die den Menschen Glückseligkeit schafft. Deshalb müßten in einem idealen Staatswesen die Herrschenden selbst zu Philosophen bzw. Wissenschaftlern werden. Siehe dazu auch Russell 1993, S. 125–140.

⁴ Zum *New Deal* siehe auch Raeithel 1995, Bd. 3, S. 5–35; Angermann 1995, S. 136–194, dort finden sich auch umfangreiche Verweise auf weitere Literatur. Mit besonderem Blick auf die Technikentwicklung vergleiche Hughes 1991, S. 355–384.

⁵ Reingold 1987, S. 301–311; Bush 1944 und Bush 1946a.

⁶ Bush datierte seine ersten Überlegungen zu diesem Thema etwa auf das Jahr 1932 zurück. Vgl. Bush 1967, S. 76 und Bush 1972, S. 190.

2. Vision und Kontinuität: Vannevar Bushs Memex

den und begann seine Karriere als Funktionär und Politiker. Andererseits hatte ihn die Entwicklung des *Differential Analyzer* in die vorderste Reihe der Analogrechnerentwicklung gebracht, in deren Folge eine erste Umwertung des Begriffs »Computer« stattfand.¹

Bis 1933 war Bush als Autor ein typischer Techniker wie viele andere gewesen, seine zahlreichen Veröffentlichungen gehörten zu den Bereichen Elektrotechnik und Maschinenbau, die Artikel waren geradlinig geschriebene wissenschaftliche Abhandlungen. Bush erreichte aber bei dem von ihm formulierten Informationsproblem die Grenzen seiner Gestaltungsmöglichkeiten, da es mit der ihm eigenen Methodik nicht zu lösen war. In dieser Situation veröffentlichte Bush im Januar 1933 in der Zeitschrift *Technology Review* den Essay »The Inscrutable 'Thirties«, in dem er das Informationsproblem der modernen Wissenschaften formulierte. Dieses Problem bestand darin, daß das naturwissenschaftlich-technische Wissen schneller zunahm, als es selbst der aufmerksamste Leser der Fachliteratur aufnehmen und verarbeiten konnte. Bush faßte dieses Problem einige Jahre später so zusammen:

»We are making enormous strides in the development of methods for creating a record of what we learn . . . We are also making strides in developing means for the transmission of ideas from one to another or from a central point to great audiences. But in one exceedingly important phase of the whole problem we are making little progress indeed. This is the phase of finding in the record the information that we need.«²

Bush versuchte, das Informationsproblem ingenieurmäßig anzugehen, indem er den Zugriff auf technisches und wissenschaftliches Wissen, das in den Bibliotheken gespeichert war, mechanisieren und damit effektiver machen wollte. Nachdem er das Problem erkannt hatte, verschaffte sich Bush zunächst einen Überblick über die bestehende Technologie, nahm Anleihen bei besonders erfolgversprechenden technischen Entwicklungen und erarbeitete dann einen eigenen Entwurf. Das Resultat dieses Prozesses waren der *Navy Comparator* und der *Rapid Selector*, relativ kleine, auf Basis der Mikrofilmtechnik arbeitende Maschinen, die vor und während des Zweiten Weltkrieges entstanden. Beim Entwurf des Memex, so wie er ihn in seinem Essay von 1945 schilderte, erweiterte Bush diese Maschinen um weitere Hochtechnologie.

Zunächst aber griff Bush auf eine zumindest in der englischsprachigen Welt verbreitete Form der Auseinandersetzung mit den zukünftigen Möglichkeiten von Wissenschaft und Technik zurück: die technisch-wissenschaftliche Utopie. Bushs

¹ Ceruzzi 1991, S. 237.

² Bush 1955, S. 303.

Aufsatz »The Inscrutable 'Thirties« ist eine literarische Satire, eine wissenschaftliche Beschreibung der dreißiger Jahre aus der Perspektive einer nicht näher erläuterten Zukunft, mit der aktuelle Probleme und Fehlentwicklungen kritisiert wurden.

Bushs Schriften über den Memex stehen in der Tradition einer positiven utopischen Literatur, die eine perfekte Zukunft ins Auge faßt und dazu aktuelle technische Entwicklungen in die Zukunft extrapoliert, um der Wissenschaft kreative Impulse geben zu können.¹ Sie sind aber auch Bestandteil der technokratischen Vorstellungen Bushs, daß eine technische und wissenschaftliche Elite an der Spitze der kulturellen und politischen Hierarchie stehen sollte; eine Vorstellung, die vor allem in Bushs späteren politischen Schriften zum Ausdruck kommt.² Dieses Sendungsbewußtsein dürfte auch der Grund dafür sein, daß Bush versuchte, seine Memex-Essays in leserstarken, aber auch anspruchsvollen Zeitschriften zu veröffentlichen.

»The Inscrutable 'Thirties« transportierte eine zweifache Botschaft. Zum einen beschrieb Bush die sich rasch verändernde Welt der frühen dreißiger Jahre, in der die industrielle Massenproduktion und der schnell zunehmende Individualverkehr große gesellschaftliche, wirtschaftliche und ökologische Probleme schuf. Für Bush waren die »unergründlichen« dreißiger Jahren aber nur eine Zeit des Übergangs, die mit den Kräften von Wissenschaft und Technik überwunden werden könnte.³

Zum anderen beschreibt Bush in seinem Aufsatz den Ablauf eines gewöhnlichen Tages im Leben eines Professors in Neuengland, darunter auch seine Arbeit in einer Bibliothek. Er versucht damit, dem Leser eine spezielle Quelle der Frustration für Wissenschaftler und Ingenieure zu vermitteln, eben das Informationsproblem.

»The library, to which our professor probably turned, was enormous. Long banks of shelves contained tons of books . . . He had to paw over cards, thumb pages, and delve by the hour. It was time-wasting and exasperating indeed. Many of us well remember the amazing incredulity which greeted the first presentation of the unabridged dictionary on a square foot of film. The idea that one might have the contents of a thousand volumes located in a couple of cubic feet in a desk, so that by depressing a few keys one could have a given page instantly projected before him, was regarded as the wildest sort of fancy. (. . .) It seemed that the greater the technical difficulties which accomplished some really revolutionary propositions, the more

¹ Nyce and Kahn 1991b, S. 47.

² Kevles 1987, S. 346ff.

³ Bush spricht in diesem Zusammenhang von »the age of frustration« und charakterisiert die Zeit der großen Depression als »the last great economic readjustment.« Vgl. Bush 1946c, passim.

2. Vision und Kontinuität: Vannevar Bushs Memex

casually the ordinary citizen accepted its consummation as being temporarily delayed but a fortnight or so.«¹

Das eigentliche Problem, so Bush in seiner elitären Haltung, liege darin, daß die Öffentlichkeit so wenig vom Stand der Technik verstehe, daß sie nicht zwischen dem Möglichen und dem Unmöglichen zu unterscheiden wisse. Er rechtfertigt so den Übergang von der wissenschaftlichen Abhandlung zur »aufrichtigen Prophezeiung« als Fortsetzung der Wissenschaft mit anderen Mitteln.²

Es verwundert daher kaum, daß die fiktive Maschine aus »The Inscrutable Thirties« – der erste Entwurf des Memex – sehr viel Ähnlichkeit mit dem *Navy Comparator* und dem *Rapid Selector* aufweist, die unter der Leitung von Vannevar Bush zwischen 1935 und 1942 am MIT entstanden.

2.4 ... und erste Lösungsversuche

2.4.1 Der Navy Comparator

Die Entwicklung des niemals fertiggestellten *Navy Comparators* fällt sowohl in die Zeit der Umstrukturierung des MIT als auch der Neuorientierung von Bushs Forschung. Nach der Fertigstellung des *Differential Analyzer* suchte Bush Sponsoren, die den Ausbau des *Center of Analysis* zum führenden Entwickler von Rechenmaschinen finanzieren würden. Dabei dachte er an den Einsatz von drei fortschrittlichen Technologien: Mikrofilm, photoelektrische Bauelemente und digitale Elektronik.

Bush trat in den frühen dreißiger Jahren an die U.S. Navy als einen der ersten potentiellen Geldgeber heran, denen er die Entwicklung einer besonders leistungsfähigen kryptoanalytischen Maschine vorschlug. Dem *Federal Bureau of Investigation* (FBI) und dem *U.S. Patent Office* schlug Bush die Entwicklung eines photoelektrischen Geräts zur Dokumentenverwaltung (*document selector*) vor. Aber erst 1935 machte die Navy Bush für einen Betrag von 10 000 \$ zum Berater bei der Entwicklung von Dechiffriergeräten. Obwohl dieser Vertrag die Entwicklung eines Gerätes nicht vorsah, ließ Bush ab Mitte 1937 am MIT unter Leitung von Waldron S. McDonald den *Navy Comparator* entwickeln. Da aber in der Zwischenzeit die Rockefeller-Stiftung als Finanzier für die großangelegte Weiterentwicklung des *Differential Analyzer* aufgetreten war, hatte das *Comparator*-Projekt für ihn eine finanziell und personalmäßig geringe Priorität.³

¹ Bush 1946c, S. 9f.

² Vannevar Bush, »Mechanization and the Record«, unveröffentlichtes Typoscript, 1939. Vannevar Bush Papers, Library of Congress, Box 138, Speech Article File.

³ Burke 1991, S. 145f.

Aufgabe des *Navy Comparators* sollte die Feststellung von Übereinstimmungen zwischen zwei verschlüsselten Nachrichten sein.¹ Diese sollten in Form von papiernen Lochstreifen codiert und mit Hilfe von Photozellen verglichen werden.² Zum Zählen der Übereinstimmungen wurden neue elektronische Digitalzählerschaltungen eingesetzt, die ursprünglich für Strahlungsmessungen in der Kernphysik entwickelt worden waren.³ Insofern betrachtete Bush den *Comparator* richtig als erstes elektronisches Datenverarbeitungsgerät. Bush hoffte, mit dieser Anordnung eine um hundertfünzigmal größere Verarbeitungsgeschwindigkeit zu erreichen als mit den leistungsfähigsten damals verwendeten Lochkartenmaschinen.⁴

Aber schon bald verzögerte sich die Entwicklung des *Comparators* aus technischen und personellen Gründen. Neben dem krankheitsbedingten mehrmonatigen Ausfall des Projektmanagers McDonald traten massive Probleme mit den Einzelkomponenten auf. Die digitalen Zähler und die optischen Leser erwiesen sich als höchst unzuverlässig, vor allem aber verweigerte der Lochstreifenstanzer, der die Verarbeitungsergebnisse ausgeben sollte, häufig seinen Dienst. Trotzdem entschied sich Bush, die Entwicklung des *Comparators* fortzuführen, und übergab am Ende der Projektlaufzeit im Frühjahr 1938 das immer noch nicht voll funktionstüchtige Gerät in Washington an die Navy. Wegen der häufigen Ausfälle und der unzureichenden technischen Kompetenz des Marinepersonals wurde der *Comparator* bereits nach kurzer Zeit wieder außer Betrieb genommen. Mit der Einstellung des Projekts zur Entwicklung von Dechiffriermaschinen durch die U.S. Navy Ende der dreißiger Jahre war schließlich auch der erste Entwurf eines Memex gestorben.⁵

2.4.2 Der Rapid Selector

Nachdem sich das *Center of Analysis* durch die Förderung der Rockefeller-Stiftung nach 1935 in die von Vannevar Bush geplante Richtung entwickelte, konnte Bush auch seine anspruchsvollen Pläne zur Verbindung von Mikrofilm,

¹ Der Grundgedanke der Dechiffrierung ist nicht die unmittelbare Entschlüsselung von chiffrierten Nachrichten sondern die Ermittlung des verwendeten Chiffrierverfahrens. Dazu werden in verschlüsselten Nachrichten, deren Klartext bekannt ist, wiederkehrende Muster identifiziert und gezählt. Mit Hilfe von logischen oder statistischen Verfahren können dann Rückschlüsse auf den Chiffrierschlüssel gezogen werden.

² Die Mechanik des *Navy Comparator* ähnelt der des *Rapid Selectors* (Vgl. Abb. 3, S. 49). Im Unterschied zum *Selector* lagen die beiden zu vergleichenden Dokumente aber im vollen Wortlaut auf Lochstreifen vor, und jedes einzelne Zeichen des Textes wurde auf Identität verglichen.

³ Wildes and Lindgren 1985, S. 230.

⁴ Burke 1991, S. 147.

⁵ Burke 1993; Burke 1994, S. 160ff.

2. Vision und Kontinuität: Vannevar Bushs Memex

Optoelektronik und Digitaltechnik in einem neuen Memex-Entwurf weiterverfolgen, der diesmal *Rapid Selector* genannt wurde.

Der *Rapid Selector* war – wie das Gerät aus »The Inscrutable 'Thirties« – als Maschine zur Verwaltung von großen Datenbeständen und zur Unterstützung wissenschaftlicher Tätigkeit gedacht. Bush nahm Kontakt zu Freunden am MIT, bei Eastman Kodak Ltd. und bei der National Cash Register Company (NCR) auf, die das *Rapid Selector*-Projekt schließlich mit 25 000 \$ für zwei Jahre finanzierten. Das Entwicklungsteam, bestehend aus John Howard, Lawrence Steinhardt und John Coombs, wurde im Frühjahr 1938 von Bush zusammengestellt. Aber auch die Entwicklung des Rapid Selectors stand unter keinem glücklichen Stern. Das Team verlor durch den Wechsel Bushs zur *Carnegie Institution* schon bald an Dynamik. So entsprach der Prototyp des *Rapid Selectors* im wesentlichen Bushs ursprünglichem Entwurf, Änderungen waren nur dort gemacht worden, wo es die praktische Umsetzung unumgänglich machte. Immerhin wurde er aber fristgerecht und unter Einhaltung des Budgets im Frühjahr 1940 fertiggestellt und im September 1942 patentiert.¹

In Bushs Entwurf (Abb. 3) werden die Dokumente (bzw. deren Zusammenfassungen) um den Faktor 25 verkleinert und auf der einen Hälfte eines 35 mm breiten Mikrofilms abgespeichert. Die andere Hälfte jedes Filmbildes sollte dreizehn Codebuchstaben enthalten, die sogenannten *associations*, mit denen eine Reihe von Schlüsselworten codiert werden konnten, die den Inhalt des Dokuments möglichst genau beschreiben. Jeder Buchstabe wurde durch eine Reihe transparenter Mikropunkte repräsentiert. Bush plante, lange Rollen 35-mm-Film mit Dokumenten und Codes so schnell an einer photoelektrischen Leseinheit vorbeizuführen, daß bei einer Geschwindigkeit von 183 cm/s über 60 000 Dokumente pro Minute überprüft werden konnten. Das wäre mehr als hundertmal schneller als bei damals existierenden Lochkartenmaschinen gewesen. Obwohl also die Leseinheiten des *Selectors* und des *Comparators* sehr ähnlich waren, gab es signifikante Unterschiede. Beim *Comparator* mußten die Codes auf zwei sich bewegenden Papierstreifen verglichen werden. Dagegen bestand die Suchmaske des *Selector* nur aus einer stationären Karte.² Während im *Comparator* die Übereinstimmungen nur *gezählt* wurden, aktivierte die vollständige Übereinstimmung der Codes beim *Selector* eine Kopiereinrichtung. Diese bestand aus einem Stroboskop-Blitz,

¹ Burke 1991, S. 151ff.; Morse 1942 – Das Patent wurde, wie bei solchen Zusammenarbeiten üblich, der Eastman Kodak Corporation erteilt, die die Entwicklung des Rapid Selectors mitfinanziert hatte.

² Bush 1972, S. 188f.; Burke 1991, S. 150f.

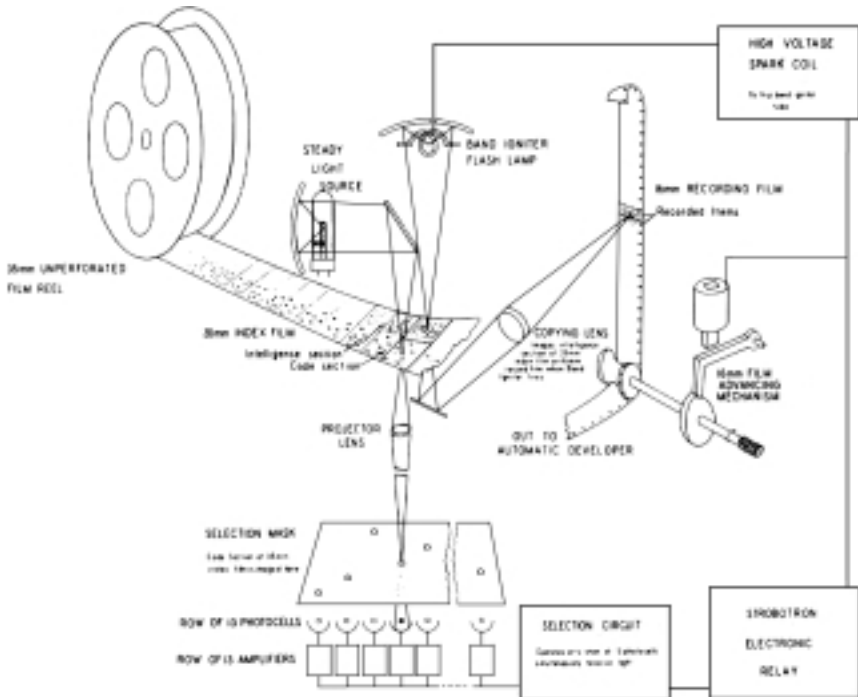


Abbildung 3. Schema des Rapid Selectors

mit dessen Hilfe die gesuchten Dokumente ohne Anhalten des Filmstreifens reproduziert werden konnten.¹

Das eigentliche Problem des *Rapid Selectors* lag allerdings weniger in der Hardware als in den noch unausgereiften Überlegungen zur Codierung der Dokumente. Die digitalen Logikschaltkreise des *Selectors* waren, wohl auch wegen der schlechten Erfahrungen beim *Navy Comparator*, weniger komplex als die der Vorgängermaschine und ließen weniger logische Verknüpfungen bei der Suche zu. Auch durch die Verwendung von nur einer statt der ursprünglich geplanten fünf Leseeinheiten wurde auf komplexere Verknüpfungsmöglichkeiten verzichtet, so daß bei einer Suche nach mehreren Schlüsseln nacheinander mehrere Suchläufe durchgeführt werden mußten. Bush war sich offensichtlich nicht be-

¹ Das Stroboskop wurde während der dreißiger Jahre von Harold Edgerton, einem Kollegen Bushs am MIT, entwickelt. Es konnte Lichtblitze von weniger als 10 ms Dauer erzeugen. Vgl. Wildes and Lindgren 1985, S. 138ff. und Bush 1972, S. 188.

2. Vision und Kontinuität: Vannevar Bushs Memex

wußt, wie limitierend ein serielles, nicht-hierarchisches Codierungsschema war.¹ Bush beschreibt auch nicht, wie die Spezialisten in unterschiedlichen Disziplinen sich über ein einheitliches Codierungsschema einigen sollen und wie man dann die konsistente Verwendung dieses Schemas sicherstellen kann. Schließlich wurde die Frage nach der Benutzbarkeit von dreizehnstelligen Codeworten z. B. HTVOQTXZ8MKMO) bei der praktischen Verwendung des *Rapid Selector* noch nicht einmal gestellt.²

Die 1940 präsentierte Maschine war auch nicht der elegante, schreibischtische große und preiswerte Memex, den Bush 1932 beschrieben hatte. Der *Selector* bestand aus einem 2,10 m hohen Schrank, der die Filmlaufwerke, die Leseeinheiten und die Kopiereinrichtung enthielt; die Elektronik nahm einen zweiten Schrank in Anspruch. Eine modifizierte elektrische Schreibmaschine zur Dateneingabe, die allein die Größe eines kompletten Schreibtisches hatte, vervollständigte den *Rapid Selector*. Im Gegensatz zum *Navy Comparator* war der *Selector* zwar funktionsfähig, erreichte aber wegen der weiterhin existierende Probleme mit mechanischen und elektronischen Bauelementen nicht die angestrebte Verarbeitungsgeschwindigkeit.³

Die Untersuchungen, die John Howard nach der Fertigstellung des Prototypen über mögliche Einsatzmöglichkeiten anstellte, waren sogar noch weit ernüchternder. Bei allen von ihm ausgewählten Anwendungsbereichen (u. a. FBI, Patentamt und Sozialversicherung) kam er zu dem Ergebnis, daß der *Selector* keinen Geschwindigkeitsvorteil gegenüber etablierten Verfahren brächte.⁴ Nachdem Howard zu der Erkenntnis gekommen war, daß es sinnvoller gewesen wäre, erst ein angemessenes Codierungsschema für eine bestimmte Anwendung zu entwickeln und erst danach eine passendes Gerät zu konstruieren, schlug er die Verwendung des *Rapid Selector* als statistische Maschine etwa im *Bureau of Census* oder bei den Geheimdiensten vor. So wurde der *Rapid Selector* zu einer Art verbesserten *Navy Comparator* umfunktioniert, der während des Zweiten Weltkrieges zur Deciffrierung japanischer Codes verwendet wurde.⁵

Mit der kriegsbedingten Auflösung des MIT-Entwicklungsteams starb auch der zweite Ansatz zur Realisierung des Memex. Erst im dritten Anlauf, der auf

¹ Mooers 1950.

² Shaw 1949, S. 168f.; Burke 1991, S. 151ff. Für die Klassifizierung eines Dokuments war stets die Verwendung *aller dreizehn* Codebuchstaben zwingend vorgeschrieben. Bei der Suche mußte außerdem stets der *komplette* Datenbestand durchsucht werden.

³ Burke 1994, S. 193f.

⁴ Die Ergebnisse dieser Untersuchungen sind allerdings wegen der sehr einfachen Suchschemata bei den untersuchten Anwendungen nicht ganz so negativ zu bewerten: Howard kam etwa zu dem Ergebnis, daß die Verwendung eines Telefonbuchs bei der Ermittlung von Telefonnummern ungleich effektiver sei als der *Rapid Selector*.

⁵ Burke 1991, S. 155f.; Burke 1994, S. 195ff., Chapter IX.

den 1939 geschriebenen Aufsatz »As we may think« zurückgeht, sollte Bush nach dem Zweiten Weltkrieg ein größeres und aufnahmeberechtigtes Publikum für seine Ideen finden.

2.5 »As we may think« und der Memex

Vannevar Bushs Aufsatz »As we may think« erschien im Juli 1945 und dürfte in der neueren Technikkultur mittlerweile einer der am häufigsten zitierten Aufsätze überhaupt sein.¹ Dies hat seinen Grund in der Ikonenfunktion, die er für eine Reihe von Forschungsgebieten hat, die »As we may think« als ihren Ausgangspunkt betrachten. Sowohl die *Information Science*² als auch Teilbereiche der Informatik (Information Retrieval, Hypertext, Informationssysteme) reklamieren Vannevar Bush als ihren geistigen Vater.³ Dabei wird nur selten berücksichtigt, unter welchen historischen Vorzeichen der Aufsatz entstanden ist oder welche Absicht Bush damit verfolgte.

Die erste Fassung des Aufsatzes entstand bereits 1939 unter dem Titel »Mechanization and the Record« und gehört noch zu Bushs Bemühungen um die Verbesserung des öffentlichen Ansehens des Ingenieurberufs. Sie sollte in Zeiten der allgemeinen Technikskepsis Zeugnis darüber ablegen, daß Wissenschaft und Technik auch einen positiven Beitrag zur gesellschaftlichen Entwicklung leisten können und nicht nur zu Arbeitslosigkeit und Massenarmut führen. Außerdem wollte Bush mit seinem Aufsatz einen Eindruck vom technischen Fortschritt der dreißiger Jahre vermitteln. In einem Memorandum an Eric Hodgins, den Herausgeber von *Fortune*, wo Bushs Aufsatz ursprünglich erscheinen sollte, schreibt Bush, er wolle darlegen, daß die Anwendung von Wissenschaft und Technik nicht von vornherein schlecht sei. Zu diesem Zweck wolle er ein fortschrittliches Gerät beschreiben, das einen wichtigen Beitrag zur Fortentwicklung des menschlichen Geistes leisten könne.⁴ Aus diesen Umständen erklärt sich die ausgefeilte Rhetorik und der eklektische Charakter des Aufsatzes, die ihm heute manchmal zu Unrecht zum Vorwurf gemacht werden.⁵

¹ Linda C. Smith führt in einem Aufsatz über die Wirkung des Aufsatzes mehr als ein Dutzend Texte an, die allein durch ihren Titel »As we ...« Bushs Aufsatz ihre Reverenz erweisen. Vgl. Smith 1991.

² Für diesen wissenschaftlichen Bereich gibt es keine feststehende deutsche Bezeichnung; die Bereiche Dokumentation, Fachinformation und Bibliothekswesen umfassen jeweils nur einen Teilbereich der *Information Science*.

³ Lilley and Trice 1989, S. 11ff.; Donohue and Karioth 1966; Nielsen 1993, S. 30ff.; Ritchie 1992.

⁴ Bush 1991b, S. 81.

⁵ Buckland 1992, S. 285ff.; Fairthorne 1958.

2. Vision und Kontinuität: Vannevar Bushs Memex

Bush versuchte mehrere Jahre vergebens, »Mechanization and the Record« im *Fortune* Magazin zu veröffentlichen. Die Veröffentlichung scheiterte aber, weil sich Bush weigerte, den anspruchsvollen Essay zu kürzen¹ und für ein Massenpublikum umzuschreiben. Bush zögerte aber auch, den Text in einer kleineren wissenschaftlichen Zeitschrift zu veröffentlichen, weil er über *Fortune* genau die Leserschaft erreichen konnte, die er für wichtig hielt: die wissenschaftlich und technisch interessierte Allgemeinheit.² Die Veröffentlichung des Aufsatzes verzögerte sich weiter, weil Bushs Tätigkeit für das O.S.R.D. während der Kriegsjahre nur wenig Zeit für andere Aktivitäten ließ.³

Zum Ende des Krieges wurde der Memex-Aufsatz schließlich doch noch veröffentlicht, zunächst im Juli 1945 unter dem Titel »As we may think« im *Atlantic Monthly*, einem kleineren, aber renommierten Magazin. Das *Life*-Magazine druckte im September 1945 eine illustrierte und vom Herausgeber gekürzte und kommentierte Version des Aufsatzes ab. So hatte Bush schließlich doch noch Anspruch und breite Öffentlichkeit vereinen können.⁴

Bevor auf die Rolle des Memex für die spätere Computerentwicklung eingegangen werden kann, soll zunächst die in »As we may think« vorgestellte Idee einer persönlichen Maschine zur Informationsverarbeitung und Bushs Anregung zu deren Realisierung näher erläutert werden.

2.5.1 Das fiktive Gerät

Bush beginnt »As we may think« – wie alle seine Memex-Essays – mit einer Darstellung des Informationsproblems und knüpft damit unmittelbar an die Schilderung in »The Inscrutable 'Thirties« an. Der seit Ende des 19. Jahrhundert exponentiell wachsende Bestand wissenschaftlicher Literatur sei mittlerweile nicht mehr überschaubar und führe zu einem unnötig großen Anteil von Mehrfachentdeckungen, vor allem aber bestehe die Gefahr, daß wichtige Forschungsergebnisse unbeachtet blieben.⁵ Dabei liegt für Bush das Problem weniger in der Informa-

¹ Das Manuskript von »Mechanization and the Record« umfaßte 46 maschinenschriftliche Seiten. Im Druck hätte dieser Aufsatz damit eine Länge von etwa 18 Seiten gehabt. Die 1945 veröffentlichte Fassung war mit 8 Seiten für ein Magazin wie *Life* oder *Fortune* bereits außerordentlich umfangreich.

² Bush 1991b, S. 81.

³ Bush 1991b, S. 82; Nyce and Kahn 1991b, S. 51ff.

⁴ Nyce and Kahn 1991b, S. 55f.

⁵ Bush führt in seinem Aufsatz explizit den Fall von Gregor Mendel an, dessen Vererbungsgesetze viele Jahre unbeachtet blieben. Vgl. Bush 1945b, S. 101.

Viele heutige Autoren bringen Bushs Argumentation unzulässigerweise mit der Informationsflut der Nachkriegszeit in Verbindung. Das exponentielle Wachstum wissenschaftlicher Erkenntnisse und Publikationen setzte bereits im späten 19. Jahrhundert ein und stellte zur Entstehungszeit von »As we may think« bereits ein *alltägliches* Problem dar. Vgl. Solla Price 1974, S. 43ff.

tionsflut selbst als in den unzureichenden technischen und organisatorischen Mitteln zur Erfassung und Verarbeitung der Information: »Professionally our methods of transmitting and reviewing the results of research are generations old and by now are totally inadequate for their purpose.«¹

Das Gerät, das Bush in »As we may think« schließlich vorstellte, war rein fiktiv. Es wurde nie gebaut, es hat noch nicht einmal ernsthafte Anstrengungen zur Konstruktion eines Memex gegeben. Selbst die Komponenten, aus denen er bestehen sollte, waren 1939 bzw. 1945 in der von Bush beschriebenen Form nicht vorhanden. Sie sind aber trotzdem keine pure Fantasie, sondern gedankliche Weiterentwicklungen von damals aktuellen Forschungsergebnissen und existierender Hochtechnologie. Die wichtigste Voraussetzung für die Realisierung des Memex war nach Bushs Überzeugung eine leistungsfähige industrielle Produktionstechnik, die zuverlässige und billige Komponenten bereitstellen kann. Diese Bedingung sei zwar zu G. W. Leibniz' und Ch. Babbages Zeiten noch nicht erfüllt gewesen, 1945 sei aber die Zeit für eine Maschine wie den Memex reif.²

Bush machte aus dem eklektischen Charakter seines Aufsatzes kein Geheimnis. Er schrieb unter anderem:

»Technical difficulties of all sorts have been ignored, certainly, but also ignored are means as yet unknown which may come any day to accelerate technical progress as violently as did the advent of the thermionic tube. In order that the picture may not be too commonplace, by reason of sticking to present-day patterns, it may be well to mention one such possibility, not to prophesy but merely to suggest, for prophesy based on extension of the known has substance, while prophesy founded on the unknown is only a doubly involved guess.«³

Ein solches Vorgehen ist für einen Aufsatz mit einem ausdrücklich aufklärerischen Anspruch ungemein geschickt; der Vorwurf fehlender Originalität, wie ihn Michael K. Buckland formuliert hat, geht deshalb auch an der ursprünglichen Intention Bushs völlig vorbei. Vielmehr wartet Bush mit einer ganz anderen Art von Originalität auf. Durch die Kombination von existierenden Verfahren und Geräten entwickelt Bush in seinem Aufsatz auf originelle Weise Lösungsvorschläge für wichtige Problemfelder.

Der Memex, so wie ihn sich Bush vorgestellt hat, ist ein Gerät für den *individuellen* Gebrauch. Es sollte die Bibliothek und die Kartei eines *Wissenschaftlers* ersetzen, für den der Memex eine Erweiterung des persönlichen Gedächtnisses werden sollte: ein *Memory Extender*, kurz Memex.⁴ Das Gerät selbst ist dement-

¹ Bush 1945b, S. 101.

² Bush 1945b, S. 102.

³ Bush 1945b, S. 108.

⁴ Bush 1945b, S. 107.

2. Vision und Kontinuität: Vannevar Bushs Memex

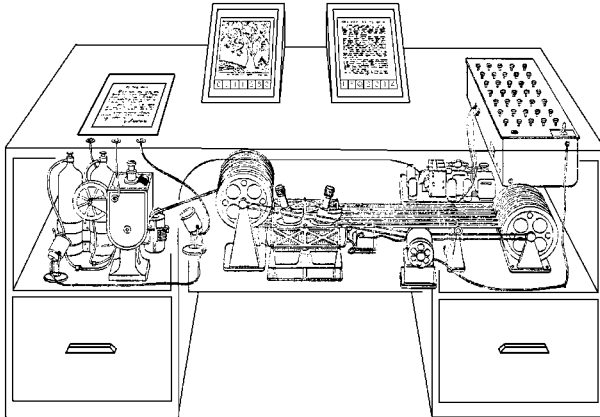


Abbildung 4. Memex
in der Form eines Arbeitstisches

sprechend kompakt aufgebaut. Auf einem der Bilder, die Alfred D. Crimi¹ unter Bushs Aufsicht für die *Life*-Version des Aufsatzes angefertigt hat (Abb. 4), sieht man ein Gerät, dessen äußerer Aufbau einfacher kaum sein könnte: Auf einem gewöhnlichen Schreibtisch sind zwei Bildschirme angeordnet, auf die die gewünschte Information projiziert wird.² Auf der rechten Seite sind eine Reihe von Hebeln und Knöpfen zur Bedienung des Memex angeordnet, das rechteckige Feld auf der linken Seite stellt eine Vorrichtung zum Einlesen von Papierdokumenten dar. Ansonsten präsentiert sich der Memex für den Benutzer als »black box«, die eigentliche Technik ist vor dem Benutzer im Inneren des Schreibtisches verborgen.³

Besonders bezeichnend ist in diesem Zusammenhang, daß Bush stets von seinem Memex als einem *gadget* spricht. *Gadget* bezeichnet im Englischen eine Erfindung oder ein technisches Gerät, hat aber eine leicht negative Nebenbedeutung. Ein *gadget* ist nämlich auch technischer Schnickschnack oder eine nutzlose Spielerei. Damit wird ein Hinweis auf die typische Rollenzuweisung der Technik in der Gesellschaft gegeben. Aufgrund einer verbreiteten Technikangst wünschte sich ein großer Teil der Bevölkerung in den dreißiger Jahren einen Maschinen-

¹ Alfred D. Crimi war ein Zeichner, der zuvor für die Sperry Gyroscope Company gearbeitet hatte. Crimi wurde für die Illustrationen ausgewählt, weil er im Frühjahr 1945 Zeichnungen der ersten Sperry-Computer für *Life* erstellt hatte. Da die Illustrationen unter Mithilfe von Bush entstanden, dürften sie Bushs Vorstellungen wohl sehr nahe kommen. Vgl. Nyce and Kahn 1989, S. 217.

² Bush selbst hat in seinem Aufsatz die Anzahl der Projektionsschirme nicht näher festgelegt, er spricht aber stets im Plural von »translucent screens«. Betrachtet man die Abmessungen des Schreibtisches, so kommt man für die Projektionsschirme auf eine Bilddiagonale von etwa 16 Zoll.

³ Bush 1945b, S. 107.

einsatz nur für untergeordnete und leicht kontrollierbare Aufgaben. Mit seiner Wortwahl drückt Bush seine Überzeugung aus, daß Maschinen, selbst sogenannte denkende Maschinen stets nur *Diener* des Menschen sein können und dürfen.¹

Letztlich sollen wir alle so sein wie der Mathematiker in »As we may think«, Bushs idealisierter Benutzer:

»All else [the mathematician] should be able to turn over to his mechanism, just as confidently as he turns over the propelling of his car to the intricate mechanism under the hood.«²

Die Maschine wird damit zur »black box«, die zwar nützlich für die Arbeit seines Benutzers ist, über deren interne Arbeitsweise er sich aber keine Gedanken machen muß. Funktionalität und technische Realisierung werden, wie bei vielen High-Tech-Produkten der Nachkriegszeit, voneinander getrennt.³ Damit steht der Memex in einem krassen Gegensatz zu den 1939 bzw. 1945 existierenden informations- und datenverarbeitenden Maschinen, bei denen die Mensch-Maschine-Schnittstelle noch kein Thema war, zumal sie sich allein durch ihre Größe und ihr Gewicht auch nicht im entferntesten als Geräte für das Büro eigneten.⁴

Obwohl der Memex in erster Linie ein elektro-optisches Gerät ist, ist er ein direkter Nachfahre von Bushs analytischen Maschinen der dreißiger und vierziger Jahre. Genau wie diese ist auch der Memex ein *Analoggerät*. Während der dreißiger Jahre, also auch zur Zeit der Entstehung von »As we may think« war diese Technologie der Stand der Technik. Außerdem hatte Bush mehr als zwanzig Jahre selbst zur Entwicklung dieser Technologie beigetragen und war deswegen tief in der Analogtechnik verwurzelt. Für Bush war dieser Typ von Technik deswegen so attraktiv, weil er ein ideales Hilfsmittel für die zu Beginn dieses Kapitels erläuterte Methodik des ingenieurmäßigen Entwurfs darstellte.

Modelle von physikalisch-technischen Systemen sind normalerweise durch einen Satz von Gleichungen beschrieben, die das Systemverhalten beschreiben, indem sie die physikalischen Größen des Systems miteinander verbinden. Beim Analogrechner ist es nun möglich, ein solches Modell unmittelbar zu berechnen, ohne auf die Verfahren der Arithmetik zurückgreifen zu müssen. Der *Differential Analyzer* und seine Nachfolger boten also die Möglichkeit, das Verhalten eines

¹ Crane 1991, S. 346.

² Bush 1945b, S. 105.

³ Lang 1996, S. 1; Ceruzzi 1989, S. 273.

⁴ Bushs *Rockefeller Differential Analyzer* wog 100 Tonnen und bestand aus 2 000 Elektronenröhren, 150 Motoren und einigen Tausend Relais. Vgl. Wildes and Lindgren 1985, S. 93. – Der erste elektronische Digitalrechner, der *ENIAC* von J. Presper Eckert und John W. Mauchly, hatte eine Grundfläche von 170 m², wog 30 Tonnen und bestand aus 17 468 Elektronenröhren, 1 500 Relais und über 6 000 manuellen Schaltern. Vgl. Flamm 1988, S. 48.

technischen Systems in vielerlei Aspekten nachzubilden. Die den Ingenieur interessierenden physikalischen Größen lassen sich dabei mit Hilfe von Maßstabsfaktoren sowohl in ihrer absoluten Größe als auch in ihrem zeitlichen Verlauf aus den Ausgangsgrößen des Analogrechners ablesen. Zur weiteren Veranschaulichung werden diese Ausgangsgrößen (z. B. Spannungen, Drehwinkel etc.) häufig in Diagramme umgewandelt.¹ Bush betrachtete seine Analogcomputer deswegen als Kombinationen von elektrischen und mechanischen Bauelementen, die sich genauso verhalten wie die sie darstellenden physikalisch-technischen Systeme.²

Im Gegensatz zum Analogcomputer werden mit Hilfe des Digitalrechners vor allem arithmetische Operationen durchgeführt, er ist von seinem Wesen her ein Hilfsmittel der *numerischen* Mathematik. Der Digitalrechner kann deswegen auch keine *kontinuierlichen* Eingabegrößen verarbeiten, sondern liefert nur Momentaufnahmen für diskrete Eingabewerte und Randbedingungen. Will man mit dem Digitalrechner die gleichen *Simulationsaufgaben* lösen wie mit einem Analogcomputer, so muß man eine hinreichend große Zahl solcher Momentaufnahmen erstellen und geeignet interpretieren. Beim Digitalrechner tritt damit eine weitere Abstraktionsebene hinzu, die zu Beginn der vierziger Jahre eher der Denkweise von Mathematikern als der von Ingenieuren entsprach.³

2.5.2 Der Mikrofilmspeicher

Als Speichermedium wollte Bush auf die Mikrofilmtechnik zurückgreifen, die er bereits für seinen *Rapid Selector* verwendet hatte und 1945 immer noch als vielversprechende analoge Speichertechnologie galt.⁴ Bush nahm an, daß sich die Auflösungsfähigkeit des Mikrofilms in absehbarer Zukunft um einen Faktor 100 verbessern lassen würde. Damit wären Verkleinerungen im Maßstab 1:10 000 möglich. Unter der Annahme, daß der Film nicht dicker als Papier ist, wäre das Informationsproblem von der reinen Menge her damit gelöst:

»The Encyclopædia Britannica could be reduced to the volume of a match-box. A library of a million volumes could be compressed into one end of a desk. If the human race has produced since the invention of movable type a total record ... having a volume corresponding to a billion books, the

¹ Johnson 1956, S. 2ff.

² Bush 1972, S. 181; Bush 1967, S. 83f.

³ Owens 1986, S. 75; Bush 1967, S. 83; Wiener 1948, S. 135ff.

⁴ Zuverlässige und preiswerte Speichermedien waren bis zur Entwicklung des Magnetkernspeichers durch Jay W. Forrester zu Beginn der fünfziger Jahre das entscheidende Probleme bei der Entwicklung leistungsfähiger elektronischer Digitalrechner. Es wurde endgültig erst mit der massenhaften Verfügbarkeit elektronischer Speicherchips Mitte der siebziger Jahre gelöst. Vgl. Pugh 1984; Gilder 1974.

whole affair, assembled and compressed, could be lugged off in a moving van.«¹

Bushs positive Einschätzung eines Mikrofilmspeichers basierte auf Entwicklungen der zwanziger und dreißiger Jahre. Bereits 1925 war ein Filmmaterial vorgestellt worden, das so feinkörnig war, daß der komplette Text der Bibel auf knapp 20 cm² Platz fand.² Durch diese Fortschritte schien sich die Möglichkeit zu eröffnen, Information universell zugänglich und für jedermann erschwinglich zu machen. Watson Davis und Edwin Slosson, die während der zwanziger Jahre das *American Documentation Institute* (später American Society for Information Science) gegründet hatten, propagierten diesen Gedanken schon seit 1926.³ Damit verbunden war auch die Hoffnung auf eine intellektuelle Revolution, wie sie der Schriftsteller H. G. Wells 1937 in seinen Vorträgen über das »Weltgedächtnis« formuliert hat. Dahinter steht die Vorstellung, daß die bessere Zugänglichkeit von Information für jedermann zu einer gerechteren Verteilung der Chancen und letztlich zu einem demokratischeren Staat führt.⁴

Die Datenkompression hätte aber gleichzeitig einen bestimmenden Einfluß auf die Kosten. Mit einer hochentwickelten Mikrofilmtechnik und einer hohen Auflage könnten die Kosten für ein Werk wie die *Encyclopædia Britannica* auf einen Bruchteil der bisherigen Kosten reduziert werden. Information könnte so theoretisch fast kostenlos werden.⁵ Bush ließ bei dieser Argumentation freilich außer acht, daß bei Informationsdienstleistungen nicht nur die Reproduktionskosten relevant sind.

Bush merkte aber gleichzeitig an, daß die Kompression und Speicherung von Information allein nicht genug ist. Sie muß auch wirklich verfügbar sein, das heißt sie muß einfach und schnell in den Memex eingegeben werden können. Vor allem aber müssen dem Benutzer Hilfsmittel zur Verfügung gestellt werden, um die gewünschten Informationen in den riesigen Datenbeständen möglichst schnell und vollständig wiederzufinden. Ist das Problem der Speicherung also durch Verwendung von Mikrofilm gelöst, so stellt sich als nächstes das Problem der Ein- und Ausgabe.

2.5.3 Eingabetechnik

Bush widmet deshalb den weitaus größten Teil seines Aufsatzes der Darstellung unterschiedlicher Ein- und Ausgabetechniken für unterschiedliche Datentypen. Er

¹ Bush 1945b, S. 103.

² Buckland 1992, S. 265f.

³ Nyce and Kahn 1991b, S. 49f.; Burke 1994, S. 114ff.; Schultz and Garwig 1969.

⁴ Wells 1938, S. 10ff., 63ff.

⁵ Bush 1945b, S. 103.



Abbildung 5. Wissenschaftler mit der Zyklophenkamera. Das weiße Rechteck kennzeichnet das in die Brille eingebaute Sucherfeld.

diskutiert dabei die Eingabe von Bildern, Sprache und gedruckter Schrift und die Ausgabe auf Mikrofilm, Papier sowie die Sprachausgabe.

Die Zyklophenkamera

Für die Eingabe von Fotos schlägt Vannevar Bush eine Mikrofilmkamera vor, die der Wissenschaftler während seiner Arbeit auf der Stirn tragen soll; diese Vorrichtung wurde in der *Life*-Version vom Herausgeber treffend als »Zyklophenkamera« bezeichnet (Abb. 5).¹

Bush selbst spricht von einer Kamera in der Größe einer Walnuß, mit der der Benutzer des Memex farbige Mikrofotos von 3×3 mm Größe aufnehmen kann. Die Bedienung dieser Kamera soll sich dabei auf die Auswahl des Bildausschnitts und die Betätigung des Auslösers beschränken. Der Sucher ist ein Rechteck aus feinen Linien, das in eine herkömmliche Brille integriert ist; der Auslöser wird am Ärmel oder Revers der Jacke befestigt, ist also mit einer Handbewegung zu erreichen. Alle anderen Funktionen der Kamera, Einstellen von Belichtungszeit, Blende und Schärfe entfallen bei der Zyklophenkamera oder werden automatisiert. Dabei dachte Bush an die Verwendung eines kurzbrennweitigen Universalobjektivs und einer Belichtungsautomatik auf der Basis einer eingebauten Photozelle.² Er griff dabei auf zwei technische Entwicklungen der dreißiger Jahre zurück.

Während der zwanziger Jahre waren in verschiedenen Forschungsinstitutionen in Europa und den Vereinigten Staaten Forschungsarbeiten zum inneren Photoeffekt von Halbleitern durchgeführt worden. In der Folge entwickelten unter anderem Bruno Lange und L. Bergmann 1931 die erste für die praktische Verwendung geeignete Selen-Photozelle.³ Gleichzeitig war Bush bei seiner Beschreibung

¹ Bush 1945a, S. 113.

² Bush 1945b, S. 102.

³ Morris 1990, S. 19. – In der heutigen Terminologie würde man nicht von Photozellen, sondern von Photoelementen sprechen.

vermutlich von den Miniaturisierungsentwicklungen im Kamerabau der dreißiger Jahre beeinflusst, als deren Höhepunkt die Entwicklung der berühmten *Minox* durch Walter Zapp im Jahre 1936 gilt. Diese Kamera hatte bereits einen eingebauten Belichtungsmesser und auch das von Bush beschriebene Universalobjektiv.¹

Für die Entwicklung der so aufgenommenen Bilder hoffte Bush auf ein trockenes Verfahren, das es ermöglichen würde, Fotos aufzunehmen und sofort anschauen zu können.²

Spracheingabe

Trotz der Möglichkeit zur Eingabe von Fotos mit Hilfe der Zyklopenkamera stellen schriftliche Informationen den weitaus größten Informationsbestand des Memex dar. Bei den 1939 üblichen Datenverarbeitungsmaschinen mußten diese Informationen erst umständlich, z. B. mit Hilfe eines Lochkartenstanzers, in eine maschinenlesbare Form gebracht und dann eingelesen werden. Bush regt in seinem Aufsatz an, bereits existierende Verfahren zur Sprachein- und -ausgabe zu kombinieren, um eine natürlichsprachige Kommunikation zwischen dem Memex und seinem Benutzer zu realisieren.

Dabei greift er explizit auf zwei Verfahren zurück, die seit Mitte der dreißiger Jahre bei den Bell Laboratorien entwickelt wurden, dort allerdings mit dem Ziel, die Verständlichkeit von Telefongesprächen über große Distanzen zu verbessern. Dabei handelte es sich einerseits um den sogenannten *Voder*, ein elektronisches Verfahren zur Erzeugung synthetischer Sprache, das bereits 1939 auf der Weltausstellung in New York als eines der publikumswirksamsten Ausstellungsobjekte der Bell Telephone Comp. öffentlich vorgestellt wurde. Bei diesen Vorführungen wurde ein Text mittels Tastatur in den *Voder* eingegeben; dieser gab den Text dann in verständlicher Sprache aus.³

Der *Vocoder* ist schließlich ein Gerät zur Analyse und Erkennung gesprochener Sprache, das ebenfalls während der dreißiger und vierziger Jahre von Homer Dudley bei den Bell Labs entwickelt wurde, allerdings mit dem Ziel, abhörsichere Telefonverbindungen während des Zweiten Weltkrieges zu ermöglichen. Beim *Vocoder* wird die über ein Mikrofon aufgenommene Sprache in verschiedenen Frequenzbändern analysiert. Die dabei ermittelten Parameter werden übertragen und beim Empfänger wieder zu einem natürlichsprachigen Signal zusammengesetzt. Bush dürfte als Direktor des O.S.R.D. auch über diese Entwicklungen, die als *Project X* der Geheimhaltung unterlagen, Kenntnis gehabt haben.⁴

¹ Gernsheim 1978, S. 1291f.

² Bush 1945b, S. 102f.

³ Bush 1945b, S. 103f.; Dudley 1940; Millman 1984, S. 99ff.

⁴ Bush 1945b, S. 104; Fagen 1978, S. 299ff. – Trotz der frühen Erfolge bei der Sprachsynthese und -verarbeitung bei den Bell Labs, die zu Bushs optimistischen Erwartungen geführt haben mögen,

2. Vision und Kontinuität: Vannevar Bushs Memex

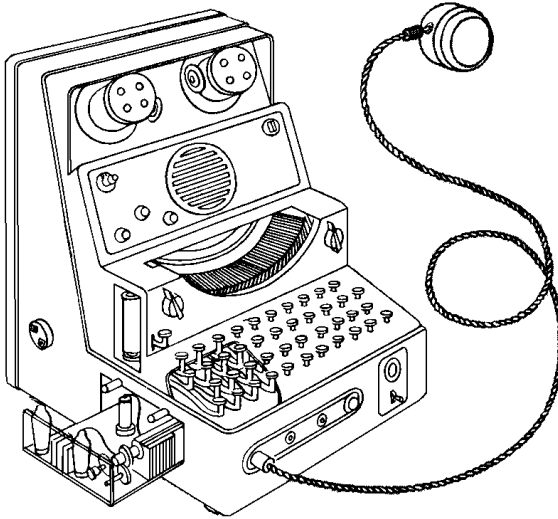


Abbildung 6. Supersekretär der Zukunft

Die Kombination der Verfahren soll zu einem Gerät führen, das eine automatische Übersetzung von gesprochener in geschriebene Sprache ermöglicht und vom Herausgeber des *Life*-Magazin als »Supersekretär der Zukunft«¹ titulierte.² Dieser Supersekretär ist nicht eigentlich Bestandteil, sondern eher ein *Zusatzgerät* zum Memex (Abb. 6).³ Mit seiner Hilfe, ergänzt durch eine drahtlose Übertragung, wird die Dateneingabe beim Memex extrem einfach. Bush selbst beschreibt diesen Vorgang folgendermaßen:

»One can now picture a future investigator in his laboratory. His hands are free, and he is not anchored. As he moves about and observes, he photographs and comments. Time is automatically recorded to tie the two records together. If he goes into the field, he may be connected by radio to his recorder. As he ponders over his notes in the evening, he again talks his comments into the record. His typed record, as well as his photographs, may both be in miniature, so that he projects them for examination.«⁴

ist die *Spracherkennung* nach wie vor ein hartes wissenschaftliches Problem in der KI-Forschung. Vgl. Kurzweil 1993, S. 263ff.

¹ Bush 1945a, S. 114.

² Bush merkte in seinem Artikel allerdings an, daß sich die normale Sprache nicht besonders für eine solche Kombination eignet. Er schlug daher die Verwendung einer Art Stenographie vor. Vgl. Bush 1945b, S. 104. – Das Problem der natürlichen Sprache als Mittel der Kommunikation zwischen Mensch und Computer ist auch ein zentrales Thema der Künstlichen Intelligenz. Vgl. Kapitel 5.

³ Bush 1945a, S. 114.

⁴ Bush 1945b, S. 103.

Einlesen gedruckter Vorlagen

Um auf den kompletten Bestand wissenschaftlicher Informationen zurückgreifen zu können, ist eine Eingabemöglichkeit für gedruckte Text- und Bildvorlagen besonders wichtig. Prinzipiell ist diese Art der Dateneingabe durch die Verwendung von Mikrofilm als Speichermedium besonders einfach. Auf der Gesamtansicht des Memex (Abb. 4, S. 54) sieht man auf der linken Seite eine transparente Scheibe, auf die die Papiervorlagen gelegt werden können. Auf Knopfdruck werden diese auf Mikrofilm abfotografiert und so zum Speicher des Memex hinzugefügt.¹

Bush zieht aber auch eine maschinelle Erkennung des einzulesenden Textes in Erwägung und baut dabei wiederum auf die Fortschritte bei der Entwicklung von Photozellen: »Machines have been made which will read typed figures by photocells and then depress the corresponding keys ... «² Die Entwicklungen in diesem Bereich befanden sich allerdings zur Entstehungszeit von »As we may think« noch in ihren Anfängen.³

2.5.4 Assoziative Datenspeicherung

Die eigentliche Datenausgabe machte aufgrund der analogen Speicher- und Verarbeitungstechnologie keine größeren Probleme. Die auf Mikrofilm gespeicherten Informationen sollten mit Hilfe einer geeigneten Optik auf die beiden Bildschirme projiziert werden. Ein sehr viel größeres Problem war in Bushs Augen das Auffinden der gewünschten Information im riesigen Speicher des Memex, der ja im günstigsten Fall die komplette Information der Menschheit beinhalten sollte.

»The prime action of use is selection, and here we are halting indeed. There may be millions of fine thoughts, and the account of the experience on which they are based, all encased within stone walls of acceptable architectural form; but if the scholar can get at only one a week by diligent search, his syntheses are not likely to keep up with the current scene.«⁴

Dieses Problem läßt sich in zwei Teilaspekte unterteilen: die technischen *Verfahren* zur Informationswiedergewinnung (engl. *information retrieval*) und eine möglichst effektive *Datenorganisation*. Das erste Problem kann durch die Verwendung der für den *Rapid Selector* entwickelten Verfahren gelöst werden. Für die Datenorganisation schlägt Bush eine Art assoziative Speicherung vor. Das

¹ Bush 1945b, S. 107.

² Bush 1945b, S. 104.

³ Weaver 1969; Nagy 1968, S. 854.

⁴ Bush 1945b, S. 105.

2. Vision und Kontinuität: Vannevar Bushs Memex

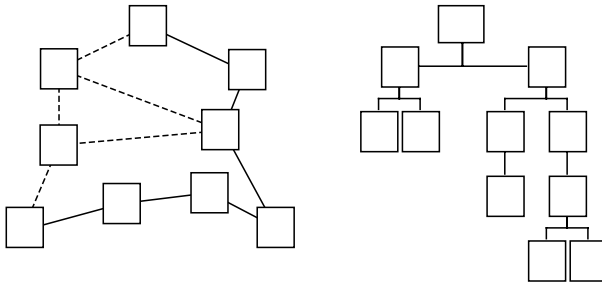


Abbildung 7. Assoziative Datenorganisation mit zwei Argumentationspfaden (links) und fünfstufige hierarchische Datenorganisation (rechts)

Vorbild ist hierbei die Art und Weise, wie der Mensch auf sein Gedächtnis zurückgreift.¹ Der Vorschlag eines assoziativen Datenzugriffs resultiert aber nicht in erster Linie aus einer Vorliebe Bushs für anthropomorphe Technik, sondern viel eher aus seiner Unzufriedenheit über existierende hierarchische Verfahren zur Datenselektion mittels Indizierung (vgl. Abb. 7). Er schreibt:

»The real heart of the matter of selection, however, goes deeper than a lag in the adoption of mechanisms by libraries, or a lack of development of devices for their use. Our ineptitude in getting at the record is largely caused by the artificiality of systems of indexing. When data of any sort are placed in storage, they are filed alphabetically or numerically, and information is found (when it is) by tracing it down from subclass to subclass. It can be in only one place, unless duplicates are used; one has to have rules as to which path will locate it, and the rules are cumbersome. Having found one item, moreover, one has to emerge from the system and re-enter on a new path.

The human mind does not work that way. It operates by association. With one item in its grasp, it snaps instantly to the next that is suggested by the association of thoughts, in accordance with some intricate web of trails carried by the cells of the brain. It has other characteristics, of course; trails that are not frequently followed are prone to fade, items are not fully permanent, memory is transitory. Yet the speed of action, the intricacy of trails, the detail of mental pictures, is awe-inspiring beyond all else in nature.«²

Bei seinen Vorschlägen zur technischen Realisierung mußte Bush allerdings sein assoziatives Konzept teilweise wieder zurücknehmen. Dazu erläutert er, wie ein Benutzer einen Argumentationspfad (*trail*) in den Memex eingibt, der die

¹ Diese zentrale Überlegung hat dem ganzen Aufsatz seinen Titel gegeben, denn die Daten sollen so organisiert werden »as we may think«.

² Bush 1945b, S. 106.

Überlegenheit des türkischen Kurzbogens über den englischen Langbogen zur Zeit der Kreuzzüge zum Thema hat (Abb. 8).

Der Besitzer des Memex projiziert zunächst zwei zur Argumentation gehörende, inhaltlich aber eventuell vollkommen divergierende Seiten, z. B. aus einer Enzyklopädie, auf die Bildschirme des Memex.¹ Zu den angezeigten Seiten kann er Kommentare eingeben und diese dem Hauptpfad zuordnen oder einem weniger wichtigen Seitenpfad.² Ein solcher Pfad kann, wenn er einmal in den Memex eingegeben wurde, aus technischen wie auch aus logischen Gründen nicht wieder gelöscht werden. Einerseits bietet die Mikrofilmtechnik keine einfache Möglichkeit zur Löschung *einzelner* Elemente auf einem einmal belichteten Dokument. Schwerer wiegt allerdings die Tatsache, daß Dokumente nicht ohne weiteres aus dem Gewebe der Dokumente entfernt werden können, ohne daß die Gefahr bestünde, das Dokument auch aus *anderen* Argumentationspfaden zu entfernen und somit eine Inkonsistenz des Datenbestandes in Kauf zu nehmen.³ Diese Randbedingung schränkt das Konzept der assoziativen Speicherung allerdings nicht wesentlich ein, da beliebig viele Pfade angelegt werden können.

Die eigentliche Verbindung zwischen den Dokumenten wird aber durch die Eingabe eines bestimmten *Codes* über die Tastatur auf der rechten Seite des Memex vorgenommen, mit dem der Benutzer des Memex die Wichtigkeit und die Art der Verbindung zwischen den Dokumenten eingibt. Dieser Code wird dann als Punktcombination in dem dafür vorgesehenen Feld auf dem Mikrofilm abgespeichert.⁴

Bush hätte also bei der Realisierung seines assoziativen Speichers auf die geschmähten Verfahren der Klassifizierung und Indizierung zurückgreifen müssen. Er führt damit ein Verfahren ein, das bei der Entwicklung des Personal Computing besonders wichtig ist: die Trennung der Funktionalität, die durch die zu bearbeitende Aufgabe und die Qualifikation des Benutzers bestimmt ist, von ihrer konkreten technischen Realisierung. Diese Vorgehensweise wird Bushs Aufsatz teilweise zum Vorwurf gemacht. So behauptet Michael Buckland, Bushs Ablehnung von maschinennahen Indizes und die gleichzeitige Verwendung von Indizes

¹ Die nähere Betrachtung der Abbildung von A. D. Crimi vermittelt allerdings den Eindruck, es handle sich bei den dargestellten Personen eher um Irokesenkrieger als um englische oder türkische Soldaten. Die Wahl des Beispiels leitet sich aus Bushs (wissenschaftlichem) Interesse an der Technik des Bogens zu Beginn der vierziger Jahre her. 1941 hatte er sogar einen entsprechenden Forschungsauftrag an einen Physiker an der University of Illinois vergeben. Vgl. Zachary 1997, S. 147ff.

² Abbildung 8 zeigt die handschriftliche Eingabe eines solchen Kommentars. In der Übersichtszeichnung (Abb. 4, S. 54) ist jedoch nicht ersichtlich, wie handschriftliche Eingaben vom Projektionschirm in den Memex eingelesen werden können.

³ Nelson 1980.

⁴ Bush 1945b, S. 107.

2. Vision und Kontinuität: Vannevar Bushs Memex

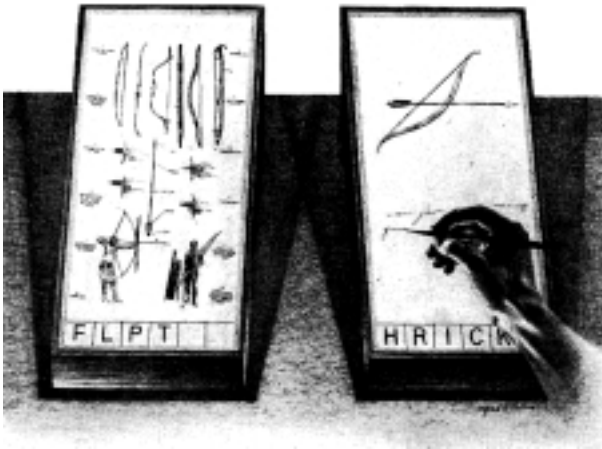


Abbildung 8. Memex während der Benutzung

zur Realisierung eines für den Benutzer assoziativ *wirkenden* Speichers sei inkonsequent und unoriginell.¹ Dabei geht es Bush doch offensichtlich um die einfache und intuitive *Bedienbarkeit* des Memex und weniger um die dazu verwendeten Verfahren. Die Ikonenrolle, die »As we may think« für Teilbereiche des *Human Factors Engineering* einnimmt, ist also durchaus berechtigt, weil der Aufsatz eine der frühesten Formulierungen eines aufgaben- und benutzerorientierten Systementwurfs im Bereich der Informationstechnik darstellt.

2.5.5 Verwendung des Memex

Im Gegensatz zum menschlichen Gedächtnis verblasen die Argumentationpfade, die einmal in den Memex eingegeben wurden, nicht im Laufe der Zeit, sondern sind jederzeit wieder abrufbar. So ist es auch nach langer Zeit ohne Schwierigkeit möglich, einen Argumentationspfad abzurufen und nachzuvollziehen. Durch die Speicherung auf Mikrofilm ist es aber auch problemlos möglich, eigene *trails* an Dritte weiterzugeben. Bush führt dazu wieder sein Bogen-Beispiel an:

»Several years later, [the user's] talk with a friend turns to the queer ways in which a people resist innovations, even of vital interest. He has an example, in the fact that the outranged Europeans still failed to adopt the Turkish

¹ Buckland 1992, S. 285. Buckland verweist dabei auf ein dem Memex bzw. dem Rapid Selector ähnliches Gerät, das Anfang der dreißiger Jahre von Emanuel Goldberg bei der Zeiss IKON AG in Dresden entwickelt worden war, aber längst nicht die Publizität des Memex erlangte.

bow. In fact he has a trail on it. A touch brings up the code book. Tapping a few keys projects the head of the trail. A lever runs through it at will, stopping at interesting items, going off on side excursions. It is an interesting trail, pertinent to the discussion. So he sets a reproducer in action, photographs the whole trail out, and passes it to his friend for insertion in his own memex, there to be linked into the more general trail.«¹

Durch diesen Informationsaustausch entsteht mit der Zeit eine Art Weltenzyklopädie, wie sie Utopisten wie H. G. Wells schon während der dreißiger Jahre als Grundlage einer globalen, auf den Wissenschaften basierenden Gesellschaft gefordert hatten.² Im Gegensatz zu Wells' Vorstellungen von einer einheitlichen, zentral erstellten Enzyklopädie, spiegelt Bushs Entwurf auch eine typisch amerikanische Vorliebe für Dezentralität wieder. Der Datenbestand eines jeden Memex würde nach und nach organisch wachsen, er würde Redundanzen, eventuell sogar Widersprüche enthalten, da das Löschen von einzelnen Daten oder Teilen eines Argumentationspfades nicht vorgesehen ist. Er wäre damit auch ein Beispiel für die amerikanische Vorstellung von Meinungsppluralität und Bushs Überzeugung, daß durch das bloße Aufsummieren von subjektiven Datenbeständen automatisch Intersubjektivität oder gar Objektivität entstünde.³ Es bleibt bei diesem System allenfalls einzuwenden, daß es ihm an einer gewissen Flexibilität mangelt, weil sich Veränderungen in den Einstellungen und Bewertungen des Benutzers nicht im Memex abspeichern lassen, ohne einen vollständig neuen Pfad anlegen zu müssen.

Bush sieht neben dieser naheliegenden Verwendung des Memex für die individuelle Verwaltung der Daten eines Wissenschaftlers aber auch andere, neuartige Einsatzmöglichkeiten für den Memex. So schildert er in leuchtenden Farben neue Typen von Enzyklopädien und ihre Bedeutung für die Allgemeinheit:⁴

- Im Bereich der Justiz lassen sich Enzyklopädien mit allen Gesetzestexten, Auslegungen und Präzedenzfällen anlegen. Gerade in Anbetracht des anglo-amerikanischen Rechtssystems mit seiner großen Bedeutung von Präzedenzfällen wäre dies eine nicht unerhebliche Erleichterung der Arbeit von Anwälten. Auch im Bereich des Patentrechts brächte der unmittelbare Zugang auf die Vielzahl der erteilten Patente erhebliche Arbeitersparnis, vor allem bei der Überprüfung der Patentfähigkeit einer Erfindung.

¹ Bush 1945b, S. 107.

² Richards 1988, S. 302.

³ Vgl. dazu Bushs gleichnishafte Darstellung des wissenschaftlichen Prozesses in seinem Aufsatz »The Builders« (1946). – Norbert Wiener war einer der wenigen zeitgenössischen Wissenschaftler, die dieser Auffassung vehement widersprachen. Vgl. Wiener 1948, S. 184f.; Wiener 1957, S. 24.

⁴ Bush 1945b, S. 108.

2. Vision und Kontinuität: Vannevar Bushs Memex

- Medizinische Enzyklopädien unterstützen den Arzt bei seiner Diagnose und bei der Auswahl von geeigneten Therapien und Medikamenten, insbesondere wenn es sich um seltene Krankheitsbilder handelt.
- Im Bereich der Geschichtswissenschaften eignet sich eine assoziativ organisierte Enzyklopädie besonders gut, um die komplexen Zusammenhänge historischer Ereignisse und Vorgänge adäquat abzubilden.

In diesem Zusammenhang erwartet Bush auch die Belebung der Produktivität und die Entstehung einer neuen Berufsgruppe, der *trail blazers*, die sich mit der Aufarbeitung von Rohdaten für bestimmte Anwendungen beschäftigt.¹ Heute würde man von Informationsdienstleistungen sprechen, die sich in der Tat seit Ende des Zweiten Weltkrieges stark entwickelt haben.

An dieser Stelle scheint es nun auch angebracht, die hochfliegenden Erwartungen Bushs und das Potential des Memex kritisch unter die Lupe zu nehmen. Bushs Vorstellungen über die Veränderung der wissenschaftlichen Arbeit durch den Memex stellen im Grunde genommen den seit Mitte des 19. Jahrhunderts gewachsenen Wissenschaftsbetrieb in Frage. Die individuelle Anhäufung von Fachwissen und die zunehmende Spezialisierung sind schließlich wichtige Gründe für die Existenz eines ausdifferenzierten und vielfach bürokratisch organisierten Wissenschaftsbetriebs. Solche Institutionen, zumal wenn sie über Generationen gewachsen sind und ihren Mitgliedern Sicherheit und Ansehen bieten, setzen jedem – auch noch so berechtigten – Reformvorhaben einen nicht zu unterschätzenden Widerstand entgegen.

Bush hoffte allerdings, daß sich Interdisziplinarität automatisch durch den Abbau von Barrieren zwischen den einzelnen Spezialdisziplinen entwickelt. Dazu soll der einfache Zugriff auf Informationen anderer Disziplinen durch die assoziative Datenorganisation beitragen. Dieser Teil von Bushs Darstellung ist deutlich weniger pragmatisch als der Rest des Artikels. Mit einem fast religiös zu nennenden Unterton tritt er für die Fortentwicklung der menschlichen Zivilisation durch die Kraft der Technik ein.² Bush läßt dabei aber außer acht, daß Interdisziplinarität auch die Bereitschaft beinhalten muß, sich mit der fremden Disziplin auseinanderzusetzen, ihre Prämissen und Arbeitsweisen zu verstehen. Die reine Verfügbarkeit von Information führt deshalb noch lange nicht zu einem besseren *Verständnis* der Zusammenhänge. Da der Memex diesen Prozeß nicht explizit unterstützen kann, müßten neue Instanzen geschaffen werden, die durch eine geeignete Aufbereitung der Daten das Überschreiten dieser unsichtbaren Grenze erleichtert.

¹ Bush 1945b, S. 108.

² Bush 1965.

Für sich allein betrachtet, unterstützt der Memex eher die entgegengesetzte Entwicklung: Durch die Zugänglichkeit von Information könnte eine weitere Spezialisierung gefördert werden, schlimmer noch, es könnte sich eine Ignoranz gegenüber allem Wissen entwickeln, das nicht unmittelbar mit der eigenen Forschung zusammenhängt. Insofern geht Bush mit seinem visionären Entwurf einerseits nicht weit genug, da er unkritisch annimmt, die schiere Verfügbarkeit von Wissen sei schon positiv für die Wissenschaft und die Allgemeinheit. Andererseits geht er nicht weit genug, weil eine Welt, in der jeder die Grenzen seines Spezialgebiets scheinbar mühelos überwinden kann, eine völlig andere, aber gewiß nicht weniger komplexe Welt wäre.

2.6 »The day is not yet here«: Das Schicksal des Memex

Obwohl »As we may think« von nachfolgenden Generationen als Vannevar Bushs wichtigster Beitrag zur *Information Science* betrachtet wurde, stand der Memex bei ihm selbst nach 1945 nicht ganz oben auf der Tagesordnung. Als er von J. Burchard, dem Leiter der MIT Bibliothek, gebeten wurde, vor der nationalen Versammlung der Bibliothekare über den Memex zu sprechen, antwortete er:

»I am now weary and the thought of trying to present this subject to a group of librarians appals me. It was all right to stir them up with the article on Memex, for they could not possibly tell whether I was being serious or not and besides it was an arm's length affair anyway. To appear directly on the same subject before a group of librarians would be quite an undertaking.«¹

In den Jahren nach 1945 war Bush hauptsächlich mit politischen Themen beschäftigt. Erst nach seinem Ausscheiden als Präsident der *Carnegie Institution* im Jahre 1955 und als Vorsitzender der *MIT Corporation* im Jahre 1959 konnte er sich wieder mit Fragen der *Information Science* und des Memex beschäftigen, zumal Ende der fünfziger Jahre offensichtlich wurde, daß der Digitalcomputer die entscheidende Rolle in der Informationsverarbeitung spielen würde.²

So entstanden zwischen 1959 und 1970 drei weitere Texte, die sich mit dem immer noch ungelösten Informationsproblem und dem Memex beschäftigten.³ Dabei kann man ein Festhalten Bushs an der Idee der persönlichen Maschine und des assoziativen Datenzugriffs, aber ein nachlassendes Interesse an Fragen der

¹ Brief von Bush an Burchard, 19. Juni 1947, zitiert nach Burke 1994, S. 329.

² Kahn and Nyce 1991, S. 119; Owens 1996.

³ Bush 1991a; Bush 1967; Bush 1972, S. 185–195.

2. Vision und Kontinuität: Vannevar Bushs Memex

konkreten Konstruktion eines Memex feststellen.¹ Im Gegensatz zu »As we may think« fand aber keiner der späten Aufsätze ein so großes und aufnahmefähiges Publikum; der explizit als Nachfolger von »As we may think« geplante Aufsatz »Memex II« aus dem Jahre 1959 wurde sogar überhaupt nicht veröffentlicht.

Trotzdem waren für Bush in seinen späten Memex-Essays die technischen Möglichkeiten der weiteren Miniaturisierung von großer Bedeutung. Er diskutierte die Vorteile, die eine Abkehr von der Mikrofilmtechnik und die Verwendung magnetischer Speichermedien bieten. Vor allem das schnelle und problemlose Einfügen und Löschen von Daten machten das Magnetband für ihn zu einem idealen Ersatz für den Mikrofilmspeicher des Memex.²

In den drei Texten bringt Bush aber auch seine Abneigung gegen den Digitalcomputer nochmals zum Ausdruck. Dieser war zwar in den sechziger Jahren schon sehr leistungsfähig und zuverlässig, Bush zog aber weiterhin die einfachere und preiswertere, vor allem aber für die Aufgaben des Ingenieurs völlig ausreichende Analogtechnik, etwa des *Rapid Selector*, vor. Dabei erkannte er sehr wohl an, daß gerade die symbolverarbeitenden Fähigkeiten den Digitalrechner zu einem mächtigen Werkzeug machen konnten. Für Aufgaben, wie sie der Memex erfüllen sollte, waren ihm die zeitgenössischen Rechner zu teuer, zu kompliziert in der Bedienung und zu sehr auf die Verarbeitung großer Zahlenmengen zugeschnitten.³ Obwohl Bush vermutete, daß die Idee des Memex letztlich doch mit Hilfe des Digitalrechners realisiert würde, so blieb doch die dahinterstehende Idee zutiefst der Analogtechnik verhaftet.⁴ Zusammenfassend äußerte sich Bush über die Zukunft des Memex und die seiner Idee einer persönlichen Maschine zur Unterstützung der menschlichen Informationsverarbeitung:

»No memex could have been built when that article [As we may think] appeared. In the quarter-century since then, the idea has been with me almost constantly, and I have watched new developments in electronics, physics,

¹ Obwohl alle drei Aufsätze Technologien aufgreifen, die an der Spitze der Entwicklung standen (Laser, Magnetband, Integrierte Schaltkreise), läßt sich erkennen, daß sich Bush aus der unmittelbaren Technikentwicklung zurückgezogen hatte. Statt dessen diskutierte er in kulturphilosophischer Weise die ethisch-moralischen Implikationen der fortgeschrittenen Informationstechnik.

² Bush 1991a, S. 167.

³ Bush 1991a, S. 165f., 169; Bush 1967, S. 82f.

⁴ Bush 1972, S. 191. Bush schreibt weiter: »Who invented the digital computer? I can write at once that I did not, in fact I had little to do with that whole development«, S. 185. Tatsache ist aber, daß sich Bush in den dreißiger Jahren zumindest ansatzweise mit der Digitaltechnik beschäftigt hat. Das MIT hätte damit eine vergleichsweise gute Ausgangsposition für die Entwicklung eines Digitalcomputers gehabt, wäre die Forschergruppe unter Bushs Schülern W. H. Radford und Wilcox Overbeck nicht 1941 kriegsbedingt aufgelöst worden. Vgl. dazu Bush 1936; Wildes and Lindgren 1985, S. 229ff. und Randell 1982b, S. 337ff.

chemistry, and logic to see how they might help to bring it to reality. That day is not yet here, but has come far closer . . . «¹

Bush selbst war es nie vergönnt, einen seiner visionären Entwürfe – abgesehen vom *Rapid Selector* mit seinen eingeschränkten Möglichkeiten – in die Realität umzusetzen. Aber er beeinflusste mit seinen Schriften eine neue Generation von Wissenschaftlern, die Bushs Visionen aufnahmen und mit Hilfe des Digitalrechners zu realisieren versuchten.

2.7 Das Vermächtnis des Memex

Es wurde bereits erwähnt, daß es nach 1945 keine ernsthaften Anstrengungen zur Entwicklung des Memex gab. Dies bedeutet allerdings nicht, daß Bushs Idee erst mit der Entwicklung leistungsfähiger Digitalrechner wieder aufgenommen wurde. Tatsächlich wurde Bushs *Rapid Selector*, der das Kernstück des Memex hätte werden sollen und der vor dem Krieg nur als Prototyp existiert hatte, nach 1945 weiterentwickelt und während der fünfziger Jahre im amerikanischen Bibliothekswesen verwendet. Ralph R. Shaw, ein Mitarbeiter der *Agricultural Library*, setzte sich seit 1947 für den Einsatz des *Rapid Selector* und ähnlicher, von den *Engineering Research Associates* (ERA) entwickelter Geräte ein und wurde dabei durch das *Office of Technical Service* (OTS) des amerikanischen Wirtschaftsministeriums unterstützt.² Der große Erfolg war aber auch diesen Maschinen nicht beschieden. Sie wurden, jeweils nur für eine kurze Zeit, vom *National Bureau of Standards*, von der *Central Intelligence Agency* und vom *Bureau of Ships* der U. S. Navy verwendet. In den frühen sechziger Jahren wurden auch die letzten *Rapid Selectors* durch fortgeschrittenere Geräte, etwa IBMs *Walnut*, ersetzt.³

Eine Fortsetzung dieser auf Bibliotheksbelange reduzierten Entwicklungstätigkeit stellte – mit einer Kombination von Digital- und Mikrofilmtechnik – zwischen 1965 und 1973 das Projekt Intrex am MIT dar, das auf Vorarbeiten von Joseph C. R. Licklider basierte und Bushs Idee einer automatisierten Bibliothek neu belebte.⁴ Licklider hatte 1960 den einflußreichen Aufsatz »Man-Computer Symbiosis« geschrieben, der viele von Bushs Ideen aufgriff und in Beziehung zur

¹ Bush 1972, S. 190.

² Engineering Research Associates 1983, S. 230ff; Anonymous 1949a.

³ Burke 1991, 159ff.; Shaw 1949; Bradshaw 1962.

⁴ Overhage and Reintjes 1974, siehe auch Kapitel 4, Seite 119.

2. Vision und Kontinuität: Vannevar Bushs Memex

Computertechnik setzte.¹ Licklider wurde während der sechziger Jahre als Direktor des *Information Processing Techniques Office* (IPTO) der *Advanced Research Projects Agency* (ARPA) zu einem der wichtigsten Förderer der Projekte, die wesentlich zur Realisierung des *personal computing* beigetragen haben.

Obwohl die unmittelbaren Folgen von »As we may think« also vergleichsweise gering blieben, hatte er doch mittelfristig einen enormen Einfluß. Bushs Aufsatz erschien, ebenso wie sein Bericht »Science – The Endless Frontier«, im Juli 1945. Der Krieg in Europa war zu Ende und die Wissenschaftler, die in den vorangegangenen vier Jahren ihren Beitrag zur Kriegsführung geleistet hatten, stellten sich die Frage nach Ziel und Organisation der Nachkriegsforschung. In dieser Situation lieferte Bush mit beiden Texten zukunftsweisende Impulse. So wie »Science – The Endless Frontier« bestimmend für die Organisation der amerikanischen Nachkriegsforschung wurde, hatte »As we may think« eine inspirierende Wirkung auf eine ganze Generation von Computer-Wissenschaftlern. Unter ihnen war auch Douglas C. Engelbart, der Bushs Aufsatz las, als er 1945 in Südostasien stationiert war und auf seine Demobilisierung wartete (vgl. Kap. 5). Er nahm mit einer zeitlichen Verzögerung von fünfzehn Jahren die von Bush vorgebrachten Ideen auf und entwickelte sie mit seinem Team am Stanford Research Institute in Menlo Park, Ca. weiter.²

Was ist also von Bushs ambitioniertem Entwurf geblieben, nachdem doch alle seine technischen Realisierungsvorschläge im Laufe von nur wenigen Jahren obsolet wurden? Es sind vor allem die *Ideen* Bushs über die Möglichkeiten und Einsatzbereiche einer informationsverarbeitenden Maschine und vor allem die Einschätzung der Bedeutung von Information und Wissen für die zukünftige Wissenschaft, Wirtschaft und Gesellschaft. Im einzelnen sind dabei zu nennen:

- Die Idee der *persönlichen Maschine*: Indem Bush erstmals von der Vorstellung einer zentralen Datenverarbeitungsmaschine abrückt, wie sie von den 1945 existierenden Maschinen nahegelegt wurde, formuliert er den typischen *Hands-On Imperative* des *Personal Computing*.
- Der Memex, wie ihn Bush beschreibt, dient zur Unterstützung *alltäglicher Aufgaben*. Er ist somit ein Werkzeug, das in keinem (Wissenschaftler-)Haushalt fehlen darf.³ Damit verbunden sind die Forderungen einer preiswerten Produktion zuverlässiger Geräte.

¹ Licklider war allerdings nach eigener Aussage nicht direkt von Bushs Schriften beeinflusst. Ein Kontakt zwischen Bush und Licklider kam erst mit Beginn des Project Intrex 1965 zustande. Vgl. Licklider 1988a; Licklider 1965, Vorwort.

² Brief von Engelbart an Vannevar Bush, 24. Mai 1962. Stanford University Libraries, Engelbart Collection, Box 6, Folder 15 (künftig SUL/EC, Box-Folder).

³ Angesichts der 1945 verbreiteten Einschätzung, wenige Computer seien für die Lösung aller weltweit existierenden Aufgaben völlig ausreichend, handelte es sich hierbei um eine sehr spekulative, vielleicht aber auch weitsichtige Erwartung. Vgl. Cohen 1998.

- Eng mit dieser Idee verbunden ist die Vorstellung einer intuitiv zu bedienenden *Mensch-Maschine-Schnittstelle*. Damit formuliert Bush auch eines der Hauptziele des *Human-Factors Engineering*.
- Schließlich betont Bush die Bedeutung von *Information und Wissen* ganz im Sinne des einleitenden Zitats von Richard W. Hamming (siehe Seite 35) und formulierte damit ein Gegenbild zu der um 1945 verbreiteten Vorstellung des Computers als Rechenautomat.

Aber auch jenseits der die Maschine selbst betreffenden Eigenschaften hat Bush *methodische* Vorschläge von langfristiger Bedeutung gemacht:

- Die *Trennung von Funktionalität und technischer Realisierung*. Diese bei Bush noch sehr unscharfe Vorstellung ist schließlich die Hauptmotivation bei der Entwicklung von grafischen Benutzungsoberflächen gewesen.
- Indem Bush dem abstrakten Entwurf der Mathematiker Alan Turing oder (in geringerem Maße) John von Neumann¹ einen funktionalen Entwurf der informationsverarbeitenden Maschine entgegensetzte, brachte er eine originär *ingenieurmäßige Methodik* in die Computerentwicklung ein, die sich allerdings nur zögernd durchsetzen konnte.

Diese Ideen wurden seitdem immer wieder aufgegriffen, modifiziert und uminterpretiert. Die Realisierung einzelner Aspekte griff dabei in immer stärkerem Maße auf die immer leistungsfähigeren Digitalcomputer zurück und befand sich meist an der Spitze der technischen Entwicklung.

Bushs Hauptverdienst ist, daß »As we may think« durch seine mitreißende Rhetorik und seine persönliche wissenschaftliche Reputation eine weitreichende stimulierende Wirkung hatte. Bushs Aufsatz kam, wie R. A. Fairthorne bereits 1958 feststellte, zur rechten Zeit und öffnete den Leuten die Augen und Brieftaschen.²

¹ Tatsächlich nimmt John von Neumann eine Art Zwischenposition zwischen dem Mathematiker Turing und den Ingenieuren unter den Computerpionieren ein. Obwohl der Computer für von Neumann immer in erster Linie eine *Rechenmaschine* blieb, hatte er dennoch konkrete, gar ingenieur-nahe Anwendungen im Kopf. Ein Beispiel für eine solche Anwendung ist der sogenannte digitale Windtunnel, den von Neumann Ende der vierziger Jahre im Zusammenhang mit seinen Arbeiten zur numerischen Lösung von Strömungsproblemen anregte. Vgl. dazu Neumann 1955 und Hoßfeld 1996, S. 3ff.

² Fairthorne 1958, S. 36.

3. Vom Analogrechner zum Minicomputer

Of what use is a new-born baby?

*Benjamin Franklin*¹

So groß auch die von Vannevar Bush 1945 formulierte Vision war, es fehlten doch zunächst die technischen Möglichkeiten zu ihrer Realisierung. Für eine Reihe von Jahren war nicht abzusehen, ob sich die analoge Rechentechnik in der Tradition des *Differential Analyzer* tatsächlich überlebt hatte und ob sich der elektronische Digitalrechner für die Zwecke der Informations- und Symbolverarbeitung eignete.² Auch bei der Entwicklung des Digitalcomputers zwischen 1945 und 1960 bedurfte es erst einiger unkonventioneller Geister, deren Vorstellung über die Verwendung des Computers über die althergebrachte Verwendung von Rechen- und Lochkartenmaschinen hinauswies und damit das Fundament für seine breite Diffusion in alle Lebensbereiche legten.

Nach der Fertigstellung des ersten elektronischen Digitalcomputers, des *Electronic Numerical Integrator and Computer* (ENIAC) an der *Moore School of Electrical Engineering* in Philadelphia zur Jahreswende 1945/46 und der Veröffentlichung der sogenannten Von-Neumann-Architektur³ bildeten sich sehr schnell drei unterschiedliche Entwicklungsrichtungen heraus, die unterschiedliche Einsatzbereiche des Computers im Auge hatten. J. Presper Eckert (1919–1995) und John Mauchly (1907–1980) verließen im Frühjahr 1946 die Moore School und gründeten die *Eckert-Mauchly Computer Corporation*, um die ihnen erteilten Patente kommerziell auszuwerten. Dabei hatten sie den Einsatz des Computers in

¹ Zitiert nach Redmond and Smith 1980, S. 147ff. – Erwidern Benjamin Franklins, als dieser nach der Durchführung der ersten Ballonflüge gefragt wurde, welchen praktischen Nutzen diese Flüge hätten.

² Edwards 1996, S. 66ff.; Owens 1996.

³ Neumann 1945 – Bei diesem Dokument handelte es sich zwar um einen der militärischen Geheimhaltung unterliegenden Bericht, dessen Inhalt aber durch die engen Kontakte innerhalb der noch kleinen Gruppe von Computerentwicklern schnell allgemein bekannt wurde. Vgl. Flamm 1988, S. 224f.; Stern 1980; Tomayako 1983.

3. Vom Analogrechner zum Minicomputer

großen Unternehmen und staatlichen Institutionen wie dem amerikanischen Büro für Volkszählung im Auge, wo der Computer die etablierte Technologie der Lochkartenmaschinen ablösen sollte.¹

John von Neumann (1903–1957), der in engem Kontakt mit der Entwicklergruppe in Philadelphia gestanden hatte, hatte bereits 1945 ein eigenes Computerprojekt am *Institute for Advanced Study* (IAS) in Princeton initiiert, das die Weiterentwicklung des Computers als mathematisches Instrument zum Ziel hatte. Aufgrund der Reputation von Neumanns und seiner zahlreichen Publikationen fand die Architektur des IAS-Computers große Verbreitung beim Entwurf und Bau von Computern für wissenschaftliche Anwendungen.²

Die von den Ingenieuren Jay W. Forrester (* 1918) und Robert R. Everett (* 1921) geleitete Gruppe am MIT entwickelte mit dem Whirlwind-I-Computer schließlich den ersten elektronischen Digitalcomputer für Echtzeitaufgaben. Aus dieser Entwicklungslinie gingen nicht nur die sogenannten Prozeßrechner, sondern Ende der fünfziger Jahre auch die ersten interaktiven und grafikorientierten Computertypen als Vorläufer moderner Personal Computer hervor. Deshalb wird sich dieses Kapitel auf die Entwicklungsgeschichte dieses Computertyps beschränken.

Am MIT wurde weiterhin die von Bush propagierte Tradition des anwendungsorientierten Entwurfs gepflegt. Gleichzeitig machte sich der Einfluß der Kybernetik bemerkbar, einer wissenschaftlichen Disziplin, die sich nach dem Ende des Krieges im Umfeld des MIT und der Harvard University rapide entwickelte.

3.1 Die kybernetische Revolution

Obwohl die Kybernetik einer der umstrittensten Wissenschaftszweige des 20. Jahrhunderts sein dürfte, so hat sie doch nach dem Zweiten Weltkrieg einen erheblichen Einfluß auf die Entwicklung einer großen Zahl von wissenschaftlichen Disziplinen ausgeübt. Die Ökologie und die kognitive Psychologie verwenden ebenso kybernetische Methoden wie die Nachrichten- und Regelungstechnik und die sogenannte »Künstliche Intelligenz«.³ Für die Entwicklung des *Personal Computing* waren vor allem die Verallgemeinerung des Rückkopplungsprinzips

¹ Stern 1981; Pugh and Aspray 1996, S. 11.

² Offizielle Nachbauten des IAS-Computers entstanden unter anderem in den Forschungszentren von Los Alamos (MANIAC, 1952) und Oak Ridge (ORACLE, 1953) sowie bei der RAND Corporation (JOHNNIAC, 1954). Außerdem gab es eine beachtliche Zahl von inoffiziellen Nachbauten bzw. Nachbauten von Nachbauten. Vgl. Flamm 1988, S. 52, 76f.

³ Mindell 1996, Ch. 9; Edwards 1996; Ch. 6; Kreibich 1986, S. 279ff.; Weizenbaum 1976, S. 232f., passim; Bowker 1993.

auf nichttechnische Systeme, die Einbeziehung des Menschen als aktive Komponente eines soziotechnischen Systems sowie eine ganzheitliche und zielorientierte Entwurfsmethodik von Bedeutung.

3.1.1 Information und Rückkopplung

Der Begriff Kybernetik (griechisch für Steuermannskunst) geht auf Arbeiten des amerikanischen Mathematikers Norbert Wiener (1894–1964, Abb. 9) zurück, der sich in den dreißiger und frühen vierziger Jahre am MIT vor allem mit der Theorie der Regelung und der harmonischen Analyse beschäftigt hatte. Im Jahre 1940 wurde er von seinem ehemaligen Kollegen und Freund Vannevar Bush gebeten, im Rahmen eines vom NDRC finanzierten Projektes einen sogenannten *Antiaircraft Director* zu konstruieren, der die Treffgenauigkeit von Flugabwehrgeschützen verbessern sollte. In der Zusammenarbeit mit dem Ingenieur Julian Bigelow entstand zwar zwischen 1940 und 1942 kein solches Gerät, wohl aber erste Arbeiten zu einer Systemtheorie, bei der von der konkreten Beschaffenheit des betrachteten Systems abstrahiert und die Gesetzmäßigkeiten ihrer Zustandsänderungen und Prozeßabläufe untersucht wurden.¹ Obwohl Wiener in diesen Schriften außer einen Aufsatz James Clerk Maxwells von 1868 keine anderen Quellen nannte, konnte er auf jahrzehntelangen Erfahrungen aufbauen, die Ingenieure bei der Konstruktion von technischen Regelungseinrichtungen gesammelt hatten. So wird in einer neueren historischen Arbeit darauf hingewiesen, daß Bushs *Differential Analyzer* von seiner Verwendung her weniger eine Rechenmaschine war, als eine Vorrichtung zur Simulation von Regelkreisen.² Obwohl der Umfang von Wieners Beitrag zur Kybernetik bis heute umstritten ist, hat ihn sein 1948 erschienenes Buch *Cybernetics or, Control and Communication in the Animal and the Machine* und seine barocke Persönlichkeit zur zentralen Person bei der Entwicklung der Kybernetik werden lassen.³

Im Rahmen dieser Untersuchungen entstand nicht nur der stochastische Informationsbegriff, den der Ingenieur Claude E. Shannon zu einer umfassenden mathematischen Theorie der Information und Kommunikation erweiterte, sondern auch eine Theorie der Regelung, die nicht nur auf technische, sondern auch auf biologische und gesellschaftliche Systeme anwendbar war.⁴

¹ Masani and Phillips 1985.

² Mindell 1996, Chapter 5 – Vergleiche auch meine Ausführungen auf Seite 55.

³ Heims 1980; Heims 1991; Galison 1994; Edwards 1996, S. 180ff. – In dieser Rolle gleicht Wiener den beiden Hauptfiguren seines Romans *Die Versuchung* (1959), in dem er die Entwicklung der Regelungstechnik von einer esoterischen Idee zu einer gewinnträchtigen Industrie schildert. Dabei werden die grundlegenden Ideen eines bescheidenen Wissenschaftlers von einem eitlen Professor und einem ehrgeizigen Industriellen populär gemacht, indem sie sie als die eigenen ausgeben und vermarkten.

⁴ Rosenblueth et al. 1943; McCulloch and Pitts 1943; Shannon 1948; Wiener 1948.

3. Vom Analogrechner zum Minicomputer



Abbildung 9. Norbert Wiener um 1950

Weil an dieser Stelle nicht im Detail auf die Entwicklung der Kybernetik und ihre Wirkung auf die Wissenschaft seit 1950 eingegangen werden kann, sollen im folgenden nur einige wichtige Ideen vorgestellt werden, die der Kybernetik entstammen und für die Computerentwicklung von Bedeutung waren.

Das zentrale Prinzip der Kybernetik ist die negative Rückkopplung in geschlossenen Regelkreisen. Es ist das Verdienst der Kybernetik, dieses bereits seit Jahrhunderten verwendete Prinzip theoretisch fundiert und auf nichttechnische Bereiche ausgedehnt zu haben. Dabei kam Wiener zu der Erkenntnis, daß die negative Rückkopplung ein universales Prinzip darstellt, das nicht nur zur Steuerung von Energie- und Materieströmen in technischen Systemen¹ verwendet werden kann, sondern in dynamischen System aller Art zu beobachten ist.² Die Erkenntnis, daß die Rückkopplung nicht nur einen technischen, sondern einen methodischen Ansatz darstellt, macht die neue Qualität der kybernetischen Denkweise aus. Sie weist nämlich über die linear-kausale Denkweise hinaus, die bis dahin Wissenschaft und Technik beherrschte, und führte zu einer Betrachtungsweise, in der die Wirkungen von Ursachen selbst wieder zur Ursache von Wirkungen

¹ Als Beispiele lassen sich der Fliehkraftregler zur Regelung einer Dampfmaschine oder das Gyroskop zur Stabilisierung von Schiffen anführen. Vgl. Bennett 1979.

² Rosenblueth et al. 1943; Wiener 1948, passim. – Als Beispiele sind neben den technischen Anwendungen vor allem die Ökologie, die Neurophysiologie und Biophysik, die Soziologie, Ökonomie und die Pädagogik zu nennen. Vgl. Kreibich 1986, Tab. 5.3-1 und 5.3-2.

werden können.¹ Damit führt das Konzept der Rückkopplung unmittelbar zu einem weiteren wichtigen Gedanken der Kybernetik, nämlich den des systemischen bzw. vernetzten Denkens.

Auch die Erkenntnis, daß sich komplexe Organisation aus Elementen zusammensetzen, die ihrerseits selbst kleine Organisationen sind, war für die Väter der Kybernetik weder neu noch ungewohnt.² Neu im Rahmen der Kybernetik war allerdings die Erkenntnis, daß die Eigenschaften des Gesamtsystems nicht durch die Summe seiner Bestandteile bestimmt werden, sondern vor allem von den Beziehungen zwischen den Elementen abhängig sind sowie die Übertragung dieser Vorstellungen auf biologische Vorgänge, wie sie etwa im menschlichen Gehirn ablaufen.³ Auf Seiten der Methodik entwickelte sich mit der Zeit deshalb ein ganzheitlicher Ansatz zum Entwurf technischer Systeme, der sich vom bisherigen analytischen Ansatz vor allem darin unterscheidet, daß die Bewertung der Systemeigenschaften nicht anhand eines theoretischen Modells, sondern durch den Vergleich mit der Realität erfolgt. Der kybernetische Entwurf legt also nicht so sehr Wert auf die Lösung von Detailproblemen, sondern auf das Erreichen des Systemziels.⁴

Unter anderem erkannte Wiener im Rahmen seiner Studien, daß in Systemen, die sowohl aus einer technischen als auch einer menschlichen Komponente bestehen, der Mensch ebenso als Regler aufgefaßt werden kann wie die technischen Systemkomponenten.⁵ In Verbindung mit der zielorientierten Planungsmethodik der Kybernetik führte dies zur Ausbildung der als *Human Factors Engineering* bezeichneten Disziplin, die sich gerade die möglichst optimale Anpassung der technischen Systemkomponenten an den Benutzer zur Aufgabe gemacht hat.⁶

3.1.2 Kybernetik und Computer

Die Kybernetik wurde stark durch die Entwicklung der Rechentechnik in den dreißiger und vierziger Jahren bestimmt. In Wieners Schriften kommt deshalb

¹ Weilenmann 1977, S. 60ff.

² Wiener führt in diesem Zusammenhang nichttechnische Beispiele wie die föderale Struktur der Schweiz, den Leviathan von Thomas Hobbes oder den Aufbau eines Organismus aus Zellen an. Vgl. Wiener 1948, S. 181.

³ Wiener 1948, S. 181ff.

⁴ Vester 1991, S. 43.

⁵ Bei Wieners und Bigelows *Antiaircraft Director* *Antiaircraft-Director-Projekt* handelte es sich um das System aus Flugzeug und Pilot. Während das Verhalten des Flugzeuges leidlich bekannt war, war das Verhalten des Piloten unsicher und nur schlecht formalisierbar. Dies erschwerte die Möglichkeit der Vorhersage der zukünftigen Flugbahn durch den *Antiaircraft Director*. Vgl. Masani and Phillips 1985, S. 169ff.; Wiener 1948, S. 13ff.; Craik 1947/48.

⁶ Sheridan 1986; Chapanis et al. 1949. Im deutschen Sprachraum sprach man lange Zeit von Anthropotechnik, heute spricht man umfassender vom Fachgebiet der Mensch-Maschine-Systeme. Vgl. Johannsen 1993, S. 3ff.

3. Vom Analogrechner zum Minicomputer

auch immer wieder die Vorstellung des Gehirns als Rechenmaschine zum Ausdruck.¹ Durch diese Analogie kam er bereits 1947 zu der Überzeugung, daß der Digitalcomputer vor allem eine logische Maschine und erst in zweiter Linie eine arithmetische Maschine ist. Nachdem Wiener außerdem nachgewiesen hatte, daß kontinuierliche Vorgänge, die Gegenstand der analogen Regelungstechnik sind, und diskrete Vorgänge, wie sie beim Digitalcomputer verwendet werden, prinzipiell äquivalent sind, kam er zu dem Schluß, daß sich der Digitalcomputer auch für Regelungs- und Steueraufgaben eigne.² Damit wurden dem Digitalcomputer – zumindest theoretisch – Problembereiche erschlossen, die bislang als Domäne der analogen Rechentechnik angesehen wurden, die sich aber bereits Ende der dreißiger Jahre wegen ihrer elektromechanischen Bauart für komplexe Aufgaben als zu langsam herausgestellt hatte.³

Wieners direkter Beitrag zur Computerentwicklung blieb hingegen bescheiden. Sein Antrag zur Entwicklung eines Digitalcomputers zur Lösung von partiellen Differentialgleichungen wurde 1940 von Vannevar Bush als Vorsitzendem des NDRC mit Hinweis auf den zu erwartenden Entwicklungsaufwand abgelehnt.⁴ An einem Nachkriegsprojekt des *Center of Analysis* am MIT, das von der Rockefeller-Stiftung finanziert wurde und dessen Leitung man ihm angeboten hatte, nahm Wiener nicht teil. Das Projekt wurde daraufhin schon bald zugunsten des *Project Whirlwind* aufgegeben.⁵

Die Kybernetik fand sehr schnell Einzug in die Entwicklung von Digitalcomputern am MIT. Dabei war es gewiß vorteilhaft, daß das *Project Whirlwind* am MIT von einem Team junger Ingenieure durchgeführt wurde, die noch nicht vom Paradigma des Computers als Rechenmaschine geprägt, wohl aber mit kybernetischen Konzepten und regelungstechnischen Systemen in Kontakt gekommen waren.

3.2 Whirlwind: Der erste Echtzeitcomputer

Am MIT wurde zwischen 1944 und 1951 im Rahmen des *Project Whirlwind* der größte und teuerste elektronische Digitalrechner der ersten Generation entwickelt.

¹ Wiener 1964, S. 267ff.; Wiener 1948, S. 9ff. Vgl. zu diesem Thema auch Neumann 1955, Shannon und McCarthy 1974. Einen historischen Überblick über diese Denkrichtung vom 18. Jahrhundert bis zur Gegenwart gibt MacCormac 1984.

² Wiener 1948, S. 74ff., 139, 153ff.; Wiener 1958, S. 8f.

³ Wiener 1987; Wiener 1964, S. 235.

⁴ Wiener 1987; Masani 1987, S. 185.

⁵ Wildes and Lindgren 1985, S. 233ff.

Weil dieses Projekt sehr gut dokumentiert und durch Kent C. Redmond und Thomas M. Smith bereits umfassend historisch aufgearbeitet ist¹, sollen im folgenden nur einige Aspekte des *Project Whirlwind* näher beleuchtet werden, die den Whirlwind-Computer und seine Nachfolger als Bindeglieder zwischen Vannevar Bushs Analogcomputern und modernen Workstations und Personal Computern ausweisen.

3.2.1 Vom Analog- zum Digitalcomputer

Die Entwicklung des Whirlwind-Computers wurde von einer Forschergruppe getragen, die dem Laboratorium für Servomechanismen angehörten, das 1940 von Gordon S. Brown gegründet worden war. Brown war, wie auch viele seiner Mitarbeiter, Schüler von Vannevar Bush und Harold Hazen, so daß die Arbeiten des Laboratoriums für Servomechanismen eine inhaltliche wie methodische Fortsetzung von Bushs Forschung darstellten.²

Es ist deshalb auch nicht überraschend, daß Captain Luis de Florez, der Leiter der *Special Devices Division* (SDD), einer Abteilung des *Bureau of Aeronautics*, im Dezember 1943 an das MIT herantrat, um die Entwicklung eines verbesserten Trainingssimulators für Bomberbesatzungen in Auftrag zu geben. Im Zentrum des im Herbst 1944 begonnenen Forschungsprojekts sollte die Entwicklung eines Analogcomputers in der Tradition von Bushs *Differential Analyzer* stehen, mit dessen Hilfe die Flugeigenschaften neuer Flugzeugtypen vor deren Fertigstellung simuliert werden konnten. Außerdem sollte den zukünftigen Piloten dieser Flugzeuge schon frühzeitig die Möglichkeit gegeben werden, bestimmte Manöver anhand des sogenannten *Aircraft Stability and Control Analyzer* (ASCA) zu trainieren, der zu diesem Zweck um eine bewegliche Flugzeugkabine und entsprechende Lenk- und Steuereinrichtungen erweitert werden sollte.³

Mit der Leitung des Projekts wurde der damals 26jährige Ingenieur Jay W. Forrester betraut, der bereits seit 1939 theoretische und praktische Erfahrungen bei der Entwicklung von servomechanischen Systemen hatte sammeln können. Sein Stellvertreter und Chefingenieur wurde der 23jährige Robert R. Everett, der seit 1943 dem MIT angehörte.⁴ Schon im Laufe des Jahres 1945 begann Forrester nach alternativen technischen Möglichkeiten zu suchen, weil immer deutlicher wurde, daß ein Analogcomputer nach dem Vorbild des *Differential Analyzer* wegen seiner elektromechanischen Bauteile für die komplexe und zeitkritische Auf-

¹ Smith 1976; Redmond and Smith 1977; Redmond and Smith 1980.

² Wildes and Lindgren 1985, S. 210ff.

³ Redmond and Smith 1977, S. 51.

⁴ Redmond and Smith 1977, S. 51f.

3. Vom Analogrechner zum Minicomputer



Abbildung 10. Jay W. Forrester (links) und Robert R. Everett (rechts)

gabe einer Flugzeugsimulation zu langsam war.¹ Nach einem Besuch des ENIAC Projekts in Philadelphia wendete er sich im Sommer und Herbst 1945 immer mehr der vielversprechenden neuen elektronischen Digitaltechnik zu, von deren Vorteilen ihn sein ehemaliger Studienkollege Perry O. Crawford, Jr. (* 1917), ebenfalls ein Schüler von Sam Caldwell und Vannevar Bush, letztlich überzeugte.² Dabei hatte die Digitaltechnik für Forrester wie für andere Ingenieure eine ganz besondere Attraktivität: Einerseits waren die Grundelemente der Digitaltechnik extrem einfach, ein Flipflop zur Speicherung einer Binärziffer bestand beispielsweise nur aus zwei Röhren und wenigen Widerständen und Kondensatoren. Andererseits ließen sich aus ihnen Systeme beliebiger Funktionalität und Komplexität aufbauen.³

¹ Redmond and Smith 1980, S. 16, 25f. – Ohne Berücksichtigung der Triebwerke waren zum damaligen Zeitpunkt 47 Gleichungen mit 53 Variablen simultan und in Echtzeit zu lösen.

² Redmond and Smith 1980, S. 26f. – Crawford hatte seine Abschußarbeit 1942 in dem von Bush initiierten und von Caldwell geleiteten Projekt zur Entwicklung des *Rockefeller Differential Analyzer* durchgeführt, der kein reiner Analogcomputer mehr war, sondern als hybride Rechnerarchitektur bezeichnet werden muß, in die auch Erfahrungen aus Bushs *Rapid Arithmetical Machine* einfließen. Vgl. auch Seite 68, Fußnote 4.

³ Redmond and Smith 1980, S. 36.

Im Frühjahr 1946 wurde das neue Konzept für den Bau eines Flugzeugsimulators auf der Basis eines Digitalcomputers vom *Office of Naval Research* (ONR)¹ akzeptiert und erhielt den Namen »Project Whirlwind«.²

3.2.2 Der Whirlwind-I-Computer

Die Hardware des Whirlwind-I-Computers³ hat erst 1959 seine endgültige Gestalt angenommen, denn sie hat sich zwischen dem Beginn des Projektes im Herbst 1945 und der offiziellen Stilllegung des Computers am 30. Juni 1959 ständig verändert. Der Befehlssatz wurde nach Bedarf erweitert, technologisch veraltete Komponenten gegen modernere ausgetauscht und Peripheriegeräte für wechselnde Anwendungen entwickelt.

Die grundlegende Systemarchitektur wurde von Robert Everetts *Block Diagrams Group* bis zum Frühjahr 1947 im wesentlichen fertiggestellt. Es folgte die Entwicklung und der Test prototypischer Schaltkreise wie Flipflops, Addierer und Multiplizierer sowie der Speicherröhren. Erst nachdem – mit Ausnahme der Speicher – alle Systemelemente zuverlässig funktionierten, wurde ab 1948 der eigentliche Computer aufgebaut. Obwohl man keinen genauen Zeitpunkt festlegen kann, war Whirlwind im Laufe des Jahres 1949 betriebsbereit. 1950 wurde der bis dahin verwendete »Testspeicher«⁴ durch Speicherröhren ersetzt. Am 14. März 1951 wurde die komplette Zentraleinheit offiziell in Betrieb genommen.⁵

Schon im Frühjahr 1949 wurde die Entwicklung eines neuen Typs von Hauptspeicher in Angriff genommen, der sich als schneller und zuverlässiger herausstellte als die problematischen Speicherröhren. Das Ergebnis der Entwicklung waren Magnetskernspeicher, die nach umfangreichen Tests im Sommer 1953 die Speicherröhren im Whirlwind-I-Computer ersetzten.⁶ Im Lauf der Zeit wurde der Hauptspeicher von 32 Registern (1949) über 1 280 Register (1951) schließlich auf 2 048 Register (1953) erweitert.⁷

¹ Die *Special Devices Division* war seit Ende des Krieges unter dem Namen *Special Devices Center* (SDC) zu einer Abteilung des ONR geworden. Sein Leiter wurde Perry Crawford. Vgl. Rees 1982, S. 112.

² Redmond and Smith 1980, S. 41f.

³ Whirlwind I war als Prototyp für einen späteren, industriell zu produzierenden Computer gedacht. Der Whirlwind II Computer, der ab 1955 von IBM ausschließlich für die Verwendung im SAGE-System (vgl. Abs. 3.3) produziert wurde, ist bekannter unter seiner militärischen Bezeichnung AN/FSQ-7. Vgl. Astrahan and Jacobs 1983.

⁴ Dabei handelte es sich um fünf Flipflop-Register und 27 manuell zu bedienende Register (*toggle switch registers*). Vgl. Everett and Swain 1947, S. 19f.

⁵ Everett 1980, S. 367, 372; Redmond and Smith 1980, Kapitel 4 und 12.

⁶ Evans 1983a; Forrester 1951; Papian 1953; Forrester 1956.

⁷ Everett and Swain 1947, Bd. 1, S. 19; Everett 1951, S. 71; Serrell et al. 1962, S. 1048.

3. Vom Analogrechner zum Minicomputer

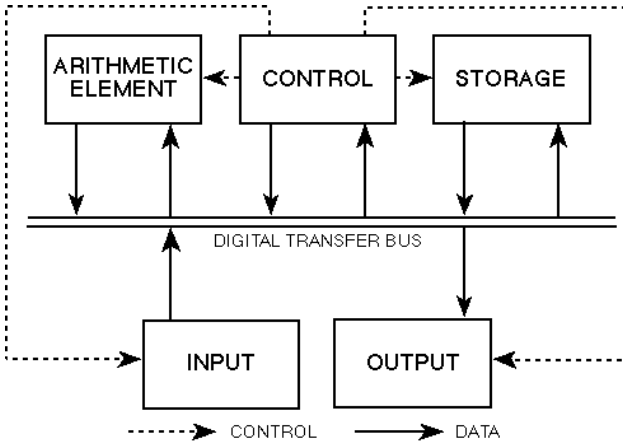


Abbildung 11. Einfaches Blockdiagramm des Whirlwind-I-Computers

Als Peripheriegeräte wurden fotoelektrische und elektromechanische Lochstreifenleser, elektromechanische Schreibmaschinen und Oszilloskopanzeigen verwendet. Außerdem bestand die Möglichkeit zum Anschluß von diversen analogen Ein- und Ausgabegeräten. Später wurden auch Magnetbandeinheiten und Trommelspeicher verwendet, sowie eine Reihe von Ein- und Ausgabegeräten, die speziell für das Luftraumüberwachungssystem SAGE (*Semi-Automatic Ground Environment*) entwickelt wurden (vgl. Abs. 3.3).¹

Beim Systementwurf richteten sich Everett und Forrester zunächst nach der Von-Neumann-Architektur, die im sogenannten EDVAC-Bericht skizziert worden war.² Sie mußten aber rasch feststellen, daß die serielle Architektur des EDVAC zwar mit vergleichsweise geringem Hardwareaufwand zu realisieren, aber für Echtzeitanwendungen wie den *Aircraft Analyzer* zu langsam war.³

Whirlwind I entstand deshalb als paralleler, synchroner Computer mit Festpunktarithmetik, bei dem negative Zahlen im 2er-Komplement dargestellt wurden. Er setzte sich – wie die meisten Computer bis zum heutigen Tage – aus dem Rechenwerk (*Arithmetic Element*), dem Steuerwerk (*Control*), dem Speicher (*Storage*) sowie Ein- und Ausgabeeinheiten (*Input*, *Output*) zusammen, die über einen zentralen Bus (*Digital Transfer Bus*) miteinander verbunden waren (Abb. 11). Whirlwind war ein speicherprogrammierbarer Computer, der mit einer Wortlänge

¹ Everett 1951, S. 74.

² Neumann 1945 – Der Electronic Discrete Variable Computer wurde 1945 als Nachfolger des ENIAC an der Moore School geplant, aber erst 1951 fertiggestellt. Vgl. Godfrey and Hendry 1993.

³ Everett and Swain 1947, S. 1; Redmond and Smith 1980, S. 52ff.



Abbildung 12. Kontrollraum um 1950. Von links nach rechts: Stephen H. Dodd, Jay Forrester, Robert Everett, Ramona Ferenz

von 16 bit arbeitete. Er verfügte über 27 Ein-Adreß-Befehle, die sich aus einer 11-bit-Adresse und einem 5-bit-Funktionscode zusammensetzten. Physisch handelte es sich um eine sehr große Maschine mit mehr als 5 000 Vakuumröhren und etwa 11 000 Kristalldioden.¹

Das Steuerwerk war in Form von zwei Diodenmatrizen realisiert. Die erste Matrix dekodierte den Funktionscode einer Anweisung, während die zweite Matrix für die Operationensteuerung des Rechenwerks verwendet wurde. Das Steuerwerk wurde mit einer Taktrate von 1 MHz betrieben und konnte bis zu 20 000 Instruktionen pro Sekunde verarbeiten.²

Das Rechenwerk des Whirlwind I wurde mit einer Taktrate von 2 MHz betrieben, bestand aus drei Registern und einem Zähler und umfaßte sieben Grundoperationen: Addition, Subtraktion, Multiplikation, Division, Schieben, Datentransfer von und zum Bus sowie die Ermittlung des Vorzeichens der im Akkumulatorregister gespeicherten Zahl. Dabei wurde die Multiplikation auf eine sukzessive Addition und die Division auf eine sukzessive Subtraktion zurückgeführt. Eine komplette Addition benötigte 3 μ s, eine Multiplikation durchschnittlich 16 μ s.³

¹ Everett and Swain 1947, S. 1, 7; Serrell et al. 1962, S. 1047.

² Everett and Swain 1947, S. 1; Everett 1951, S. 70f.

³ Everett and Swain 1947, S. 18, 21ff. – Mit dem Befehl »Vorzeichenermittlung« bezeichneten Everett und Swain die Klasse der Sprung- bzw. Verzweigungsbefehle.

3. Vom Analogrechner zum Minicomputer



Abbildung 13. Whirlwind-I-Computer im Aufbau (um 1949)

Damit war der Whirlwind-I-Computer zu Beginn der fünfziger Jahre leistungsfähiger als alle anderen bis dahin fertiggestellten Computer.¹

Trotz dieser unbestrittenen Stärken des Whirlwind-I-Computers wurde er von den Zeitgenossen mit Skepsis betrachtet. Dies betraf vor allem die Wortlänge von nur 16 bit: Sie sei für wissenschaftliche Berechnungen in Hinblick auf eine ausreichende Genauigkeit der Ergebnisse nicht ausreichend, arbeiteten doch andere Computer der Zeit mit Wortlängen zwischen 35 und 45 bit:

»The Whirlwind I computer was a 16-binary-digit machine which many people looked upon as too short a register length to be of any practical use because the scientific problems and the problems of interest to mathematicians were thought to require a 30- or 40-binary-digit length. We chose 16 mainly to hold down the size and the complexity of the machine.«²

Die Besonderheiten des Systementwurfs, die Leistungsfähigkeit und Zuverlässigkeit der Systemkomponenten hatten ihren Grund in der Art der Anwendungen, die mit Hilfe des Whirlwind-I-Computers realisiert werden sollten.

¹ Flamm 1988, S. 9, Tab. 2-1.

² Evans 1983b, S. 400.

3.2.3 Ein Computer auf der Suche nach Anwendungen

Die meisten der Computerprojekte der vierziger Jahre zielten auf mathematische, kaufmännische oder statistische Anwendungen ab. In diesen Bereichen gab es bereits eine lange Tradition in der Verwendung von Rechen- und Lochkartenmaschinen sowie in der Verarbeitung von diskreten Werten. Der Weg war hier also für die Einführung des Digitalcomputers gebahnt.

Anders sah die Situation im Bereich der heute als Prozeßrechner bezeichneten Maschinen aus, die Regelungs- und Steueraufgaben durchführen. Die moderne Regelungstechnik war während der vierziger Jahre wie angedeutet selbst noch eine in der Entstehung begriffene Disziplin, die sich vor allem mit der Entwicklung von Analogtechnik befaßte, weil die zu steuernden Prozesse normalerweise analogen Charakter haben. Die Etablierung des Prozeßrechners fand daher unter ungleich ungünstigeren Randbedingungen statt als die wissenschaftlicher, statistischer oder kaufmännischer Datenverarbeitungssysteme. Das *Project Whirlwind* legt für diese problematische Etablierung ein beredtes Zeugnis ab und ist gleichzeitig ein Beispiel für einen anwendungsorientierten Technikentwurf.

Unter den frühen Computerentwicklungen nahm das *Project Whirlwind* eine Sonderrolle ein, da es nach Aussage von Jay Forrester in erster Linie nicht die Weiterentwicklung der Rechentechnik, sondern die Realisierung bestimmter Anwendungen mit Hilfe des Computers zum Ziel hatte:

»The Project Whirlwind group considers its principal long-range objective to be research in digital computer applications, although such research does normally imply a substantial effort in the development and construction of required equipment.«¹

Dieses Credo Forresters hatte die völlige Unterordnung aller technischen Anforderungen und der sich daraus ergebenden Problemstellungen unter die beabsichtigte Aufgabe des Computers zur Folge. Die entwicklungsleitende Anwendung war zunächst der *Aircraft Analyzer*, also eine Flugzeugkanzel, die mit Hilfe des Computers gesteuert werden sollte. Der Flugzeugsimulator hatte allerdings seit dem Übergang zur Digitaltechnik eine immer geringere Priorität erhalten, weil die Konstruktion des Computers immer mehr Ressourcen band. Das sich gleichzeitig entwickelnde Konzept des Universalcomputers ließ außerdem die Konzentration auf eine einzige Aufgabenstellung unnötig einschränkend erscheinen.² Mit der Aufgabe der ursprünglichen Simulatoridee im Juni 1948 wurde der entscheidende Schritt vom *Spezial-* zum *Universalrechner* getan, der sich bereits seit Sommer 1947 abgezeichnet hatte.³

¹ Forrester 1980, S. 225.

² Redmond and Smith 1980, S. 57ff.

³ Smith 1976, S. 456; Redmond and Smith 1980, S. 59.

3. Vom Analogrechner zum Minicomputer

Durch diese Entscheidung gerieten Jay Forrester und Robert Everett angesichts der hohen Kosten¹ und der für das *Office of Naval Research* nicht mehr erkennbaren Einsatzbereiche ihres Computers in Rechtfertigungszwang. Sie entwickelten daher eine Reihe von Vorschlägen, wie man den schnellen und teuren Computer einsetzen könnte, um auch weiterhin durch das ONR finanziert zu werden. Obwohl sie dabei besonders die Eignung des Computers für Regel- und Steuerungsaufgaben betonten, wiesen sie in ihren Vorschlägen auch auf die universelle Verwendbarkeit von Whirlwind für andere Aufgabenbereiche hin.

Damit geriet das *Project Whirlwind* allerdings auch in einen grundsätzlichen Konflikt mit dem grundlagenorientierten ONR, das in den späten vierziger Jahren die wichtigsten amerikanischen Computerprojekte finanzierte und dabei seine Hauptaufgabe in der Förderung der reinen und angewandten Mathematik in Hinblick auf die wissenschaftliche Nutzung des Digitalcomputers sah. Die Verantwortlichen des ONR stellten zeitweise die Erfolgsaussichten des *Project Whirlwind* grundsätzlich in Frage, indem sie auf die mangelnde mathematische Qualifikation des hauptsächlich aus Ingenieuren bestehenden Entwicklungsteams hinwiesen. Offensichtlich spielte hier von Neumanns IAS-Projekt eine wichtige Rolle für die Vorstellung des ONR, wie ein erfolversprechendes Computerprojekt aussehen sollte.² Schließlich gingen sie ganz im Sinne des *Assembly-Line-Modells* davon aus, daß die Weiterentwicklung des Computers besonders von den Fortschritten der Mathematik als Grundlagenwissenschaft abhängig sei.³

Forrester und Everett griffen bei ihren Vorschlägen Überlegungen auf, die Perry Crawford bereits in seinem Beitrag zu den *Moore School Lectures*⁴ im Sommer 1946 formuliert hatte. Crawford, mittlerweile Direktor des *Special Devices Centers*, sah in der entstehenden Digitalrechner-technik die Möglichkeit, den Benutzer eines technischen Systems von der drohenden Informationsüberflutung zu befreien:

¹ Das *Project Whirlwind* hatte zu dieser Zeit monatliche Kosten von über 140 000 \$. Insgesamt war das Projekt mit 4 bis 5 Mio. \$ mehr als viermal teurer als das zweitteuerste Computerprojekt (Harvard Mark III) und sogar mehr als sechsmal teurer als John von Neumanns IAS Computer. Vgl. Redmond and Smith 1980, S. 108, 166; Flamm 1988, S. 76.

² Redmond and Smith 1980, S. 69.

³ Rees 1950; Rees 1982, S. 103, 105 – Der Grund für den Konflikt zwischen Forrester und dem ONR lag nicht allein an der Grundlagenorientierung des ONR. Für das Finanzjahr 1948 war der Etat des SDG von 11 auf 5 Mio. \$ gekürzt worden. Der von Forrester verlangte Betrag von 1.5 Mio. \$ hätte fast 10% des kompletten ONR-Etats verschlungen. Vgl. Edwards 1996, S. 78f.

⁴ 1946 wurde an der Moore School of Electrical Engineering in Philadelphia eine Sommerschule über den Entwurf von elektronischen Digitalcomputern durchgeführt, an der ein Großteil der englischen und amerikanischen Computerentwickler teilnahm. Die dabei vermittelten Konzepte waren impulsgebend für die Entwicklungen der nachfolgenden Jahre. Vgl. Campbell-Kelly and Williams 1985a.

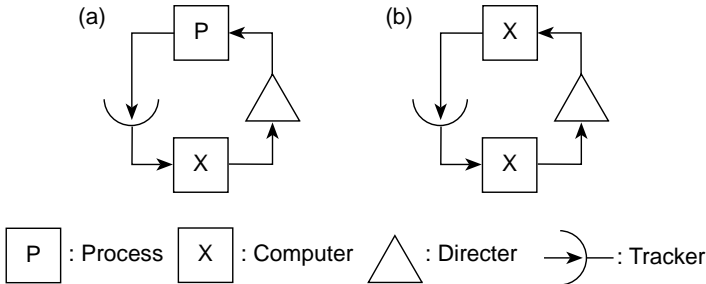


Abbildung 14. Schema einer Steuerung (a) und eines Simulators (b) unter Verwendung eines Digitalcomputers nach Perry Crawford. Tracker bezeichnet die Dateneingabe, Director die Datenausgabe über einen Servomechanismus.

»... automatic computers constitute means for enlarging the capacity of human beings to handle, and, in particular, to interpret information. The actions of human beings that involve interpretation ... have as their objective control of some ›process‹ occurring in the physical world.«¹

Crawford, der mit dieser Auffassung Vannevar Bush nahestand, schlug in der Folge drei konkrete Anwendungsbereiche für den Digitalcomputer vor: wissenschaftliche Berechnungen, Steuerungen und Simulationen. Er griff dabei in Zielsetzung und Methodik explizit auf die Erkenntnisse aus der Servomechanik zurück, insbesondere die Systembetrachtung und den rückgekoppelten Regelkreis. Bei der Verwendung für Steuerungsaufgaben sollte der Computer die Aufgaben der bisherigen Servomechanik übernehmen. Indem er vorschlug, der Computer könne auch das Verhalten der zu regelnden Größe nachbilden, formulierte Crawford das Prinzip der (digitalen) Computersimulation (Abb. 14). Durch die Betonung des Informationsflusses innerhalb eines solches Systems bekam die Entwicklung von geeigneten Ein- und Ausgabegeräten eine ähnlich große Bedeutung wie die Rechenleistung und die Zuverlässigkeit.²

Forrester und Everett orientierten sich bei ihren Vorschläge zunächst eng an der Kernaufgabe des *Special Devices Center*, nämlich der Entwicklung von Geräten für den Marineeinsatz. Sie empfahlen im September 1947 in zwei provokanten Memoranda an das ONR den Einsatz des Whirlwind-Computers für strategische Aufgaben bei der U-Boot-Bekämpfung.

Im ersten Memorandum schlugen sie einen Simulator vor, mit dem die U-Boot-Bekämpfung trainiert werden konnte. Dazu sollten die Kommandostände von zehn Schiffen und fünf U-Booten mit realistischen Navigationseinrichtungen

¹ Crawford 1985, S. 375.

² Crawford 1985, S. 376.

3. Vom Analogrechner zum Minicomputer

(Steuer, Kompaß, Radar, Sonar etc.) versehen und mit dem zentralen Computer verbunden werden. Die Daten über Kurs und Geschwindigkeit jedes einzelnen (simulierten) Schiffes und U-Bootes sollten etwa alle 140 ms gemessen und daraus die aktuellen Positionen aller Schiffe und U-Boote errechnet und an die Ausgabegeräte der Kommandostände übertragen werden.¹

Im zweiten Memorandum wurde der Einsatz eines solchen Überwachungssystems an Bord des Kommandoschiffes eines mit der U-Boot-Bekämpfung beauftragten Flottenverbandes angeregt. Dort sollten die Daten der dem Flottenverband angehörenden Schiffe und der entdeckten feindlichen U-Boote gesammelt und mit Hilfe des Computers verarbeitet werden, um den kommandierenden Offizieren jederzeit ein Gesamtbild der Lage geben zu können.² Damit beschrieben Forrester und Everett prototypische militärische Command and Control Systems, die bis in die siebziger Jahre im Zentrum der militärischen Computernutzung standen.³ Vor allem kamen sie zu dem – nicht sonderlich überraschenden – Ergebnis, daß sich der von ihnen konstruierte Whirlwind-Computer für den in dieser Zeit beginnenden kalten Krieg besonders gut eignete.⁴

Forrester und Everett sahen auch, daß sich die Verwendung des Digitalcomputers für Steuer- und Regelungsaufgaben nicht auf den militärischen Bereich beschränken mußte. In einem weiteren Memorandum vom Juli 1948 hielten sie auch die Steuerung industrieller Prozesse (vor allem im Bereich der ölverarbeitenden und chemischen Industrie) und die Überwachung des Luftraums mit Hilfe des Computers innerhalb weniger Jahre für realisierbar.⁵

Forrester und Everett waren sich aber auch im klaren, daß ein so leistungsfähiger Computer wie Whirlwind I zwar besonders gut für Echtzeitanwendungen geeignet war, daß er als Universalcomputer aber auch zur Bewältigung konventioneller Aufgaben eingesetzt werden konnte. Schon aus Gründen einer wirtschaftlichen Nutzung schlugen sie deshalb auch eine Verwendung für mathematische und ingenieurwissenschaftliche Berechnungen vor.⁶ Da der Whirlwind-Computer im

¹ Forrester et al. 1949, S. 1ff.

² Forrester and Everett 1949, S. 1ff.

³ Eurich 1991, S. 111f., 167f. – Das MIT wurde seit etwa 1950 mit seinem Lincoln Laboratory und der 1958 ausgegründeten MITRE Corporation zu einem Kernbestandteil des militärisch-industriellen Komplexes. Vgl. U. S. Congress, Office of Technology Assessment 1995; Leslie 1993.

⁴ Forrester et al. 1949, 28f.; Forrester and Everett 1949, S. 16.

⁵ Forrester 1980, S. 225f.; Forrester 1958, S. 382f.

⁶ In der Tat wurde Whirlwind I in den ersten acht Monaten seines offiziellen Betriebs für Berechnungen im Bereich der industriellen Produktionsplanung, für magnetische Flußberechnungen und zur Bestimmung optischer Konstanten und Autokorrelationskoeffizienten verwendet. Nicht zuletzt wurde mit Hilfe von Whirlwind die Programmierung der ersten am MIT entwickelten numerisch gesteuerten Werkzeugmaschine vorgenommen. Vgl. Everett 1951, S. 70f.

damals üblichen Einprozeß- und Einbenutzerbetrieb für mathematische Probleme überdimensioniert war, regte Forrester den Aufbau eines »Problemlösetztes« (*problem solving network*) an, bei dem eine Vielzahl von Nutzern über Telefon- oder Telegrafentelefonleitungen mit dem zentralen Computer verbunden werden sollten – eine frühe Formulierung der seit Beginn der sechziger Jahre entwickelten Teilnehmer- oder *Time-Sharing*-Systeme (vgl. Kapitel 4).¹

Auf Anfrage von MIT-Präsident Karl Compton, der wegen des schwelenden Konflikts mit dem ONR beunruhigt war, faßten Forrester, Everett und ihre Mitarbeiter ihre Überlegungen zur Verwendung des Whirlwind-Computers im Sommer 1948 in einem Bericht zusammen, in dem sie auch die finanziellen Ressourcen abschätzten, die in den nächsten 15 Jahren notwendig wären. Die vorgeschlagenen (militärischen) Anwendungsgebiete reichten vom Luftraumüberwachungssystem, über Feuerleitsysteme und Simulatoren bis zu logistischen Planungssystemen. Für die Realisierung dieses Szenarios kamen sie auf einen Finanzbedarf von über 2 Mrd. \$², einer 1948 übertrieben hoch erscheinenden Summe, die sich jedoch im Rückblick als korrekt herausgestellt hat.³

Keines dieser Anwendungsszenarien wurde tatsächlich am MIT realisiert. Dennoch hatten Forresters und Everetts Memoranda eine beachtliche Wirkung. Als das neugegründete *Air Defense System Engineering Committee*, nach seinem Vorsitzenden George Valley auch Valley-Kommission genannt⁴, 1949 feststellte, daß ein flächendeckendes Luftverteidigungssystem nur mit Hilfe eines leistungsfähigen Digitalcomputers zu realisieren war, machte Jerome B. Wiesner vom MIT-Forschungslaboratorium für Elektronik die hauptsächlich mit MIT-Professoren besetzte Kommission auf den Whirlwind-Computer aufmerksam, der sich offenbar für diesen Zweck eignete.⁵ Die Einbeziehung des Whirlwind-Computers in das ab 1950 aufgebaute strategische Luftüberwachungssystem SAGE garantierte nicht nur die Fortsetzung des Projekts⁶, sondern stellte auch die Flexibilität von Forresters und Everetts Konzept bei zeitkritischen Aufgaben unter Beweis (vgl. Abs. 3.3).⁷ Diesen Prozeß hat der Historiker Paul Edwards

¹ Forrester 1980, 232ff.

² Forrester et al. 1948, Figure 6.

³ Forrester, in Tropp et al. 1983, S. 380f.

⁴ Die Gründung des *Air Defense System Engineering Committee* war eine Reaktion auf die Explosion der ersten sowjetischen Atombombe im August 1949. Vor dem Hintergrund des kalten Krieges, der Berliner Blockade und des Koreakrieges wurde erstmals ein sowjetischer Luftangriff auf die USA für möglich gehalten. Vgl. Jacobs 1983, S. 323 und Hughes 1998, S. 18ff.

⁵ Valley 1985, S. 206ff.

⁶ Das ONR hatte die finanzielle Unterstützung des *Project Whirlwind* ab dem Finanzjahr 1950 stark eingeschränkt. Ohne die finanzielle Unterstützung durch die U.S. Air Force hätte das *Project Whirlwind* seine Forschungsarbeiten stark einschränken oder gar einstellen müssen. Vgl. Redmond and Smith 1980, S. 155ff.

⁷ Jacobs 1986, S. 22ff.

3. Vom Analogrechner zum Minicomputer

als zentralen Mechanismus bei der Ausbildung von Leitbildern in der militärisch orientierten Computerentwicklung identifiziert und als »wechselseitige Ausrichtung« (*mutual orientation*) bezeichnet.¹

Das Hauptverdienst des *Project Whirlwind* liegt somit in der Verbindung, die zwischen der Mathematikertechnik des Digitalcomputers und den Ingenieurwendungen geschaffen wurde. Es steht damit am Beginn einer Diffusion der Computertechnik, die heute alle Bereiche der Technik (und des Alltages) erreicht hat. Es bedurfte jedoch noch einiger Jahre, bis dies anerkannt wurde. Schon Ende der fünfziger Jahre stellte Maurice Wilkes – selbst einer der Computerpioniere der ersten Stunde – fest, daß die wahre Bedeutung eines technischen Fortschritts nicht in der bloßen Beschleunigung von bereits bekannten Anwendungen liege, sondern darin, bislang unmögliche Anwendungen zu ermöglichen. Dennoch äußerte er Zweifel, ob sich Computersteuerungen außer in Spezialbereichen überhaupt durchsetzen würden.²

Das Muster, nach dem die Ingenieure des *Project Whirlwind* die Digitalcomputertechnik in ihre bisherige Forschungstätigkeit integrierten, unterschied sich hingegen nicht auffällig von dem der Mathematiker. So wie die Mathematiker den Computer als mathematisches Instrument betrachteten, sollte er in der Vorstellung der Ingenieure für die Berechnung technischer Systeme und die Entwicklung von typisch ingenieurwissenschaftlichen Anwendungen in der Regel- und Steuerungstechnik eingesetzt werden.

3.2.4 Wie Ingenieure Computer entwerfen

Digitalcomputer besaßen eine Komplexität wie kein anderes elektronisches Gerät zuvor und stellten die Entwickler deshalb vor Probleme einer bislang unbekanntenen Größenordnung.³ Sie erforderten eine inhaltliche, organisatorische und methodische Weiterentwicklung von Entwicklungsstrategien unter Einbeziehung von kybernetischen Methoden zur Analyse und Synthese von technischen Systemen. Beim *Project Whirlwind* läßt sich eine solche systemorientierte Entwurfsmethodik erkennen, die man seit den fünfziger Jahren als Systemtechnik (*Systems Engineering*) bezeichnet.

Das *Project Whirlwind* war – anders als viele andere Computerprojekte der vierziger Jahre – nie ein wissenschaftliches Vorhaben. Da die Ingenieure am MIT

¹ Edwards 1996, S. 81ff.

² Wilkes 1958, S. 100ff. – In den vergangenen Jahren ist in einer Reihe von Innovationsstudien der langfristige technische und wirtschaftliche Einfluß des *Project Whirlwind* detailliert nachgewiesen worden. Vgl. dazu Bruce S. Old Associates, Inc. 1981 und Hughes 1998, Ch. II.

³ Goldstine 1980, S. 153f.

den Digitalrechner vor allem als Mittel zum Zweck betrachteten, wurden automatisch andere Anforderungen an sie gestellt. Eine den Echtzeitanforderungen angemessene Leistung und ein zuverlässiger Betrieb über mehrere Stunden waren unabdingbare Voraussetzungen für den Betrieb eines Flugzeugsimulators oder eines Luftraumüberwachungssystems. Die Frage nach geeigneten technischen Systemkomponenten, vor allem zuverlässigen Röhrenschaltkreisen und Speicherelementen, hatte damit eine ungleich größere Bedeutung als bei reinen Forschungsrechnern, die zur gleichen Zeit in der Entwicklung waren. Sie wurden mit Blick auf theoretische Probleme entworfen, wie sie in der Kryptografie, der numerischen Mathematik oder der Physik auftreten und räumten der Entwicklung von praxistauglichen Baugruppen und anwendungsorientierten Systemarchitekturen nur einen untergeordneten Stellenwert ein. Damit setzte sich die schon erwähnte Betonung der Grundlagenforschung gegenüber der ingenieurwissenschaftlichen Entwicklung fort, die sich auch in der Ausrichtung der militärischen Forschungsförderung wiederfindet.

So war innerhalb des ONR die mathematische Abteilung unter Mina Rees für die Entwicklung von Computern zuständig. Das anwendungsorientierte *Special Devices Center* unter der Leitung von Perry Crawford wurde 1946 in diese Abteilung eingegliedert.¹ Bereits 1941 hatte Thornton C. Fry, der als Mathematiker bei den Bell Laboratorien und für das NDRC arbeitete, auf die unterschiedliche Methodik von Mathematikern und Ingenieuren hingewiesen:

»The typical mathematician feels great confidence in a conclusion reached by careful reasoning. He is not convinced to the same degree by experimental evidence. For the typical engineer these statements may be reversed. (...) Because of this confidence in thought processes the mathematician turns naturally to paper and pencil in many situations in which the engineer would resort to the lab. (...) For the mathematician, an argument is either perfect in every detail, in form as well as in substance, or else it is wrong. There are no intermediate classes. (...) The typical engineer calls it hair splitting.«²

Obwohl pauschale Charakterisierungen nie ganz auf konkrete Personen zutreffen, charakterisiert das beschriebene Spannungsverhältnis treffend die Situation, in der nach dem Zweiten Weltkrieg Digitalcomputer gebaut wurden. Obwohl sich John von Neumann mit der Computerentwicklung im Bereich der angewandten Wissenschaft bewegte und sich gar als Innovator betätigte, ließ er nie einen Zweifel daran, daß für ihn der reine Erkenntnisgewinn eine höhere Priorität hatte als

¹ Rees 1982, S. 112.

² Fry 1941, zitiert nach Stern 1980, S. 356.

3. Vom Analogrechner zum Minicomputer

die Entwicklung von kommerziell vermarktbareren oder auf die Belange des Militärs zugeschnittenen Computern. Diese Einstellung war durch von Neumanns Stellung als Vordenker der Computerwissenschaft und die Förderung durch das mathematisch-grundlagenorientierte ONR weit verbreitet und behinderte Computerentwürfe, die andere Ziele verfolgte.¹

Forrester, Everett und ihre Mitarbeiter waren keine Mathematiker oder Physiker, sondern Ingenieure, die während des Krieges in militärischen Entwicklungsprojekten geschult worden waren. Sie hatten dabei gelernt, abstrakte Konzepte zu erfassen und diese möglichst schnell in zuverlässig funktionierende, technische Apparate umzusetzen und bis zur Produktionsreife zu entwickeln.²

Forrester und Everett verfolgten mit Interesse auch die Arbeiten von Neumanns, schließlich mußten sie sich vom ONR immer wieder an seinen Leistungen messen lassen. Trotzdem kam in den Forschungsberichten des *Project Whirlwind* immer wieder zum Ausdruck, daß die Arbeit von Neumanns zwar wichtig, aber für die konkrete Konstruktion eines Digitalcomputers mit bestimmten Leistungsmerkmalen keine notwendige Voraussetzung sei.³ Das Personal des *Project Whirlwind* rekrutierte sich zunächst aus Mitarbeitern und Studenten des Laboratoriums für Servomechanismen. Nach dem Übergang von der Analog- zur Digitaltechnik wurden zusätzlich Ingenieure mit Radarerfahrung eingestellt.⁴ Die Radartechnik, zu deren Entwicklung das MIT ab 1941 wesentlich beigetragen hatte, umfaßte unter anderem die bis dahin komplexesten elektronischen Systeme, in denen, wie beim Digitalcomputer, impulsförmige Signale verarbeitet wurden.⁵

Aufgrund ihrer Ausbildung, ihrer bisherigen Tätigkeit und einer Nähe zu kybernetischen Ideen verfolgten Forrester und sein Team beim Entwurf und bei der Entwicklung des Whirlwind-I-Computer einen typischen Systemansatz. Es wurden Arbeitsgruppen eingerichtet, die sich mit der grundlegenden Systemarchitektur, mit Rechenschaltungen, Mathematik, Mechanik, Speichertechnik und anderen elektronischen Problemen befaßten. Die Federführung übernahm dabei nicht die mathematische Forschungsgruppe, wie dies bei anderen Projekten der Fall

¹ Redmond and Smith 1980, S. 138ff.; Stern 1980, S. 356ff.; Rees 1950.

² Forrester, in Tropp et al. 1983, S. 376, 387

³ Everett und Swain zitieren in ihrem Bericht über die Whirlwind-Architektur nur ein einziges Mal einen Text von Burks, Goldstine und von Neumann (1961), in dem unter anderem der Erwartungswert für die Anzahl der Überträge bei der Addition von zwei 40-bit-Zahlen berechnet wird. Aus dem Zusammenhang wird deutlich, daß solche Fragen beim Entwurf des Addierers für den Whirlwind-I-Computer offenbar keine Rolle gespielt haben. Vgl. Everett and Swain 1947, S. 21; auch Redmond and Smith 1980, S. 138ff.

⁴ Redmond and Smith 1980, S. 36.

⁵ Wildes and Lindgren 1985, S. 190ff.

war, sondern die von Everett geleitete *Block Diagrams Group*, die die Grundstruktur des Whirlwind erarbeitete.¹

Die *Block Diagrams Group* führte die seit den dreißiger Jahren in der Radio-, Fernseh- und Radartechnik verwendete Entwurfstechnik des Blockdiagramms auch in der Computertechnik ein.² Blockdiagramme sind zunächst Hilfsmittel, um die Komplexität von Systemen zu verringern. Dazu wird der Detaillierungsgrad des Konstruktionsplans eines technischen Systems variiert. Für das grundlegende Verständnis reicht es vielfach, das System in Blöcke zu zerlegen, die unterschiedliche Aufgaben erfüllen und über genau definierte Schnittstellen miteinander verbunden sind. So zeigt Abbildung 11 auf Seite 82 lediglich die Hauptelemente des Whirlwind-I-Computers. Diese sind über eine gemeinsame Schnittstelle, den zentralen Bus, miteinander verbunden. Jeder dieser Blöcke kann hierarchisch in weitere Funktionsblöcke zerlegt werden (Abb. 15). Erst auf der untersten Ebene wird die konkrete Realisierung dargestellt. Weil sich der Digitalcomputer aber nur aus einer relativ geringen Zahl von Grundschaltungen zusammensetzt, handelt es sich um ein elegantes Verfahren, um nicht nur die Größe der Schaltpläne in handhabbaren Dimensionen zu halten, sondern auch das Verständnis für den modularen Aufbau des Computers zu fördern.³ Vor allem dieser pädagogische Aspekt von Blockdiagrammen hat dazu beigetragen, daß die Whirlwind Blockdiagramme – obwohl sie bis Ende der fünfziger Jahre als *confidential* eingestuft waren – während der fünfziger Jahre eine enorme Verbreitung unter Computeringenieuren gefunden haben.⁴

Daneben ermöglichte die Verwendung von Blockdiagrammen eine effiziente Form des Projektmanagements. Ein komplexes Gesamtsystem kann in seiner Gesamtheit entworfen werden, ohne daß die konkrete Realisierung der einzelnen Komponenten vollständig bekannt ist. Es genügt vielmehr, die Funktion einer Komponente und seine Schnittstellen zum restlichen System festzulegen. Im folgenden können dann parallel für alle Systemkomponenten technische Alternativen der Realisierung entwickelt und getestet werden. Die Vielzahl der technischen Innovationen, aber auch die hohen Kosten des Projekts sind eine Folge dieses aus der amerikanischen Kriegsforschung stammenden Stils, der später auch zum Bestandteil des industriellen Managements wurde.

Um das Jahr 1950 waren mit der Inbetriebnahme des Whirlwind-I-Computers die technischen Voraussetzungen für die Umsetzung der von Forrester, Everett und Crawford propagierten Anwendungen des Digitalcomputers geschaffen. Mit

¹ Redmond and Smith 1980, S. 48; Forrester 1980, S. 226.

² Weiss 1991; Ridenour 1947; Taylor 1950, S. 1419.

³ Everett and Swain 1947, S. 2f.

⁴ Email von Rick Smith an den Autor, 9. Mai 1996; Rosen 1969, S. 9; Redmond and Smith 1980, S. 217.

3. Vom Analogrechner zum Minicomputer

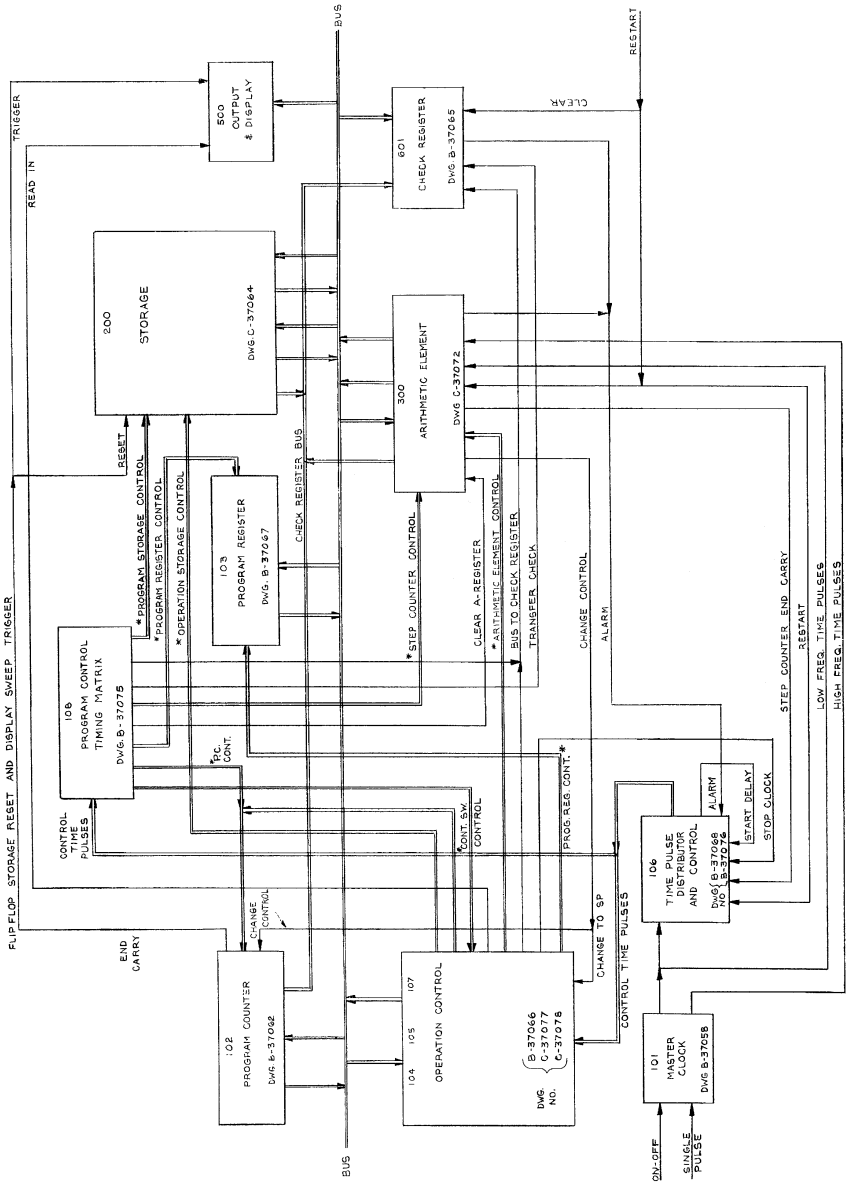


Abbildung 15. Detailliertes Blockdiagramm des Whirlwind-I-Computers (ohne Ein- und Ausgabe)

der nun einsetzenden anwendungsnahen Phase des Systementwurfs verschob sich allmählich auch die Perspektive der Ingenieure in bezug auf die Rolle des Menschen innerhalb des Computersystems. Während zu Beginn des *Project Whirlwind* der Mensch als ein Bestandteil eines geschlossenen Regelkreises verstanden wurde, trat er nun mit seinen kognitiven und sensorischen Fähigkeiten immer stärker ins Zentrum des Systementwurfs. Der daraus entstandene anthropozentrische Ansatz führte Ende der fünfziger Jahre schließlich zur Entwicklung kleiner Computer, der sogenannten Minicomputer, mit denen der Benutzer über geeignete Ein- und Ausgabegeräte interagieren konnte.

3.3 SAGE: Ein komplexes Mensch-Maschine-System

Ab 1950 wurde am MIT mit finanzieller Unterstützung durch die U.S. Air Force an der Entwicklung eines computergestützten Luftverteidigungssystems gearbeitet, das später den Namen *Semi-Automatic Ground Environment* (SAGE) erhielt und zum Prototyp aller weiteren militärischen *Command and Control Systems* wurde.¹ Parallel zur Arbeit der Valley-Kommission wurden im September 1950 erste Versuche zur Übertragung von Radardaten über Telefonleitungen durchgeführt. Im Frühjahr 1951 wurde am MIT das *Project Charles* initiiert, das eine Untersuchung des bestehenden Luftverteidigungssystems zum Inhalt hatte. Im April 1951 wurde erstmals ein automatisches Abfangen von Flugzeugen mit Hilfe des Whirlwind-Computers und einer Radarstation vorgeführt. Das Ergebnis der erfolgreichen Tests war die Einrichtung des *Project Lincoln*, das im Frühjahr 1952 in *Lincoln Laboratory* umbenannt wurde und ab 1953 in Lexington außerhalb von Boston angesiedelt war. Die Entwicklergruppe des MIT wurde als Division 6 in das Lincoln Laboratory eingegliedert, blieb aber bis 1954 in Cambridge. Im Februar 1952 wies Luftwaffenminister Thomas K. Finletter dem Luftverteidigungsproblem höchste Priorität zu und sicherte dem MIT unbegrenzte finanzielle Unterstützung zu.²

Im Juni 1952 wurde schließlich mit der Entwicklung des sogenannten Cape-Cod-Systems begonnen, einer realistischen Simulation des geplanten landesweiten Luftverteidigungssystems. Gleichzeitig wurde ein industrieller Hersteller für den Nachfolger des Whirlwind-Computer gesucht und schließlich in IBM gefunden. Das erste von 23 SAGE Befehlszentren wurde im Juni 1958, das letzte im Jahre 1963 in Betrieb genommen.³

¹ Edwards 1996, S. 107ff.

² Jacobs 1983, S. 328f.; Hughes 1998, S. 18–30.

³ Jacobs 1983, S. 329; Anonymous 1958 – Das SAGE-System war zum Zeitpunkt seiner Inbetriebnahme allerdings durch die Einführung von interkontinentalen ballistischen Raketen mit einer Vor-

3. Vom Analogrechner zum Minicomputer

War schon der Whirlwind-I-Computer ein technisches Artefakt von bislang nicht gekannter Komplexität, das neue Entwurfs- und Managementmethoden erforderte, so galt dies in noch größerem Maße für das Luftverteidigungssystem SAGE. Zu den Hardwareproblemen traten im Rahmen des SAGE-Systems vor allem die Fragen der Softwareentwicklung, der Kommunikation von Computern untereinander sowie zwischen Mensch und Computer. Damit wurde es erstmals notwendig, ein komplexes Mensch-Computer-System zu gestalten, in dessen Zentrum ein Digitalcomputer stand. Die Entwickler griffen dazu auf Erkenntnisse aus verschiedenen wissenschaftlichen Bereichen zurück: der Kybernetik mit ihrer Theorie der Rückkopplung, der Psychologie mit ihren Erkenntnissen über die menschliche Wahrnehmung und Informationsverarbeitung und der militärischen Planungsmethoden, die nach dem Zweiten Weltkrieg in sogenannten *Think Tanks* wie der RAND Corporation entwickelt worden waren.¹

3.3.1 Mensch-Computer-Kommunikation

Die Theorien der Kybernetik wurden – ähnlich wie die Computertechnik – zunächst entwickelt, um bislang vom Menschen ausgeführte militärische Tätigkeiten zu rationalisieren. Eine vollständige Automatisierung der meisten dieser Tätigkeiten war allerdings mit den technischen Möglichkeiten der vierziger und fünfziger Jahren nicht realistisch. Stattdessen übernahm der Computer einen Teil der Aufgabe, während der Mensch, häufig in enger Verbindung mit der Maschine, den Rest erledigte. Eine effektive Gestaltung des Gesamtsystems hing deswegen von einer effektiven Mensch-Maschine-Kommunikation ab. Dabei war es hilfreich, daß die Kybernetik eine Beschreibung menschlicher Tätigkeit anbot, die eine Parallele zwischen Mensch und Computer zog.² Robert Everett charakterisierte die Aufgabenteilung zwischen Mensch und Maschine innerhalb des SAGE-Systems 1957 folgendermaßen:

»Although SAGE has made many of the data-processing functions in a direction center automatic, many tasks remain that are better performed by the man. Operators ... can recognize certain patterns more rapidly and meaningfully than any of our present computers and take appropriate action. The most important, operators are required for tactical judgements such as aircraft identification or weapon deployment and commitment.

warnzeit von nur wenigen Minuten strategisch wertlos. Vgl. Fellows 1981, S. 59. Dennoch wurde erst 1983 der letzte AN/FSQ-7-Computer in einem SAGE-Befehlszentrum außer Betrieb genommen. Vgl. Bell 1983.

¹ Edwards 1996, Chapter 4; Benington, in Tropp et al. 1983, S. 385; Baum 1981; Smith 1966.

² Zum Beispiel in Wiener and Campbell 1954 oder Solow 1956.

3.3 SAGE: Ein komplexes Mensch-Maschine-System

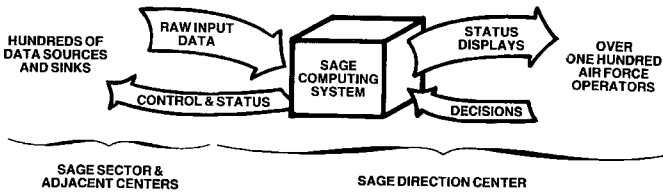


Abbildung 16. Datenfluß im SAGE-System

(...) [The] computer can rapidly summarize and filter ... data for individual presentation to more than 100 air force personnel who both assist and direct air-defense operations.«¹

Nachdem mit dem Whirlwind ein Computer zur Verfügung stand, der für die automatische Verarbeitung großer Datenmengen in Echtzeit leistungsfähig genug war, wurde die Gestaltung der Ein- und Ausgabe zur kritischen Größe des entstehenden Mensch-Maschine-Systems. Das Resultat der Bemühungen, die Datenein- und -ausgabe im Dialog mit dem Computer, stellte eine völlig neue Qualität in der noch jungen Datenverarbeitung dar. Bei allen bisherigen Maschinen, sofern es sich um speicherprogrammierbare Computer handelte, waren die Programme und die zu verarbeitenden Daten mit Hilfe von Lochkarten oder Lochstreifen eingelesen worden, einer Eingabetechnik, deren Ursprung bis auf die Webstuhlsteuerung von Jacquard zurückgeht und von Hollerith 1890 in die Datenverarbeitung eingeführt wurde. Die Verwendung von Lochkarten zog den sogenannten Stapel- oder Batchbetrieb nach sich, bei dem jeder Auftrag vor der Bearbeitung mit *allen* benötigten Daten vorliegen muß.² Für ein System wie SAGE, dessen Hauptaufgabe die Überwachung und Interaktion mit der Umwelt war, mußten demnach Schnittstellen für den Dialog zwischen Computer und Umwelt geschaffen werden.³

Der interaktive Computerbetrieb, für den es zu Beginn der fünfziger Jahre noch keine feste Bezeichnung gab, stand in enger Wechselwirkung zu einer zentralen Überlegung der Kybernetik. Die These, daß sich das menschliche Verhalten nicht grundsätzlich von dem der Maschine unterscheidet, war zunächst nur Ausgangspunkt für eine nach wissenschaftlichen Methoden gestaltete Kommunikation zwischen Mensch und Maschine gewesen. Die Existenz von Computern, die – zunächst zwar nur in rudimentärer Form – in Interaktion mit dem Menschen traten, bestärkte viele Wissenschaftler in der Vorstellung, mit genügendem

¹ Everett et al. 1957, S. 150.

² Lindner et al. 1984, S. 60ff.; Campbell-Kelly and Aspray 1996, S. 175, 207f.

³ Die Programme wurden allerdings weiterhin auf althergebrachte Weise mittels Lochkarten in den Computer eingelesen.

3. Vom Analogrechner zum Minicomputer

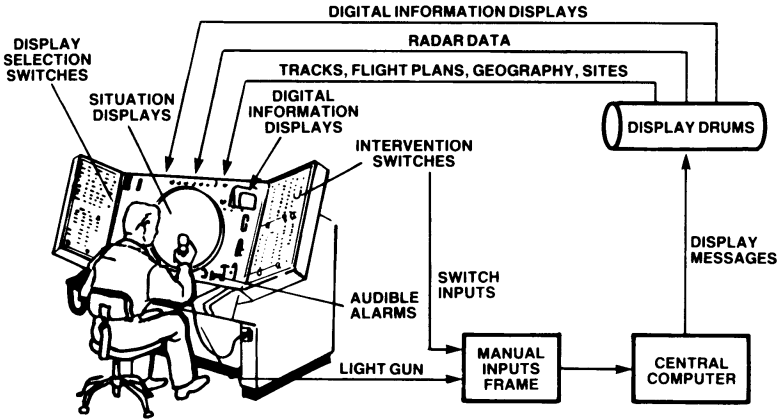


Abbildung 17. Funktionen des Menschen im SAGE-System

Aufwand könne man alle kognitiven Prozesse des Menschen mit dem Computer nachbilden.¹

Der Wert eines Systems wie SAGE wurde vor allem vom verzögerungsfreien Fluß von Daten durch das System bestimmt (Abb. 16). Der Computer empfing kontinuierlich Daten von Hunderten von Datenquellen, vor allem Radarstationen innerhalb des zu überwachenden Sektors und von benachbarten SAGE Befehlszentren. Zu diesem Zweck mußten Konverter entwickelt werden, die die analogen Radarsignale in digitale Signale umwandeln konnten. Für die Übertragung von Daten über Telefonleitungen wurden das Modem (*Modulator/Demodulator*) und eine Reihe von Codierungs- und Datenkompressionsverfahren entwickelt, mit deren Hilfe das für analoge Sprachsignale gedachte Telefonnetz auch zur sicheren und schnellen Übertragung von Binärdaten benutzt werden konnte. Die Entwicklungen in diesem Bereich stellen den Ausgangspunkt für alle späteren Datennetze dar.²

Wichtiger für die Entwicklung des interaktiven Computers war allerdings die Entwicklung einer Bedienerkonsole mit Ein- und Ausgabegeräten, die, teilweise in weiterentwickelter Form, bis heute Bestandteile eines Computersystems sind. Dies sind Eingabegeräte wie die alphanumerische Tastatur und Zeigeinstrumente (*pointing devices*) zur Eingabe von grafischen Daten sowie die Kathodenstrahlröhre als Ausgabegerät. Über die Bedeutung einer sorgfältig gestalteten Konsole

¹ Zum Beispiel in Newell and Simon 1958.

² Wieser 1983; Hellige 1992; Hellige 1994.

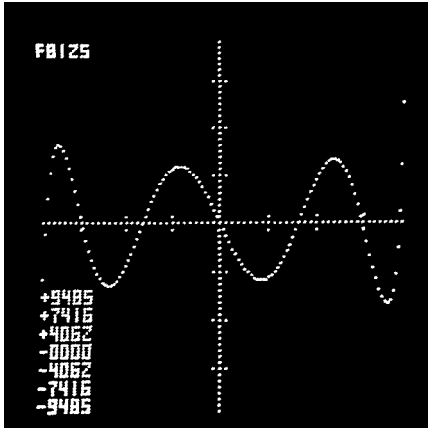


Abbildung 18. Ausgabe auf dem Bildschirm des Oszilloskops

für die Mensch-Computer-Kommunikation schrieb C. Robert Wieser, der Verantwortliche für den Aufbau des Cape-Cod-Systems, im Rückblick:

»[The] Construction of a realistic direction center depended heavily on the development of a versatile display console that provided the operators the information they needed to make decisions and also provided them the means to send commands to the computer. (...) What was wanted was a computer-generated PPI (=Plan Position Indicator, d. Autor) display that would include alphanumeric characters (for labels on aircraft tracks) and a separate electronic tote-board status display. The operators had to be able to select display categories of information ... without being distracted by all the information available.«¹

Die von Everett erwähnte Assistenzfunktion des Computers bestand demnach in einer grafischen Darstellung des überwachten Luftraums, bei der die erfaßten Flugzeuge mit Informationen über Geschwindigkeit, Höhe und Freund-/Feindkennung in einer Kartendarstellung zusammengefaßt wurden. Eine solche Darstellung ermöglichte dem geübten Offizier, in Anlehnung an ältere taktische Hilfsmittel, eine Lagebeurteilung *auf einen Blick*. Aus diesem Grund wurde beim SAGE-System erstmals in größerem Umfang die Kathodenstrahlröhre als Computerausgabegerät verwendet.

Die Kathodenstrahlröhre war zu Beginn der fünfziger Jahre bereits eine weitverbreitete elektronische Systemkomponente. Als Bildröhre war sie seit Anfang der dreißiger Jahre zum Zentrum des Fernsehempfängers geworden. Auch in der

¹ Wieser 1983, S. 367.

3. Vom Analogrechner zum Minicomputer

Elektronik wurde die Kathodenstrahlröhre als Oszillographenschirm zum Kern einer neuen Form von Meßgerät. Seit Beginn der vierziger Jahre wurde sie schließlich auch als Radarschirm verwendet. Die Benutzung der Kathodenstrahlröhre als Ausgabegerät in der Computertechnik geht im wesentlichen auf die beiden letztgenannten Bereiche zurück.¹

Bereits während des *Project Whirlwind* waren Kathodenstrahlröhren in Testschaltungen verwendet worden, mit denen der Inhalt der Speicherröhren überprüft und Programme getestet wurden.² Es zeigte sich aber, daß sich die Oszilloskopanzeige auch als reguläres Ausgabegerät verwenden ließ. Nach der Erinnerung von Norman Taylor wurden diese Fähigkeiten zunächst spielerisch verwendet, etwa im Rahmen von Vorführungen für Besucher, denn es war klar, »[that] displays attracted potential users – computer code did not«. ³

Es fanden sich aber auch sehr schnell ernsthaftere Anwendungen, wie die in Abbildung 18 gezeigte grafische Darstellung von Funktionen bei mathematischen Berechnungen verdeutlicht. Zu diesem Zweck wurden Digital-Analog-Konverter an zwei der Flipflop-Register angeschlossen, in denen die x- und y-Koordinaten des auszugebenden Punktes gespeichert wurden. Auf diese Weise konnten Buchstaben und Grafiken in einfacher Matrixdarstellung auf dem Bildschirm erzeugt werden.⁴ Solch eine Bildschirmausgabe ist allerdings nur für Darstellungen geeignet, die aus sehr wenigen Bildpunkten bestehen. Da die Bildschirmsignale nicht zwischengespeichert wurden, mußten sämtliche Bildpunkte immer wieder neu berechnet und in den für die Ausgabe verwendeten Registern abgelegt werden. Damit wurde ein beträchtlicher Teil der Rechenleistung für die Bildschirmausgabe verwendet. Für ein Luftraumverteidigungssystem mit einer Vielzahl von darzustellenden Informationen und zeitkritischen Berechnungen war ein solches Verfahren nicht sinnvoll.⁵

Für die Konsole des Cape-Cod-System machte man sich deswegen auf die Suche nach einer ausgereiften Kathodenstrahlröhre und fand sie im sogenannten Charactron, das von der *Convair Co.* in San Diego entwickelt worden war, einem Elektronikunternehmen aus dem Umkreis der Flugzeugindustrie.⁶ Von seinem Aufbau her war das Charactron eine konventionelle Kathodenstrahlröhre,

¹ Soller et al. 1948.

² Ross 1988, S. 61f.; Everett 1951; Dodd et al. 1950; S. 994f. – Dieses indirekte Vorgehen war notwendig, da der phosphoreszierende Belag bei der Speicherröhre durch ein (undurchsichtiges) Dielektrikum ersetzt war.

³ Taylor, in Hurst et al. 1989b, S. 20. Auf ähnlich zufällige und spielerische Weise wurden auch die Speicherröhren des EDSAC im englischen Cambridge als Ausgabemedien entdeckt. Vgl. Campbell-Kelly 1998.

⁴ Everett 1980, S. 735f.

⁵ Sutherland 1966, S. 95. Laut Norman Taylor rechnete man bei der Planung von SAGE mit über 400 zu überwachenden Flugzeugen. Vgl. Taylor, in Hurst et al. 1989b, S. 22.

⁶ Taylor, in Hurst et al. 1989b, S. 22f.; Anonymous 1949b; McNaney 1942.

3. Vom Analogrechner zum Minicomputer

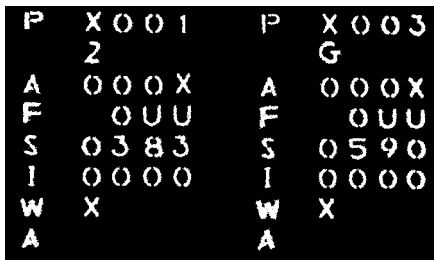


Abbildung 20. Anzeige des Digital Information Display

bei der eine metallische Schablone mit 64 verschiedenen Symbolen in den Strahlengang eingesetzt war (Abb. 19, oben). Der Elektronenstrahl wurde zunächst so abgelenkt, daß er durch das gewünschte Symbol auf der Schablone fiel, dann refokussiert und ein zweites Mal auf die gewünschte Stelle des Bildschirms abgelenkt wurde.¹ Obwohl das Prinzip des Charactron relativ einfach erscheint, gab es erhebliche optische Probleme, weil der Elektronenstrahl nach dem Durchgang durch die Schablone wieder so fokussiert werden mußte, daß das Zeichen an jeder Stelle des Bildschirms scharf dargestellt werden konnte.²

Mit einer Charactron Röhre von 19 Zoll Durchmesser wurde schließlich das *Situation Display*, das Kernstück der Bedienerkonsole realisiert, auf dem alle Luftraumdaten grafisch dargestellt wurden. In Abbildung 19 ist auch die Darstellung der Küstenlinie von Neuengland und wichtiger Radarstellungen auf dem *Situation Display* wiedergegeben. Dabei ist die Küstenlinie aus Linienstücken unterschiedlicher Neigung zusammengesetzt, die Bestandteile des Charactron-Zeichensatzes waren.³ Der Computer aktualisierte alle 2,5 s die Ausgabedaten für alle 82 Bedienerkonsolen, bestehend aus 18 000 Punkten und 5 000 Linien. Um den Computer nicht unnötig mit Ein- und Ausgabearbeiten zu beschäftigen, wurden diese Daten auf einer Magnettrommel mit einer Kapazität von 26 000 Worten zwischengespeichert (vgl. Abb. 17).⁴

Neben dem *Situation Display* war noch ein weiterer Bildschirm in die Bedienerkonsole eingebaut. Das *Digital Information Display* wurde benötigt, um Statusinformationen oder die aktuelle Wetterlage anzuzeigen (Abb. 20). Dazu wurde eine Typotron-Röhre mit einem Durchmesser von 4 Zoll verwendet, die von der *Hughes Products Co.* hergestellt wurden. Das Typotron arbeitete im wesentlichen nach dem gleichen Prinzip wie das Charactron, war aber eine Speicherröhre. Die darzustellende Information mußte also nur *einmal* vom Computer zum Typotron

¹ Wieser 1983, S. 367; Carroll 1956, S. 146.

² Taylor, in Hurst et al. 1989b, S. 23; Wieser 1983, S. 367.

³ Taylor, in Hurst et al. 1989b, S 25; Carroll 1956.

⁴ Everett et al. 1957, S. 153; Taylor, in Hurst et al. 1989b, S. 24.

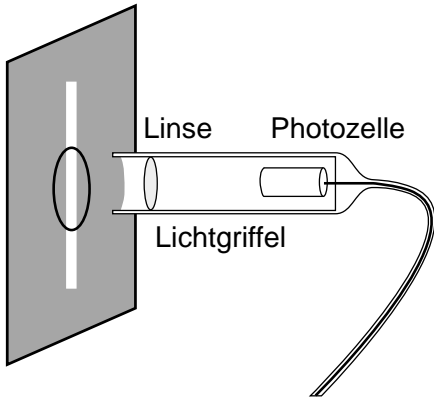


Abbildung 21. Aufbau eines Lightpen

übertragen werden und wurde dann so lange angezeigt, bis ein neues Zeichen übertragen wurde. So konnte der Datenverkehr zwischen Computer und Ausgabegerät für nicht zeitkritische Daten minimiert werden.¹

Ähnlich wie die Kathodenstrahlröhre wurde auch das erste grafische Eingabegerät, die *Lightgun* entwickelt, um – zusammen mit einem Oszilloskop – die Speicherröhren des Whirlwind-Computers zu testen. Zu diesem Zweck wurde ein Programm gestartet, das jede Speicherzelle einzeln auslas. Wurde eine EINS gelesen, wurde der Programmablauf fortgesetzt, ansonsten stoppte es. Um in diesem Fall die betreffende Speicherzelle identifizieren zu können, ließ Robert Everett Ende 1948 die erste *Lightgun* bauen.²

Die *Lightgun* bestand ebenso wie ihr Nachfolger, der Lichtgriffel (*Lightpen*), aus einem Tubus, in den eine Linse und eine Photozelle als Empfänger eingebaut waren (Abb. 21). Wenn man die *Lightgun* auf den Bildschirm richtete, registrierte die Photozelle immer dann einen Lichtimpuls, wenn ein Bildschirmpunkt innerhalb ihres Sichtfeldes neu geschrieben oder aufgefrischt wurde. So ließ sich mit geringem schaltungstechnischem Aufwand feststellen, auf welchen Punkt des Bildschirms die *Lightgun* gerichtet war.³

Als 1951 erste Versuche durchgeführt wurden, die Flugbahn eines Flugzeuges mit dem Computer zu verfolgen, stellte sich die Frage, wie man bestimmte Flugzeuge bzw. deren Flugbahn auswählen könnte, um ihnen bestimmte Eigenschaften, etwa eine Freund-/Feindkennung zuzuweisen, oder weitere Daten wie Höhe und Geschwindigkeit abzurufen. Auf einen Vorschlag von Robert Everett

¹ Taylor, in Hurst et al. 1989b, S. 23; Carroll 1956, S. 146.

² Taylor, in Hurst et al. 1989b, S. 20.

³ Newman and Sproull 1979, S. 152ff.; Everett 1980, S. 376.

3. Vom Analogrechner zum Minicomputer



Abbildung 22. *Lightgun* (1953, links) und *Lightpen* (siebziger Jahre, rechts)

hin wurde hierfür die *Lightgun* verwendet.¹ Ende der fünfziger Jahre wurde die relativ klobige *Lightgun* durch Ben Gurley von der *Digital Equipment Corporation* zum Lichtgriffel weiterentwickelt, der nur noch eine Länge von 14 cm und einen Durchmesser von 1 cm hatte und weniger als 30 Gramm wog (Abb. 22). Der Lichtgriffel blieb in dieser Form bis weit in die siebziger Jahre hinein das wichtigste Eingabeinstrument für Grafikanwendungen (z. B. Computer-Aided Design).²

Der Whirlwind bzw. der AN/FSQ-7-Computer waren so programmiert, daß auf dem *Situation Display* auch die Namen einer Reihe von Standardfunktionen angezeigt wurden. Wenn der Bediener ein Objekt auf dem Bildschirm mit der *Lightgun* markierte und dann eine dieser Funktionen aktivierte, wurde diese auf das vorher angewählte Objekt angewendet. Die auf dem Bildschirm ausgegebenen Daten, bzw. die von ihnen repräsentierten Objekte wurden also unmittelbar für die weitere Eingabe verwendet. So war es z. B. sehr schnell möglich, Höhe und Geschwindigkeit eines Flugzeugs abzufragen.³

Die Anordnung der Bedienelemente, zu denen neben dem *Situation Display*, dem *Digital Information Display* und der *Lightgun* auch eine Reihe von Schaltern und Tastern gehörte, erfolgte weitgehend nach einem experimentellen Verfahren. Verschiedene Testpersonen wurden gebeten, an einem mit Hilfe von Kar-

¹ Everett, in Tropp et al. 1983, S. 391f.

² Gilmore, in Hurst et al. 1989a, S. 45; Gurley and Woodward 1959.

³ Perry and Voelcker 1989.



Abbildung 23. Operator mit Lightgun an einem SAGE-Bildschirmterminal

tons hergestellten Modell typische Aufgaben der Luftüberwachung durchzugehen. Danach wurden sie nach Verbesserungen in der Anordnung, z. B. die Lage und Neigung der Bildschirme, befragt. Die so ermittelten Werte wurden gemittelt und gingen in das endgültige Layout der Bedienkonsole ein, die gleichfalls von IBM gefertigt wurde. Auch die Arbeitsplatzgestaltung wurde also als Mittel zur Steigerung der Leistungsfähigkeit des Menschen im Mensch-Maschine-System erkannt, obwohl die dazu verwendeten Verfahren aus heutiger Sicht noch wenig differenziert und theoretisch fundiert erscheinen.¹

SAGE war aber nicht nur in Hinblick auf die Rechnerhardware und die Ein- und Ausgabetechnik wichtig für die Entwicklung des interaktiven Computers, sondern auch in Hinblick auf die Programme, die Software.

3.3.2 Systematische Programmentwicklung

»The computer programming was probably the most underestimated task in the entire SAGE project.«²

¹ Wieser 1983, S. 368. – Die ergonomische Arbeitsplatzgestaltung wurde erst im Zusammenhang mit der Entwicklung von komplexen Waffensystemen für die (amerikanische) Luftwaffe seit Mitte der fünfziger Jahre zum Gegenstand systematischer wissenschaftlicher Untersuchungen. Vgl. Sheridan 1986, S. 1f.

² Jacobs, in Tropp et al. 1983, S. 386.

3. Vom Analogrechner zum Minicomputer

Diese Äußerung von John F. Jacobs wirft ein bezeichnendes Schlaglicht auf das Verhältnis zwischen Computerhardware und -software in den fünfziger Jahren. Computerentwicklung hieß vor allem mathematischer Entwurf von Rechnerarchitektur oder ingenieurmäßige Entwicklung von elektronischen Schaltkreisen. Daß der Computer seine wahre Stärke und Flexibilität aber erst durch seine Programmierung erhält, blieb weitgehend unbeachtet. Programmierung war dementsprechend bis in die fünfziger Jahre eine untergeordnete, wenn nicht gar als geringwertig erachtete Tätigkeit. Dies läßt sich unter anderem daran ablesen, daß die Programmierung in den vierziger und frühen fünfziger Jahren vorwiegend eine Frauentätigkeit war¹ und daß es bis Mitte der fünfziger Jahre so gut wie keine Literatur über die Programmierung von Digitalrechnern gab.²

Ebenso wie andere Computerpioniere sahen auch die Mitglieder des *Project Whirlwind* das Programm, zunächst noch als »code« bezeichnet, als Bestandteil des Computers, als auswechselbares Element, das es erlaubte, verschiedene Aufgaben mit der gleichen Maschine zu bearbeiten. Die Akteure waren zu dieser Zeit noch Geräteentwickler, Programmierer und Anwender in einer Person und gaben ihre Programme direkt im Maschinencode (d. h. im Fall von Whirlwind als 16-stellige Binärzahlen) in die Maschine ein. Angesichts des anfangs noch geringen Problem- und Programmumfangs stellte dies zunächst keine echte Einschränkung dar.³

Nachdem der Whirlwind-Computer ab 1951 sowohl für Berechnungen am MIT als auch für das anlaufende SAGE-Projekt verwendet wurde, wurden bald Werkzeuge entwickelt, die die Programmentwicklung unterstützen sollten. Am MIT entstand die *Science and Engineering Computation Group* unter der Leitung von Charles W. Adams und Jack T. Gilmore, Jr., die eine Bibliothek mit vielverwendeten Unterprogrammroutrinen erstellte, das sogenannte *comprehensive system*, das schon deswegen nötig war, weil der eingeschränkte Befehlssatz bereits bei einfachen mathematischen Problemen durch einen erheblichen Programmaufwand kompensiert werden mußte.⁴ Außerdem entwickelten sie ein Programm, das man heute als symbolischen Assembler bezeichnen würde. Dabei mußte der Programmierer seinen Code zwar immer noch als Folge von Maschinenbefehlen formulieren. Er mußte diese aber nicht mehr als 16-stellige Binärzahlen codieren, sondern er konnte besser zu behaltende symbolische Notationen, sogenannte Mnemonics verwenden, die von einem Hilfsprogramm in den entsprechenden Bi-

¹ Hoffmann 1987; Goyal 1996 und in bezug zu SAGE Benington, in Tropp et al. 1983, S. 386.

² Die wichtigste Ausnahme war das Buch von Wilkes, Wheeler und Gill (1951) über die Programmierung des englischen EDSAC.

³ Rosen 1967, S. 4.

⁴ Rosen 1967, S. 4.

närcode umgewandelt wurden. Dadurch konnte ein Großteil der Schreibfehler bei der Programmerstellung vermieden werden.¹

Bei der Entwicklung des SAGE-Systems stellten sich weitere, nicht weniger wichtige Fragen. Zunächst mußten eine Vielzahl von Ein- und Ausgabegeräten mit höchst unterschiedlichen Eigenschaften, vor allem in Hinblick auf ihre Geschwindigkeit vom zentralen Computer verwaltet werden, ohne daß es zu Datenverlusten kam. Zu diesem Zweck wurde eine Anzahl von Routinen entwickelt, die das sogenannte *control program* bildeten, das man als eine Art einfaches Betriebssystem bezeichnen könnte, obwohl eine Trennung zwischen System- und Anwendungsprogrammen bei einem komplexen System wie SAGE nicht einfach ist und auch während der fünfziger Jahre noch nicht vorgenommen wurde.²

Allein der Umfang der zu entwickelnden Programme machte eine völlig neue Herangehensweise nötig. Die Programme für das Cape-Cod-System, das immerhin einen Umfang von etwa 20 000 Instruktionen hatte, wurde noch von Robert Wiesers Gruppe am *Lincoln Laboratory* bewältigt. Wie sehr der Aufwand der Programmentwicklung zunächst unterschätzt wurde, mag man an der Tatsache ablesen, daß diese Gruppe schließlich 120 Mitarbeiter hatte, gegenüber einer durchschnittlichen Gruppenstärke von zehn Personen.³

Ebenso wie für die Produktion des AN/FSQ-7-Computers wurde auch für die Entwicklung der Programme für das SAGE-System ein Vertragspartner gesucht, nachdem sich das MIT bzw. das Lincoln Lab angesichts des zu erwartenden Aufwands gegen eine eigene Entwicklung ausgesprochen hatte. IBM lehnte ein entsprechendes Angebot ab, da man für diese Aufgabe 2 000 neue Programmierer hätte einstellen müssen und dies zu einer Zeit, als es in den USA nicht mehr als 3 000 bis 4 000 professionelle Programmierer gab. Auch die *Bell Telephone Laboratories* wollten die Programmentwicklung nicht übernehmen.⁴ Im Dezember 1955 gründete schließlich die *RAND Corporation*, einer der *Think Tanks* der amerikanischen Luftwaffe speziell für diese Aufgabe das erste Softwareunternehmen, die *System Development Corporation* (SDC) im kalifornischen Santa Monica. SDC engagierte in der Folge eine große Zahl von Computerlaien und bildete sie zu Programmierern aus. Damit veränderte sich das Umfeld, in dem Programmentwicklung stattfand. Es mußte eine Technik zur arbeitsteiligen Erstellung von Programmen entwickelt werden, bei der berücksichtigt wurde, daß

¹ Laning and Zierler 1954; Wildes and Lindgren 1985, S. 339; auch Wilkes 1985, S. 176f. – Statt 10010 x konnte der Programmierer `add x` schreiben, wenn er den Inhalt der Speicherzelle x zum Akkumulatorregister addieren wollte, wobei x für eine 11-stellige Binäradresse stand. Vgl. Bell and Newell 1971, S. 145.

² Rosen 1967, S. 17; Benington 1983, S. 355.

³ Wieser, in Tropp et al. 1983, S. 386; Everett 1980, S. 379.

⁴ Benington und Crago, in Tropp et al. 1983, S. 385f.

3. Vom Analogrechner zum Minicomputer

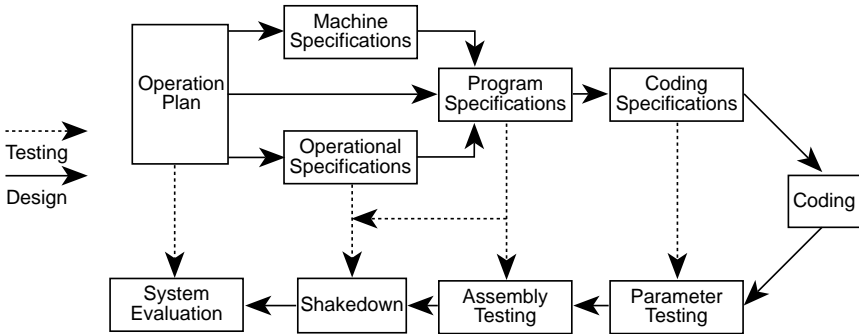


Abbildung 24. Programmentwicklungszyklus

die Programmierer keine Computerexperten oder Mathematiker waren.¹ Da auch das zu entwickelnde Computerprogramm eine bislang nicht erreichte Komplexität haben würde, bot es sich an, eine ähnliche Managementstrategie anzuwenden, wie schon beim *Project Whirlwind*.²

Im Ergebnis wurde die Programmentwicklung in eine Vielzahl von Einzelschritten aufgeteilt und durch Rückkopplungsmechanismen gesteuert (Abb. 24). Ohne zu sehr auf die Details eingehen zu wollen, kann man diese Vorgehensweise als einen Top-Down-Ansatz und somit als Vorläufer des strukturierten Programmierens bezeichnen. Zu Beginn wurden gemeinsam von den Programmierern und den späteren Anwendern in einem *operation plan* die allgemeinen Anforderungen definiert. Unter Berücksichtigung der verwendeten Hardware (*Machine Specifications*) und der Eigenschaften der existierenden Systemroutinen (*Operational Specifications*) wurden dann die *Program Specifications* festgelegt, eine Art Ablaufdiagramm des zu entwickelnden Programmes, die auch die Kommunikation (also die Schnittstelle) zwischen den einzelnen Komponenten festlegte. Das genaue Verhalten der Komponenten wurde in den *Coding Specifications* festgelegt, erst dann wurde das eigentliche Programm geschrieben. Die Eigenschaften jeder dieser Hierarchiestufen wurden im nachhinein an den Anforderungen der nächsthöheren Stufe überprüft, d. h. das Verhalten einer Routine wurde nicht an seinem Code, sondern an den *Coding Specifications* überprüft.³

Aufgrund dieser Methodik, die analog zu derjenigen des *Project Whirlwind* war, konnte die Programmentwicklung auch mit angelernten Programmierern bewältigt werden. Sie führte aber auch zu den negativen Erfahrungen, die in dem

¹ Benington, in Tropp et al. 1983, S. 386f.

² Benington 1983, S. 351.

³ Benington 1983, S. 356f.

eingangs zitierten Resümee von John F. Jacobs zum Ausdruck kommt. Die Vielzahl der Tests und Überprüfungen während der Programmentwicklung sollte dazu beitragen, einen möglich fehlerfreien Code zu produzieren, hatte aber einen erheblichen zeitlichen und finanziellen Aufwand zur Folge. So wurden die Software für das SAGE-System erst Anfang 1957 mit einer mehr als einjährigen Verzögerung fertiggestellt. Die Programme, für deren Entwicklung aus Effizienzgründen keine der damals aufkommenden höheren Programmiersprachen verwendet wurde, hatten schließlich einen Umfang von über 100 000 Instruktionen.¹ Die Kosten von mehr als 5 Mio. \$ lagen in der gleichen Größenordnung wie die der Hardware und widerlegten die anfängliche Überzeugung, daß sich Software schnell und preiswert erstellen und ändern ließe.²

Die meisten der skizzierten Techniken zur Programmentwicklung gingen schnell wieder verloren, weil sich das Interesse des Fachpublikums nach 1955 mehr auf die Entwicklung höherer Programmiersprachen wie FORTRAN, COBOL und ALGOL konzentrierte. Einen anderen Grund mag man darin sehen, daß die Tätigkeit des Programmierens eher den Charakter eines Handwerks als den einer Wissenschaft hat. Dabei spielt die individuelle Erfahrung eines Entwicklers scheinbar eine wesentlich wichtigere Rolle als bestimmte theoretisch fundierte Methodiken.³

Die großangelegte Programmentwicklung für SAGE führte in den fünfziger Jahren noch nicht zur Herausbildung einer eigenständigen Berufsgruppe. Sie führte aber den Verantwortlichen in Militär und Politik auf eindrucksvolle Weise die entscheidende Bedeutung der Software für den Erfolg oder Mißerfolg von Computersystemen vor Augen. Sie legte damit den Grundstein für die umfassende staatliche Unterstützung von Projekten zur Entwicklung von Software durch die 1958 gegründete *Advanced Research Projects Agency* während der sechziger Jahre, die Gegenstand von Kapitel 4 ist.

Mitte der fünfziger Jahre, als SAGE die anfänglichen Hürden überwunden und eine gewisse Eigendynamik entwickelt hatte, kam es zu personellen Änderungen. Jay Forrester, der seit mehr als zehn Jahren an der Front der Computerentwicklung gestanden hatte, trat 1956 von seinem Posten als Direktor der Division 6 des Lincoln Laboratory zurück und beschäftigte sich in der Folge als Professor für industrielles Management am MIT mit der Übertragung seines kybernetischen An-

¹ Everett et al. 1957, S. 152, 155; Benington 1983, S. 357; Jacobs 1986, S. 108ff., passim.

² Benington 1983, S. 350.

³ Benington 1983, S. 350ff.; Bernstein 1978; Weinberg 1971 – Zu einer ähnlichen Einschätzung gelangte auch Michael S. Mahoney in seiner historischen Untersuchung über das »Software Engineering«, das Ende der sechziger Jahre die gleichen Ziele und Methoden propagierte wie Herbert Benington zehn Jahre zuvor. Vgl. Mahoney 1990.

3. Vom Analogrechner zum Minicomputer

satzes auf soziale und wirtschaftliche Prozesse.¹ Aus der Division 6 des Lincoln Lab wurde 1958 schließlich die MITRE Corporation, ein weiteres staatlich finanziertes Forschungsunternehmen mit Robert Everett als technischem Direktor. Die Computerentwicklung am Lincoln Lab des MIT wurde von Schülern Forresters und Everetts in ihrem Sinne fortgeführt.

3.4 Whirlwinds Nachfahren

3.4.1 Memory Test Computer

Whirlwind I blieb nicht der einzige am MIT entworfene Computer, er hatte sogar eine ganze Reihe von fast ebenso einflußreichen Nachfolgern. 1951 hatten Harlan Anderson und Kenneth H. Olsen unter dem Namen *Memory Test Computer* (MTC) eine kleinere Kopie des Whirlwind entworfen, in dem die ersten Kernspeicher, aber auch weitere Komponenten – neue Logikschaltkreise, Ein- und Ausgabegeräte – für das SAGE-System getestet wurden.² Daneben wurde der MTC auch zur Aufbereitung von Radardaten, zur Erstellung von Programmen für den Whirlwind und weitere unterstützende Anwendungen eingesetzt. Als SAGE 1958 in Betrieb ging, wurde dieser »kleine Bruder« des Whirlwind-Computers im Februar 1958 stillgelegt.³

Benutzer des MTC konnten erstmals mehrere Stunden Arbeitszeit am Computer zugewiesen bekommen und erhielten damit die Möglichkeit, ihre Programmierarbeiten und Programmtests in unmittelbarer Interaktion mit dem Computer durchzuführen.⁴ Aus dieser neuen Form der Computernutzung entwickelten einige Computerentwickler eine neue Einstellung zur Rolle des Computers. Wesley A. Clark, ab Mitte der fünfziger Jahre einer der wichtigsten Computerkonstruktoren am MIT, schrieb dazu:

»It was in these sessions that I began to learn . . . many of the basic attitudes toward computers that I hold firmly to this day: Computers are tools; convenience of use is the most important single design factor. Big computers are for big jobs; small computers, for small jobs. (...) Digital computers should handle analog signals as well. And so on . . . «⁵

¹ Forrester 1989 – Das bekannteste Ergebnis aus dieser Phase von Jay Forresters Karriere dürfte die Entwicklung eines sogenannten »Weltmodells« sein, daß Anfang der siebziger Jahre von Dennis Meadows für die aufsehenerregende Studie über die Grenzen des Wachstums verwendet wurde. Vgl. Meadows 1972.

² Redmond and Smith 1980, S. 206; Papian 1953, S. 40f.; Olsen 1988.

³ Gilmore 1958, S. 39.

⁴ Clark 1988, S. 353f.

⁵ Clark 1988, S. 354.

Nach dieser Maxime konstruierten die Ingenieure des MIT ab Mitte der fünfziger Jahre ihre nächsten Computer.

3.4.2 Lincoln TX-0 und TX-2

1955 begannen Wesley Clark und Ken Olsen mit dem Bau eines neuen Computers, des *Lincoln Laboratory Test-Experimental Computers Model 0* (TX-0), in dem unter anderem die Verwendung von Transistorschaltkreisen und eines sehr großen Magnetkernspeichers getestet werden sollten.¹ Dieser Computer wurde im August 1956 fertiggestellt; parallel begannen sie mit der Konstruktion des ambitionierten TX-2 Computers, bei dem neue Konzepte der Systemorganisation, neue elektronische Bauelemente und Ein-/Ausgabegeräte zu einem besonders fortschrittlichen und leistungsfähigen Computer zusammengefaßt werden sollten.² Der TX-2, zu seiner Entstehungszeit einer der komplexesten und leistungsfähigsten Computer überhaupt³, wurde in den folgenden zehn Jahren am Lincoln Lab vor allem für die Entwicklung von Ein- und Ausgabegeräten, für Computergrafikanwendungen – darunter Ivan Sutherlands zukunftsweisendes Grafikprogramm Sketchpad (1963) – sowie für die Entwicklung von Time-Sharing-Betriebssystemen verwendet.⁴

Obwohl auch der TX-0 und der TX-2 Computer mit finanzieller Unterstützung des Militärs gebaut wurden, waren die Ziele völlig andere als z. B. beim *Project Whirlwind*. Vorrangiges Ziel war nicht mehr allein eine möglichst hohe Verarbeitungsgeschwindigkeit und Zuverlässigkeit. Unter Einbeziehung der Erfahrungen aus dem *Project Whirlwind* und der SAGE Entwicklung wurde nun verstärkt auf die Einfachheit und Flexibilität des Computersystems geachtet.⁵

Nachdem bereits für den AN/FSQ-7-Computer erwogen worden war, die Elektronenröhren durch Transistoren zu ersetzen, dies aber wegen deren noch geringer Zuverlässigkeit und schlechten Hochfrequenzeigenschaften wieder verworfen wurde⁶, war der TX-0 der erste am MIT gebaute Transistorcomputer und einer der ersten Transistorcomputer überhaupt. Mit dem Oberflächengrenzschicht-

¹ Zuvor hatten Clark und Olsen dem Management des Lincoln Lab einen Plan zum Bau eines großen Transistorcomputers, des TX-1 vorgelegt, der jedoch abgelehnt wurde. Vgl. Olsen 1988. Physisch war auch der TX-0 immer noch ein sehr großer Computer, der einen ganzen Raum ausfüllte (Abb. 25).

² Mitchell and Olsen 1956; Clark 1957; Papian 1957.

³ Lincoln TX-2: 30 000 Transistoren, 64k + 4k Hauptspeicher mit einer Zugriffszeit von 6 bzw. 5,5 μ s, 36 bit Wortlänge, 150 000 Additionen/s, 80 000 Multiplikationen/s, 64 Befehle. Vgl. Gilmore 1958, S. 40.

⁴ Sutherland 1963; Sutherland et al. 1969.

⁵ Clark 1957, S. 144.

⁶ Astrahan and Jacobs 1983, S. 343.

3. Vom Analogrechner zum Minicomputer



Abbildung 25. TX-0 und TX-2 Computer (um 1958). Auf der rechten Seite des Bedienpults ist ein Flexowriter zu erkennen.

Transistor (*Surface Barrier Transistor*) hatte Philco 1953 ein Bauelement entwickelt, das sich für Taktraten von mehreren Megahertz und damit für den Einsatz in Digitalcomputern eignete. Der TX-0 diente aber nicht nur als Versuchsfeld für die Transistorschaltkreise, sondern auch für einen neuentwickelten Kernspeicher mit der für diese Zeit außerordentlich großen Kapazität von 65 536 Worten à 18 bit.¹ Die Architektur des TX-0 war extrem einfach. Er verfügte über nur vier Befehle, drei adressierbare und einen programmierbaren Befehl.² Im Rechenwerk des TX-0 waren nur die Addition und die Komplementbildung fest verdrahtet, während Multiplikation und Division als Unterprogrammroutrinen ausgeführt waren.³

Andererseits war der TX-0 (wie auch der TX-2) reichhaltig mit Peripheriegeräten ausgestattet. Dabei griffen die Konstrukteure auf Entwicklungen für das SAGE-System zurück. Die Computer waren mit einem Bildschirm ausgestattet,

¹ Anonymous 1954; Mitchell and Olsen 1956; Pearson 1992, S. 6.

² Die drei adressierbaren Befehle waren: Lesen aus dem Speicher, Schreiben in den Speicher und bedingtes Verzweigen. Der vierte Befehl konnte durch das Setzen der restlichen 16 Bits für verschiedene Funktionen genutzt werden.

³ Clark 1990; Clark 1957, S. 143f.

über den alphanumerische Zeichen ausgegeben werden konnten. Da jedes Zeichen mit 6 bit kodiert (= 64 Zeichen) war, ging man beim TX-0 auf eine Wortlänge von 18 bit über, nachdem die bisherigen Computer 16- bzw. 32-bit-Maschinen gewesen waren.¹ Schließlich war – wie schon bei Whirlwind – ein Lautsprecher Bestandteil des Computers. Dieser war an ein einzelnes Bit des Akkumulatorregisters angeschlossen; damit wurde dem Benutzer die Möglichkeit zur akustischen Kontrolle des Programmlaufs gegeben.²

Für die Dateneingabe wurde neben dem Lichtgriffel vor allem der Flexowriter verwendet, eine Art Schreibmaschine, mit der Lochstreifen geschrieben und gelesen werden konnten. Obwohl es mit Hilfe des Flexowriters noch nicht möglich war, Programme am Bildschirm zu erstellen, wurde der Zyklus von Programmstellung, -test und -modifikation erheblich beschleunigt. In der Tat konnte der Programmierer beim TX-0 und TX-2 seine komplette Arbeit direkt an der Konsole durchführen.³ Damit unterschied sich die Arbeitsweise bei dieser Maschine erheblich vom Stapelbetrieb, der bei gängigen Großcomputer der Zeit – so auch beim IBM 704 im Rechenzentrum des MIT – allgemein üblich war.

Trotz seiner spartanischen Architektur erlangte der TX-0 einen großen Einfluß auf die weitere Entwicklung von interaktiven Computern. Zum einen unterlag er als reiner Forschungscomputer nicht der militärischen Geheimhaltung, so daß sämtliche konstruktive Details in der Fachpresse publiziert werden konnten.⁴ Vor allem aber wurde der TX-0 zu einem wichtigen Bestandteil in der Ausbildung von Studenten am MIT. Nach der Inbetriebnahme des TX-2 wurde der TX-0 schon im Juli 1958 außer Betrieb genommen und dem Fachbereich Elektrotechnik des MIT zur Verfügung gestellt. Obwohl der große Speicher (64k) in den TX-2 transferiert und durch einen kleineren Speicher (4k) ersetzt wurde⁵, machte er eine ganze Generation von Studenten mit den Möglichkeiten des interaktiven Computerbetriebs bekannt und vermittelte ihnen die zitierte neue Einstellung zum Computer, die als *Hands-On Imperative* bezeichnet wurde.⁶ Aus ihren Reihen gingen auch die ersten »Hacker« hervor, junge idealistische Computerenthusiasten, die erkannten, daß sich der interaktive Computer allein mit Hilfe der Programmierung zu einem Werkzeug mit fast grenzenlosen Möglichkeiten machen läßt.

¹ Olsen 1988.

² Levy 1984, S. 28f; Clark 1988, S. 353ff.; Email von Les Earnest an Community Memory: Discussion list on the History of Cyberspace, Re: Computer Noise, 24. Februar 1999.

³ Levy 1984, S. 28.

⁴ Beispielsweise sind der Vorstellung des TX-2 im Tagungsband der *1957 Western Joint Computer Conference* fast 30 Seiten gewidmet. Vgl. Clark 1957; Frankovich and Peterson 1957; Forgie 1957; Best 1957; Olsen 1957.

⁵ Gilmore 1958, S. 39.

⁶ Olsen 1988, S. 18.

3. Vom Analogrechner zum Minicomputer

Insgesamt war der TX-0 ein weiterer wichtiger Schritt auf dem Weg zu kleinen, interaktiven und persönlichen Computern. Man kann sogar in gewissem Maße Ken Olsens Bewertung teilen, der 1988 in einem Interview äußerte:

»[The TX-0] is a very close equivalent to the modern personal computer. Somebody sits down in front of the oscilloscope, with a light pen and plays games, does things, is creative. Word processing wasn't yet developed. Games weren't yet well developed. But in general it was, you might say, one of the first personal computers.«¹

Obwohl diese Einschätzung gewiß ein wenig überspitzt ist, ist es doch eine Tatsache, daß es Ende der fünfziger Jahre im Umfeld des MIT eine kleine Gemeinschaft von Computerentwicklern und studentischen Programmierern entstanden war, die während ihrer universitären und beruflichen Laufbahn kaum mit der sonst üblichen Stapelverarbeitung in Berührung gekommen war.² Diese Gruppe trat in den folgenden Jahren mit sehr viel Erfolg für den interaktiven Betrieb von vergleichsweise kleinen Computern und für eine Aufwertung der Programmierung zu einem eigenständigen Entwicklungsfeld ein. Viele der frühen Hacker haben dann in den sechziger und siebziger Jahren wichtige Positionen in aufstrebenden Computerunternehmen oder Forschungszentren eingenommen.

3.4.3 Digital Equipment und der Minicomputer

Im Mai 1957 glaubten Kenneth und Stanley Olsen sowie Harlan Anderson schließlich, daß sich interaktive Computer, wie sie sie am MIT gebaut hatten, mit wirtschaftlichem Erfolg vermarkten ließen und gründeten die *Digital Equipment Corporation* (DEC). Der Schritt, sich Ende der fünfziger Jahre mit der Herstellung von Computern selbständig zu machen, war nicht ohne Risiko. Die Computerindustrie hatte Mitte der fünfziger Jahre ihre erste Krise mit einer Reihe von Unternehmenszusammenbrüchen hinter sich und noch 1957 hatte das Wirtschaftsmagazin *Fortune* berichtet, daß sich mit Computern kein Geld verdienen ließe.³ Hinzu kam, daß die Idee eines kleinen, interaktiven Computers zu dieser Zeit – außer in bestimmten akademischen Kreisen – wenig Anklang fand:

»The concept of an interactive computer was strange. Some people thought it was wrong. Almost spoke in ethical terms. Computers are serious, you shouldn't treat them lightly. You shouldn't have fun with them. They

¹ Olsen 1988, S. 18; vgl. auch Johnstone 1992.

² Clark 1988, S. 357; Levy 1984, S. 25f.; Corbató 1989/90.

³ Harris 1957, S. 139. – Dort heißt es zu den Problemen der Computerindustrie lapidar: »*Profits: There are none, not even for I.B.M.*«.



Abbildung 26. DEC PDP-1/A-Prototyp (ca. 1960)

shouldn't be exciting. They should be formal and distant with red tape involved.«¹

Die vergleichsweise ungünstigen Rahmenbedingungen lassen sich auch an der Entwicklung von DEC ablesen. Das Startkapital von 70 000 \$ wurde Ken Olsen von der *American Research & Development Company* zur Verfügung gestellt, einem der ersten Unternehmen, dessen Ziel die Wagnisfinanzierung junger Unternehmen war. Die ersten von DEC hergestellten Produkte waren Transistorschaltkreise, Module genannt, die unmittelbar auf den von Olsen für den TX-0 entwickelten Schaltkreisen aufbauten. Gedacht waren sie für die Verwendung im Entwicklungslabor, wo sie zu komplexeren logischen Schaltungen verschaltet werden konnten. Größter Abnehmer war das MIT, waren doch die Module in ihrer Schaltung, Geschwindigkeit und ihrem Signalpegel identisch mit den im Lincoln Lab verwendeten Schaltkreisen.²

Aus den DEC-Systemmodulen wurde schließlich auch der erste Minicomputer³ aufgebaut, der im Dezember 1959 unter dem Namen *Programmed Da-*

¹ Olsen 1988, S. 18.

² Pearson 1992, S. 10ff.; Olsen 1988; Best et al. 1978.

³ Die Bezeichnung Minicomputer wurde allerdings erstmals für den 1963 auf den Markt gebrachte PDP-5 verwendet. Vgl. Bell and Bell 1983, S. 555.

3. Vom Analogrechner zum Minicomputer

Tabelle 1. Daten für ausgewählte kommerzielle Computer, 1961

Computer	Addition (μ s)	Preis (US\$)	Wort- länge	Anzahl der Befehle
IBM 7090 [†]	4	3 630 000	36	187
PDP-1 [†]	5	120 000	18	25
IBM 704	24	1 994 000	36	91
IBM 709	24	2 630 000	36	187
IBM 650*	ca. 2736	250 000	34	89
Librascope LGP 30*	ca. 5500	55 860	32	16
Bendix G-15*	ca. 7520	77 300	29	125
Whirlwind I	24		16	27

[†] Transistorcomputer * Bitserielle Architektur, Magnettrommelspeicher

ta *Processor-1* (PDP-1)¹ auf der *Western Joint Computer Conference* in Boston vorgestellt wurde. Aufgrund seiner architektonischen und technologischen Herkunft könnte man diesen Rechner als die kommerzielle Variante des TX-0 bezeichnen. Der PDP-1 war mit einer Verarbeitungsgeschwindigkeit von 100 000 Additionen/s bzw. 40 000 Multiplikationen/s durchaus mit der Leistung des sehr viel teureren Transistorcomputers 7090 von IBM vergleichbar, wenngleich mit einer deutlich geringeren Wortlänge und einem kleineren Befehlssatz (Tab. 1).²

Der erste von insgesamt fünfzig PDP-1, der serienmäßig mit einem Bildschirm, einem Lightpen und einem Flexowriter ausgestattet war (Abb. 26), wurde zu einem damals revolutionär niedrigen Preis von 120 000 \$ an Bolt, Beranek and Newman, Inc. (BBN) verkauft, ein Bostoner Beratungsunternehmen, das ebenfalls von ehemaligen Mitarbeitern des MIT gegründet worden war und sich unter anderem mit der Gestaltung von Mensch-Maschine-Systemen beschäftigte. Der zweite Computer ging an die Fakultät für Elektrotechnik des MIT, wo er bei den studentischen Hackern bald dem TX-0 den Rang ablief.³

Obwohl das Geschäft zunächst nur schleppend anlief, wurde DEC im Laufe der sechziger Jahre mit einem deutlich verbreiterten Produktsortiment zum Marktführer bei Minicomputern und zu einem wichtigen Produzenten von Großcomputern (PDP-6, PDP-10) für wissenschaftlich-technische Anwendungen.⁴

¹ Der Name sollte ein Signal setzen, daß es sich nicht um einen herkömmlichen Computer handelte. Auch die erwähnte negative wirtschaftliche Einschätzung des Computers spielte eine Rolle.

² Anonymous 1959; Digital Equipment Corporation 1963; Smith 1989, S. 295.

³ Pearson 1992, S. 16ff.; Bertoni and Castleman 1976, S. 454; Levy 1984, S. 52ff.

⁴ Flamm 1988, S. 127ff. – Wichtig für diesen Erfolg war auch, daß DEC sehr viel Wert auf Kundendienst legte, etwa die Modifikation der Hardware, damit der Kunde spezielle Ein- und Ausgabegeräte anschließen konnte.

Zu Beginn der sechziger Jahre hatten sich somit die noch heute im wesentlichen gültigen Standards in der Computerarchitektur und -hardware herausgebildet. Die Zeit der Einzelcomputer und Kleinserienprodukte war endgültig vorbei. Computer waren durch den von der Kybernetik ausgehenden und der militärischen Technikentwicklung geförderten Diskurs vor dem Hintergrund des Kalten Krieges zum industriell gefertigten Massenprodukt geworden.¹

Gleichzeitig teilte sich die Entwicklungslinie, die mit dem *Project Whirlwind* ihren Anfang genommen hatte, in unterschiedliche Richtungen auf, die sich durch die Rolle des Menschen im System unterschieden. In den *Command and Control Systems* wurde die mit dem SAGE-System begonnene Tendenz zur Steuerung realer Vorgänge fortgeführt. Diese Linie reicht vom amerikanischen Raketenfrühwarnsystem bis zur sogenannten strategischen Verteidigungsinitiative SDI während der achtziger Jahre. Der Zweck eines solchen Systems ist nicht durch den Bediener (*operator*) definiert, der vielmehr einen technisch nicht beeinflussbaren und vergleichsweise unsicheren Bestandteil des Gesamtsystems darstellte. Deshalb war die vollständige Automatisierung, d. h. die Entfernung des »Störfaktors Mensch« aus derartigen Computersystemen ein vorrangiges Entwicklungsziel.²

Bei der zweiten Entwicklungslinie, die mit der Entwicklung von Minicomputern verbunden ist, trat der Mensch völlig ins Zentrum des Computersystems, er wurde als Benutzer (*user*) zum einzigen Grund für die Existenz des Computers, an seinen Bedürfnissen mußte sich die Gestaltung aller anderen Systemkomponenten folglich orientieren.

Dabei hatte sie unter dem Namen »Künstliche Intelligenz« (KI) zunächst die Automatisierung intellektueller Tätigkeiten zum Ziel und war damit ein unmittelbarer Nachfolger bestimmter Richtungen der Kybernetik. Der Ausgangspunkt für die im nächsten Kapitel dargestellten Forschungen waren einerseits die ersten Mißerfolge der KI-Forschung, der es nicht gelang, die geweckten hohen Erwartungen einzulösen und andererseits die in diesem Kapitel geschilderten Erfahrungen bei der Computer-Systementwicklung in den fünfziger Jahren.³ Als Gegenreaktion schraubten eine Reihe von Wissenschaftlern ihre kurzfristigen Ziele drastisch zurück, ohne das Fernziel einer Automatisierung geistiger Tätigkeiten aufzugeben. Für sie wurde die Entwicklung von Software-Werkzeugen, die die geistige und kreative Arbeit des Menschen, sei es als Einzelperson oder als Mitglied einer Gruppe, unterstützen sollten, zu einem ersten Schritt auf dem Weg zur künstlichen Intelligenz.

Mit der Auffächerung der Einsatzgebiete des Computers und der gleichzeitigen Etablierung der *Computer Science* bzw. der Informatik als wissenschaftlicher

¹ Pugh and Aspray 1996, S. 17; Edwards 1996, Chapter 3.

² Edwards 1996; Eurich 1991.

³ Winograd und Flores 1992; McCorduck 1987, Teil III.

3. Vom Analogrechner zum Minicomputer

Disziplin, nahmen die Programme – seit Anfang der sechziger Jahre explizit als Software bezeichnet – eine immer größere Rolle für die Leistungsfähigkeit eines Computersystems ein.¹ Deshalb flossen in steigendem Maße (militärische) Fördergelder in die Entwicklung von Software. Damit waren die Voraussetzungen geschaffen, damit die von Vannevar Bush formulierte Vision der informationsverarbeitenden Maschine bei den Programmmentwicklern wieder eine entwicklungsleitende Rolle einnehmen konnte.

¹ Rosen 1967, S. 18; Ceruzzi 1989.

4. Mensch-Computer-Symbiose: Joseph Licklider und die Denkmachine

I guess you could say I had a kind of religious conversion.

J. C. R. Licklider¹

Seit Mitte der fünfziger Jahre entwickelte sich die Frage nach der Organisation nicht nur von wissenschaftlicher, sondern ganz allgemein von geistiger Arbeit und ihrer Unterstützung durch den Computer zu einem wichtigen Forschungsthema. Damit kam man auf Fragestellungen zurück, die bereits Vannevar Bush im Zusammenhang mit dem »Rapid Selector« und dem fiktiven »Memex« beschäftigt hatten.

So nahm z. B. die wirtschaftswissenschaftliche Organisationslehre begeistert kybernetische Verfahren auf, die im Rahmen militärischer Projektplanung und computerunterstützter »Kriegsspiele« entwickelt worden waren.² Auch die 1956 entstandene Disziplin der Künstlichen Intelligenz (KI) setzte sich mit der Automatisierung intellektueller Vorgänge mit Hilfe des Digitalcomputers auseinander. Dabei ging man während der fünfziger und frühen sechziger Jahre von der naiven und daher nicht unumstrittenen Vorstellung aus, man könne mit genügend großem Aufwand das menschliche Denken in Form eines Computerprogramms nachbilden.³

4.1 Der Computer als Partner des Menschen

Die einflußreichsten Ideen zur Neudefinition des Verhältnisses von Mensch und Computer stammten allerdings von Joseph C. R. Licklider (1915–1990), einer der Schlüsselfiguren bei der Entwicklung interaktiver Computer in den sechziger Jahren.

¹ J. C. R. Licklider, zitiert in Rheingold 1985, S. 136

² Johnson et al. 1963; Malcolm 1960.

³ Newell 1983, S. 199ff.; McCorduck 1987; Crevier 1994.

4. Mensch-Computer-Symbiose: Joseph Licklider und die Denkmaschine



Abbildung 27. Joseph C. R. Licklider (1915–1990) um 1960

Licklider hatte bis 1938 an der Washington University in St. Louis Physik, Mathematik und Psychologie studiert. Nachdem er 1942 an der University of Rochester in Psychologie promoviert hatte, arbeitete er am *Psycho-Acoustic Laboratory* der Harvard University, wo während des Zweiten Weltkrieges unter der Leitung von George A. Miller wichtige Grundlagen der Kybernetik, des *Human Factors Engineering* und nicht zuletzt der kognitiven Psychologie erarbeitet wurden.¹

1951 wechselte Licklider an das MIT, um dort die Abteilung für Psychologie aufzubauen, die sich am Forschungsprogramm von Harvard orientierte; gleichzeitig wurde er Mitglied der elektrotechnischen Fakultät des MIT. In dieser Doppelrolle war Licklider auch am *Project Charles* beteiligt und leitete im Rahmen des SAGE-Projekts eine Forschergruppe, die sich mit der Psychologie der Mensch-Maschine-Kommunikation beschäftigte. Zusätzlich beriet er das *Air Force Scientific Advisory Board*, das amerikanische Verteidigungsministerium und eine Reihe von Forschungslaboratorien und Unternehmen, die in der militärischen Forschung und Entwicklung tätig waren.²

Licklider war also ein Wissenschaftler, der die Entwicklung des Digitalcomputers aus nächster Nähe miterlebt hatte und zu den frühen Computerbenutzern gehörte. Der Computer hatte für ihn allerdings zunächst nur instrumentellen Charakter, sei es als Gegenstand seiner Forschungen über die Gestaltung von Mensch-Maschine-Schnittstellen oder als Werkzeug zur Berechnung von komplexen psychoakustischen Modellen. Da Licklider als vielbeschäftigter Berater innerhalb des

¹ Edwards 1996, Chapter 6.

² Norberg 1996, S. 44, Licklider 1988a.

militärisch-industriellen Komplexes tätig war, konnte er bei der Weiterentwicklung des interaktiven Computers eine zentrale Rolle einnehmen, indem er die Kommunikation zwischen den Wissenschaftlern verschiedener Disziplinen und den militärischen Stellen in Gang setzte und aufrechterhielt. Nathan Rosenberg und W. E. Steinmueller haben solche Akteure in der Technikentwicklung als Torwächter (engl. *gatekeeper*) bezeichnet und ihre Bedeutung für die Durchsetzung neuer Konzepte hervorgehoben.¹ Die weitere Entwicklung des *Personal Computing* – soviel sei vorweggenommen – wurde in entscheidender Weise durch Torwächter wie Licklider geprägt.

Obwohl sich Licklider in erster Linie für die Vorgänge interessierte, die beim Hören im menschlichen Gehirn ablaufen, hatte er den Eindruck, daß er einen erheblichen Teil seiner Zeit damit verbrachte, Dinge aus seinen Unterlagen herauszusuchen oder dort abzulegen und eine wachsende Menge von Informationen zu verarbeiten, die er für die Konstruktion seiner Modelle benötigte. Er fragte sich, ob zu diesem Thema bereits wissenschaftliche Untersuchungen existierten, mußte aber feststellen, daß bislang keine »Bewegungsstudien« über geistige Tätigkeiten vorhanden waren. Deshalb entschied er sich – obwohl er sich der Unangemessenheit seines Experiments in methodischer Hinsicht bewußt war – für eine Selbstbeobachtung und protokollierte im Frühjahr 1957 alle Tätigkeiten, die er während seines Arbeitstages als Professor am MIT durchführte.²

Licklider kam dabei bei der Auswertung dieses Experiments zu der für ihn überraschenden Erkenntnis, daß er sogar den weitaus größten Teil seiner Zeit für repetitive Routinetätigkeiten verwendete. Darunter fiel vor allem die Umwandlung von Rohdaten in wissenschaftlich aussagekräftige Informationen und deren Transformation von einer Darstellungsform in eine andere:

»About 85 percent of my ›thinking time‹ was spent getting into a position to think, to make a decision, to learn something I needed to know. Much more time went to the finding or obtaining information than into digesting it. Hours went into the plotting of graphs, and other hours into instructing an assistant how to plot. When the graphs were finished, the relations were obvious at once, but the plotting had to be done in order to make them so. (...) Throughout the period I examined, in short, my ›thinking‹ time was

¹ Rosenberg and Steinmueller 1982, S. 189–191. Rosenberg und Steinmueller zeigen dort am Beispiel der Mikroelektronik, daß technologische Torwächter in der Regel sowohl ein Verständnis für die Möglichkeiten einer neuen Technologie als auch für die Bedürfnisse potentieller Techniknutzer besitzen, ohne selbst Spezialisten zu sein. Sie können deshalb sowohl neue Entwicklungsrichtungen anregen als auch für die praktische Nutzung von Innovationen werben. Durch ihre Vermittlerrolle zwischen Akteuren aus unterschiedlichen Wissenskulturen tragen technologische Torwächter zu einem besseren Transfer von wissenschaftlichen Erkenntnissen in die praktische Verwendung bei.

² Licklider 1960, S. 5f.; Licklider 1988a.

4. Mensch-Computer-Symbiose: Joseph Licklider und die Denkmaschine

devoted mainly to activities that were essentially clerical or mechanical: searching, calculating, plotting, transforming, determining the logical or dynamic consequences of a set of assumptions or hypotheses, preparing the way for a decision or an insight.«¹

Die Schlußfolgerung, die Licklider aus seiner informellen Analyse zog, war einfach und schockierend: Der größte Teil der sogenannten geistigen bzw. wissenschaftlichen Arbeit konnte von Maschinen ausgeführt werden. Da sich die routinemäßigen Anteile der geistigen Tätigkeit in Form einer Verarbeitungsvorschrift, also eines Algorithmus präzise beschreiben lassen, sei es vorteilhafter, sie durch den Computer statt durch den Menschen ausführen zu lassen, zumal der Computer sie schneller und genauer ausführen könne.²

Im Gegensatz zu den jungen KI-Wissenschaftlern wollte Licklider somit nicht gleich den ganzen Wissenschaftler durch ein Computersystem ersetzen, ihm ging es eher um die Schaffung eines Intelligenzverstärkenden Werkzeugs, das die Rolle eines Assistenten übernehmen sollte. Da Licklider zu diesem Zeitpunkt nicht viel von Computertechnik wußte, konnte er unbefangen eine völlige Neuorientierung des Verhältnisses von Mensch und Computer vorschlagen. Wenn man aus einer großen Zahl von Daten eine Kurve oder das grafische Modell der Strömung an einem Flügelprofil ermitteln wollte, war der Stapelbetrieb mit einer »Antwortzeit« von mehreren Stunden völlig ungeeignet. In einem solchen Fall wollte der Wissenschaftler die Parameter des Modells am Terminal verändern und unmittelbar die daraus folgenden Veränderungen der Ergebnisgrößen beobachten können. Der Computer sollte sich also nicht nur für die *Berechnung*, sondern auch für die interaktive *Modellierung* von Problemen verwenden lassen.³

Lickliders Vorstellungen lassen sich damit problemlos in die seit langem am MIT verbreitete Denkweise einordnen. Ähnlich wie die Analogtechnik zu Zeiten Vannevar Bushs sollte der Digitalcomputer zu einem Instrument für die wissenschaftliche Modellbildung und Simulation, langfristig gar ein Werkzeug zur Unterstützung jeglicher geistigen Arbeit werden. In der Tradition der Kybernetik und des SAGE-Systems stand hingegen die Erkenntnis, daß sich die wenigsten Aufgaben vollständig automatisieren lassen und deshalb eine angemessene Aufgabenteilung und effiziente Kommunikation zwischen Mensch und Computer notwendig ist.

Eine solche partnerschaftliche Aufgabenteilung zwischen Mensch und Computer, von Licklider als *Mensch-Computer-Symbiose* bezeichnet, war aber wegen der speziellen Anforderungen nicht mit den existierenden Computertypen zu realisieren, da die Bearbeitung einer Aufgabe im ständigen Dialog ablaufen mußte.

¹ Licklider 1960, S. 6.

² Licklider 1960, S. 6.

³ Licklider 1960, S. 5.

Aus diesem Grund war die Entwicklung von interaktiven Computern für Licklider ein erster wichtiger Schritt zur Realisierung der Mensch-Computer-Symbiose.¹

Noch im gleichen Jahr (1957) verließ Licklider das MIT und wurde Vizepräsident von Bolt, Beranek and Newman, Inc. (BBN). Er leitete dort den von ihm aufgebauten Bereich Mensch-Computer-Interaktion, für den das Unternehmen 1959 den ersten PDP-1 von DEC kaufte.² So konnte Licklider schließlich erste Erfahrungen mit einem grafikfähigen, interaktiven Computer sammeln, die ihn nach eigenen Aussagen endgültig »zum interaktiven Computerbetrieb bekehrt haben«.³

Obwohl Licklider feststellte, daß der PDP-1 die richtige *Art* von Computer war, um seine Modelle zu erstellen, war der Computer mit dieser Aufgabe letztlich hoffnungslos überfordert. Dafür hatte er viel zu wenig Hauptspeicher und nicht genügend Rechenleistung. Obwohl er zu diesem Zeitpunkt nicht wußte, ob und wann der Digitalcomputer leistungsfähig genug sein würde, um einen intelligenten Laborassistenten zu konstruieren, kam Licklider allmählich zu der Überzeugung, daß die Möglichkeiten des Digitalcomputers weit über die bisherigen Einsatzbereiche als reine »Zahlenfresser« hinausgingen.⁴

Licklider faßte seine Erfahrungen aus der Computergestaltung und -nutzung sowie sein Wissen über den Stand der KI-Forschung in seinem 1960 erschienenen Aufsatz »Man-Computer Symbiosis« zusammen. Darin beschrieb Licklider seine Vision eines Computerwerkzeugs für geistige Arbeit und machte Vorschläge, wie ein solches Gerät auf der Basis bestehender und noch zu entwickelnder Technologie zu realisieren sei. So wie bei Vannevar Bush fünfzehn Jahre zuvor hatte sein Text vor allem einen eklektischen und rhetorischen Charakter, der bislang getrennte Ideen zusammenführte, und unter dem Schlagwort »Mensch-Computer-Symbiose« ein großes gemeinsames Ziel formulierte, das schnell zu einem einflußreichen Leitbild in der Computerwissenschaft wurde.⁵

Neben der Formulierung eines Ziels identifizierte Licklider eine Reihe von Problemen, deren Lösung seiner Meinung nach für die Realisierung der »Mensch-Computer-Symbiose« von entscheidender Bedeutung waren und gleichzeitig die Forschungsinteressen der Bostoner Universitäten widerspiegeln. Dies waren unter anderem die Technik und Organisation von Computerspeichern, die Entwicklung von problemorientierten Programmiersprachen, von Ein- und Ausgabegeräten, vor allem aber die Entwicklung sogenannter Time-Sharing-Betriebssysteme.⁶

¹ Licklider 1960, S. 5.

² Norberg 1996, S. 44, Licklider 1988a; Bertoni and Castleman 1976.

³ Licklider in einem Interview mit Howard Rheingold (1983), zitiert in Rheingold 1985, S. 136.

⁴ Licklider 1965; Bobrow et al. 1966.

⁵ Zum Beispiel Taylor 1967; Taylor 1989; Fano 1989; Corbató 1989/90; Engelbart 1988, S. 191f. und Lee 1992a, passim.

⁶ Licklider 1960, S. 7ff.; Norberg et al. 1996, S. 90.

4. Mensch-Computer-Symbiose: Joseph Licklider und die Denkmachine

So gelangte Joseph Licklider, dessen Beschäftigung mit dem Computer ursprünglich in der Formulierung eines Rechenmodells der menschlichen Wahrnehmung bestanden hatte, zur Vision einer Maschine, die ihm helfen konnte, *über* Modelle nachzudenken. Seitdem stellte sich für ihn nicht mehr die Frage, ob der Mensch durch den Computer zu ersetzen war oder nicht, sondern in welcher Weise man Mensch und Maschine »zusammenarbeiten« lassen konnte, um komplexe Aufgabenstellungen effizienter bearbeiten zu können.¹

4.2 Interaktive Computer für das Pentagon

Wie viele andere Visionäre vor und nach ihm wußte auch Licklider, daß eine Vision, mochte sie auch noch so großartig sein, nicht ausreicht, um in der realen Welt etwas zu verändern. Ein Professor für experimentelle Psychologie, selbst am renommierten MIT, war kaum in der Lage, eine ganze Armee von Wissenschaftlern und Ingenieuren für eine interaktive Computerzukunft zu mobilisieren. Erst im Oktober 1962 erhielt Licklider die Möglichkeit zur Realisierung seiner Ideen, als er zum ersten Direktor des neugeschaffenen *Information Processing Techniques Office* (IPTO) und des *Behavioral Science Programs* der *Advanced Research Projects Agency* (ARPA) ernannt wurde.

Die Gründung der ARPA als einer Agentur des amerikanischen Verteidigungsministeriums im Februar 1958 war eine unmittelbare Folge des Sputnik-Schocks. Sie sollte die bislang konkurrierenden Forschungsanstrengungen der einzelnen Truppenteile so koordinieren, daß einerseits keine Doppelentwicklungen gefördert wurden und andererseits keine technologische Lücke gegenüber der Sowjetunion entstand.² Zu den Bereichen, die die ARPA in ihren ersten Jahren förderte, gehörten folglich die Raumfahrt, Raketenabwehr und die Entwicklung von Verfahren zur Detektion von Atomwaffentests.³

Die Förderung von Computerprojekten durch die ARPA wurde erst seit dem Beginn der Präsidentschaft von John F. Kennedy im Frühjahr 1961 verstärkt und

¹ Obwohl die KI-Forschung in den sechziger Jahren erheblich von staatlicher Seite gefördert wurde, blieben die Geldgeber skeptisch, was die von den KI-Forschern geschürten Hoffnungen anging. Robert Taylor bemerkte dazu in einem Interview: »The AI people, who were getting support ... may have thought that the reason that I was supporting AI was because I believed in AI, qua AI. ... I was supporting it because of its influence on the rest of the field, not because I believed that they would indeed be able to make a ping-pong playing machine in the next three years, but because it was an important stimulus to the rest of the field.« Vgl. Taylor 1989, Tape 2, Side 1.

² Vor dem gleichen Hintergrund erfolgte auch die Gründung der *National Aeronautics and Space Agency* (NASA) und die Berufung eines *Director for Defense Research and Development* (DDR&D) innerhalb des Pentagon.

³ Ruina 1989; Norberg et al. 1996, S. 4f.; Lerner 1983, S. 72.

war Bestandteil einer umfassenden Technologiekampagne, die auch den Bereich der Interkontinentalraketen, der bemannten Raumfahrt und der kommerziellen Kerntechnik umfaßte.¹ Diese Entwicklung muß ebenfalls im Zusammenhang mit der Umorientierung gesehen werden, die Kennedy und sein Verteidigungsminister Robert S. McNamara im Bereich der Militärstrategie durchsetzten, indem sie das Konzept der »massiven Vergeltung« durch die Strategie der »flexiblen Antwort« als Militärdoktrin ersetzten.²

Dabei wurde immer deutlicher, daß die bisherigen militärischen Entscheidungsunterstützungssysteme, auch als *Command and Control Systems* bezeichnet, nicht den Anforderungen der neuen Strategie genügten. Im März 1961 rief Präsident Kennedy in einer Rede vor dem Kongreß zur Weiterentwicklung von *Command-and-Control-Systemen* auf, und im Juni 1961 erhielt die ARPA den offiziellen Auftrag zur Einrichtung eines *Command-and-Control-Projektes*.³ Die Einrichtung des *Information Processing Techniques Office* im Jahr 1962 war schließlich das Resultat dieser Vorgänge. Als Direktor des Büros wählten Jack P. Ruina (ARPA-Direktor) und Eugene Fubini (*Director for Defense Research and Development*) nach einigen Überlegungen Joseph Licklider aus: Er war ein etabliertes Mitglied der Bostoner Wissenschaftsgemeinschaft und hatte Erfahrung mit militärischen Forschungsaufträgen und nicht zuletzt mit der Entwicklung von interaktiven Computersystemen für militärische und zivile Anwendungen.⁴

Obwohl die Forschungsrichtung vorgegeben war, prägte Licklider dem Programm des IPTO seinen persönlichen Stempel auf. Für ihn waren die Erfordernisse militärischer *Command and Control Systems* und seine in »Man-Computer Symbiosis« formulierten Ziele weitgehend deckungsgleich. Die für die militärische Entscheidungsunterstützung notwendigen kurzen Antwortzeiten waren nach seiner Überzeugung nur durch die Entwicklung flexibler interaktiver Computersysteme zu erreichen, die Verwendung von Computern im etablierten Stapelbetrieb hielt er für schlicht lächerlich.⁵ Als technische Lösung favorisierte Licklider dabei sogenannte Time-Sharing-Systeme (dt. auch Teilnehmersysteme), bei denen mehrere Benutzer gleichzeitig so mit dem Computer interagieren können, als hätten sie ihn zu ihrer alleinigen Verfügung.⁶

Das Forschungsprogramm des IPTO hatte damit einen janusköpfigen Charakter. Für die Öffentlichkeit, insbesondere für den amerikanischen Kongreß wurde gern auf die Bedeutung der Verfahren zur Mensch-Computer-Interaktion für die konkrete Verwendung in militärischen Systemen hingewiesen, während man

¹ Kaiser 1992, S. 300, 457; MacKenzie 1990, S. 112f.

² Angermann 1995, S. 382.

³ Norberg et al. 1996, S. 10.

⁴ Licklider 1988a; Ruina 1989.

⁵ Licklider 1988a; Morris 1977, S. 177ff.

⁶ Corbató 1983, S. 1521.

4. Mensch-Computer-Symbiose: Joseph Licklider und die Denkmachine

Tabelle 2. Direktoren des Information Processing Techniques Office, 1962–1979

	Amtszeit	Ausbildung
J. C. R. Licklider	10/62–07/64	Ph. D., Psychologie, Univ. of Rochester, 1942
Ivan E. Sutherland	07/64–06/66	Ph. D., Elektrotechnik, MIT, 1963
Robert W. Taylor	06/66–03/69	M. A., Psychologie, Univ. of Texas, 1964
Lawrence G. Roberts	03/69–09/73	Ph. D., Elektrotechnik, MIT, 1963
J. C. R. Licklider	01/74–08/75	Ph. D., Psychologie, Univ. of Rochester, 1942
David C. Russell	09/75–08/79	Ph. D., Physik, Tulane Univ., 1968

den Computerwissenschaftlern die Freiheit gab, innerhalb des vorgegebenen Rahmens ihre eigenen Ideen ohne äußere Einflußnahme zu entwickeln.¹ Aus diesem Grund betrieb die Mehrzahl der in den Jahren bis 1970 vom IPTO geförderten Institutionen Grundlagenforschung. Es stellte sich bei den Wissenschaftlern sogar das Gefühl ein, die von der ARPA geförderte Computerforschung sei völlig zweckfrei gewesen und habe keinen Bezug zur militärischen Verwendung gehabt.² Tatsächlich dürfte, ähnlich wie im Fall des *Project Whirlwind*, eine gegenseitige Beeinflussung von Grundlagenwissenschaft und militärischer Verwendung stattgefunden haben, bei der die beteiligten Wissenschaftler durchaus das Gefühl völliger Autonomie gehabt haben können, zumal Licklider und seine Nachfolger (Tab. 2) nur solche Projekte zu fördern versuchten, die mehr als einen marginalen Fortschritt versprachen.³

Als Folge dieser Politik verstand sich das IPTO vor allem als auswählende und koordinierende Instanz. Die Auswahl von Projekten erfolgte nicht wie bei der zivilen Forschungsförderung (etwa der *National Science Foundation*) in der langwierigen Form von unabhängigen externen Gutachten (*peer reviews*). Über die gestellten Anträge wurde vielmehr relativ schnell direkt vom IPTO-Direktor und seinem Stellvertreter entschieden. Hatte ein Antragsteller diese erst einmal vom innovativen Potential einer Idee überzeugt und einen Vertrag erhalten, dann hatte er in der Regel bei der Durchführung seines Projektes für drei Jahre freie Hand, zumal die Fördersummen auch dreißig bis vierzig mal größer als etwa bei der *National Science Foundation* waren.⁴

Die Koordination der Projekte fand direkt durch den IPTO-Direktor bzw. seinen Stellvertreter sowie durch regelmäßige Treffen der Projektleiter (*Principal Investigators*) statt, wo im kleinen Kreis über den Stand der Projekte berichtet und diskutiert wurde (Abb. 28). Die Teilnehmer dieser Treffen weisen rückblickend regelmäßig darauf hin, daß der informelle Gedankenaustausch zwischen

¹ Ruina 1989; Norberg et al. 1996, S. 12f.

² Marvin L. Minsky, zitiert in Brand 1990, S. 200

³ Taylor 1989.

⁴ Norberg et al. 1996, S. 13; Hughes 1998, S. 256f.



Abbildung 28. Konferenz der ARPA Principal Investigators im Dezember 1962 in Santa Monica. Am Kopfende des Tisches J. C. R. Licklider (links) und Herbert Benington (rechts), weitere Anwesende: Marvin L. Minsky (4. v. l.), Fernando J. Corbató (am Tisch, vorne rechts), Alan Perlis (rechts vorne) und Jules I. Schwartz (rechts hinten)

den Wissenschaftlern und die Möglichkeit zur offenen und sachlichen Kritik in einem kooperativen Klima nicht nur dazu beigetragen haben, Probleme zu lösen und Irrwege frühzeitig zu erkennen, sondern daß die Treffen auch eine gemeinschaftsstiftende Wirkung hatten und eine regelmäßigen Refokussierung auf das gemeinsame Ziel ermöglichten.¹

Berücksichtigt man zusätzlich, daß die ARPA eine sehr flache Hierarchie besaß und im Gegensatz zu vielen anderen staatlichen Bürokratien viel Wert auf die Förderung von wenigen *Centers of Excellence* legte, ist es nicht unangebracht zu behaupten, daß die ARPA eine Art Neuauflage des im Zweiten Weltkrieg so erfolgreichen O.S.R.D. für den Kalten Krieg war.²

4.3 Time-Sharing

Die in »Man-Computer Symbiosis« artikulierten Ziele wurden 1962 ohne größere Änderungen in das Programm des IPTO unter Lickliders Leitung übernommen:

¹ Norberg et al. 1996, S. 44f.; Feigenbaum 1989; Taylor 1989.

² Edwards 1996, S. 261 – Für eine ausführlichere Geschichte der *Advanced Research Projects Agency* siehe Richard J. Barber Associates 1975; Reed et al. 1990/91 und Norberg et al. 1996. Zur Problematik der wissenschaftlichen Autonomie bei der militärischen Forschungsförderung vergleiche Forman 1987.

4. Mensch-Computer-Symbiose: Joseph Licklider und die Denkmachine

Time-Sharing, Computergrafik und Künstliche Intelligenz erhielten zunächst den Löwenanteil der vom IPTO vergebenen Fördermittel.

Obwohl in den vorangegangenen Jahren bereits große Fortschritte im Bereich der Mensch-Computer-Kommunikation gemacht worden waren, erschienen Licklider alle existierenden interaktiven Computersysteme für die Realisierung der Mensch-Computer-Symbiose ungeeignet.

Die großen militärischen Computersysteme wie SAGE waren zwar nicht durch das Gebot einer möglichst ökonomischen Nutzung der Hardware belastet, aber sie waren ursprünglich computer- und nicht benutzerorientiert konzipiert worden. Für die speziellen Aufgaben dieser Systeme waren einfache Interaktionstechniken und Ein- und Ausgabegeräte wie Charactron-Bildröhre, Lichtgriffel und Steuerknöpfe vollkommen ausreichend. Dem Bediener eines solchen Computers waren bei der Entscheidungsfindung wie auch bei der Mensch-Computer-Interaktion enge Grenzen gesetzt, da alle möglichen Aktionen bereits vor Beginn der Programmentwicklung festgelegt worden waren. Es fehlte somit das kreative Moment, die Möglichkeit zur kontinuierlichen Weiterentwicklung des Systems (d. h. des Computerprogramms) durch seine Benutzer.¹

Offenbar existierte unter den Computerverwendern zu Beginn der sechziger Jahre bereits eine wachsende Nachfrage nach interaktiven Computersystemen. Darunter fielen neben den bereits erwähnten Programmierern und Militärstrategen des Pentagon auch kommandierende Offiziere, die sich bei den automatisierten Systemen in der Tradition von SAGE weitgehend ihrer Entscheidungsbefugnis beraubt sahen. Sie hofften, daß ihnen die neue Generation der *Command and Control Systems* diese wichtige Kompetenz zurückgeben würde. Schließlich machte Licklider deutlich, daß sich die Verwendungsmöglichkeiten von Time-Sharing-Computern nicht nur auf die militärische Entscheidungsunterstützung und die interaktive Programmerstellung beschränkte. Er nannte auch Einsatzmöglichkeiten bei der Planung und beim Entwurf technischer Systeme², in der computerunterstützten Ausbildung und bei sogenannten halbautomatischen Laboratorien, in denen der Computer die Modellbildung aus einer Vielzahl von zur Verfügung stehenden Daten ermöglichen sollte.³

Licklider griff bei seinem Förderplan auf Ideen zurück, die Ende der fünfziger Jahre unter Computerwissenschaftlern lebhaft diskutiert wurden. Nachdem Herbert Benington den Begriff »Time-Sharing« schon 1956 in einem Aufsatz über die Programmierung des SAGE-Systems verwendet hatte⁴, machte der Brite Christo-

¹ Licklider and Clark 1962, S. 113; Norberg et al. 1996, S. 73.

² In diesen Bereich fallen alle heute als Computer Aided Design (CAD), Computer Aided Engineering (CAE) usw. bezeichneten Computeranwendungen.

³ Licklider and Clark 1962, S. 113f.; Edwards 1996, S. 106.

⁴ Benington 1983, S. 355.

pher Strachey (1916–1975) 1959 erstmals den Vorschlag, Systemprogramme zu entwickeln, mit deren Hilfe mehrere Programme gleichzeitig auf einem Computer ablaufen konnten.¹ Noch einflußreicher war das Konzept, das John McCarthy (* 1927) 1959/60 am MIT entwickelte und auch Licklider im Detail bekannt war.² Da die Verarbeitungsgeschwindigkeit des Computers um ein Vielfaches größer war als die Fähigkeit des Menschen zur Dateneingabe, schlug McCarthy vor, die in Millisekundenabschnitte unterteilte Rechenzeit des Computers reihum auf die Programme einer Vielzahl von Nutzern aufzuteilen und so die Auslastung der Maschine zu erhöhen. Jeder Nutzer erhielt auf diese Weise die Möglichkeit, im Dialog mit dem Computer zu arbeiten (d. h. unmittelbare Darstellung der eingegebenen Daten auf dem Bildschirm und eine Antwortzeit bei der Programmausführung im Sekundenbereich) und hatte das Gefühl, er habe den ganzen, wenn auch weniger leistungsfähigen Computer zu seiner alleinigen Verfügung.³

Dabei sollte keine Unterscheidung mehr zwischen Programmierer (off-line) und Bediener (on-line) gemacht werden, wie sie noch beim SAGE-System üblich war. Vielmehr orientierte man sich an der ersten Generation der Hacker, die es schätzen gelernt hatten, Programme direkt an der Bedienerkonsole des Computers zu schreiben, zu überprüfen und gegebenenfalls zu verbessern. (vgl. S. 113).⁴

Noch bevor Licklider als Direktor des IPTO die Entwicklung von Time-Sharing-Systemen massiv propagierte und finanziell förderte, waren erste experimentelle Time-Sharing-Systeme entstanden. Bei Bolt, Beranek and Newman, Inc. entwickelte Edward Fredkin 1961 auf Anregung Lickliders ein erstes Time-Sharing-System für den DEC PDP-1, das sich aber für eine sinnvolle Verwendung als zu leistungsschwach herausstellte.⁵ Am MIT wurde 1961 unter der Leitung von Fernando J. Corbató (* 1926) das *Compatible Time-Sharing System* (CTSS) fertiggestellt, das auf einem IBM-7090-Computer lief und eine gleichzeitige Benutzung der Hardware im Stapel- und im Dialogbetrieb mit drei Terminals ermöglichte.⁶ Ein weiteres System für den DEC PDP-1 wurde von Jack Dennis, einem ehemaligen Mitarbeiter von John McCarthy ebenfalls am MIT entworfen und 1964 als PDP-1 TSS in Betrieb genommen.⁷

¹ Strachey 1960, S. 338 – Diese Form des Time-Sharings wird heute gewöhnlich »multiprogramming« genannt. Bis weit in die sechziger Jahre wurden unter Fachleuten haarspalterische Diskussionen über die »richtige« Definition von Time-Sharing geführt. Vgl. Rosenberg 1966, S. 66.

² Licklider 1988a.

³ McCarthy 1959; McCarthy 1983 – Tatsächlich sind die Forderungen nach möglichst hoher Auslastung des Computers und nach einer möglichst kurzen Antwortzeit antagonistische Ziele.

⁴ Licklider and Clark 1962, S. 121 – Eine ausführlichere Geschichte der Time-Sharing-Entwicklung während der sechziger und frühen siebziger Jahre findet sich in Hellige 1996a und Norberg et al. 1996, Chapter 2.

⁵ McCarthy et al. 1963; Licklider 1988a.

⁶ Corbató et al. 1962.

⁷ Lee 1992b; McCarthy 1983.

4. Mensch-Computer-Symbiose: Joseph Licklider und die Denkmachine

Ebenfalls noch vor Lickliders offiziellem Amtsantritt war auf Veranlassung der ARPA ein *Command-and-Control-Laboratorium* bei der *System Development Corporation* (SDC) in Santa Monica, Ca. eingerichtet worden, wo man bereits seit den fünfziger Jahren Erfahrungen mit dem Entwurf und der Realisierung von interaktiven Computersystemen besaß. Dieses Labor führte 1961 eine Vorabstudie über die Erfordernisse von computergestützten *Command-and-Control-Systemen* durch. Zu diesem Zweck erhielt SDC im Juli 1961 einen AN/FSQ-32-Computer, der ursprünglich zum Kernstück eines *Super Combat Center* hätte werden sollen, dessen Entwicklung aber im April 1960 eingestellt worden war.¹

Während des Jahres 1962 veränderte sich die Zielsetzung der bei SDC durchgeführten Forschung. Der designierte IPTO-Direktor Licklider bestand, trotz eines anderslautenden Vertrages, auf der Entwicklung eines experimentellen Time-Sharing-Systems, das dem damaligen Stand der Technik entsprechen sollte. Der AN/FSQ-32-Computer war für die Entwicklung eines solchen Systems besonders gut geeignet, handelte es sich doch um den von IBM hergestellten transistorisierter Nachfolger des SAGE-Computers AN/FSQ-7, der bereits für den Betrieb einer Vielzahl von interaktiven Terminals entworfen worden war. Unter der Leitung des erfahrenen Programmentwicklers Jules I. Schwartz und unter Mitwirkung von Edward Fredkin wurde bis Juni 1963 eine erste Version des Q-32-Time-Sharing-Systems (TSS) fertiggestellt und bei der Frühjahrstagung der *American Federation of Information Processing Societies* (AFIPS) dem Fachpublikum vorgestellt.² Obwohl TSS auf einem der leistungsfähigsten Computer seiner Zeit lief und bei der Programmierung den Stand der Technik repräsentierte, konnte es anfangs lediglich acht Terminals gleichzeitig bedienen und stellte den Benutzern keinen Plattenspeicher zur Verfügung. Dieses Schwächen wurden innerhalb kurzer Zeit behoben, und im Frühjahr 1964 konnte Jules Schwartz mit einigem Stolz feststellen, daß mittlerweile 48 Terminals an den Q-32 angeschlossen seien und daß man den Benutzern eine Reihe von stabil laufenden Anwendungsprogrammen zur Verfügung stellen könne, darunter die Programmiersprachen JOVIAL und IPL.³

Zur Unterstützung der Forschungsaktivitäten bei SDC förderte Licklider seit Ende 1963 eine größere Zahl von weiteren Projekten an der amerikanischen Westküste. Er vergab Aufträge an die *University of California* in Los Angeles, an

¹ Baum 1981, S. 89f. – Der Q-32-Computer hatte 1963 eine Wortbreite von 48 bit, einen Hauptspeicher von 65 536 Worten mit einer Zugriffszeit von 2,5 μ s, einen zusätzlichen Speicher von 16 384 Worten, der als Puffer für die Ein- und Ausgabe verwendet wurde, einen Trommelspeicher mit einer Kapazität von 500 000 Worten und einen umfangreichen Befehlssatz. Vgl. Schwartz et al. 1964, S. 398.

² Baum 1981, S. 89f.

³ Schwartz et al. 1964. JOVIAL (Jules' Own Version of the International Arithmetic Language) war ein von Jules Schwartz entwickelter ALGOL-Dialekt, IPL (Information Processing Language) eine listenorientierte Programmiersprache von A. Newell, J. C. Shaw und H. A. Simon.

die *University of California* in Berkeley, an die *Stanford University* und an das *Stanford Research Institute* (SRI) in Menlo Park. Dabei legten die Verträge eine Verwendung des Q-32-Time-Sharing-Systems fest, das allein auf dem in Santa Monica installierten AN/FSQ-32-Prototypen lief, mit dem die Institutionen per Datenfernübertragung verbunden waren. Auf diese Weise, so hoffte Licklider, würde aus einer zunächst inhomogenen Gruppe von Wissenschaftlern eine Gemeinschaft mit ähnlichen Interessen und Zielen entstehen, die Licklider gelegentlich als *California Network Group* bezeichnete. Auf diese Weise wurde das ursprünglich auf die Bostoner Wissenschaftsgemeinschaft beschränkte Engagement für Time-Sharing und interaktiven Computerbetrieb auch nach Kalifornien übertragen.¹

Im Zentrum der Förderung standen allerdings nicht die Projekte in Kalifornien, sondern das *Project MAC*, das unter der Leitung von Robert M. Fano (* 1917) am MIT durchgeführt wurde. MAC stand dabei sowohl für *Multiple Access Computer* als auch für *Machine Aided Cognition*. Das ehrgeizige Projekt hatte also nicht nur die Entwicklung eines Time-Sharing-Betriebssystems zum Gegenstand, sondern war gleichzeitig Bestandteil der am MIT intensiv betriebenen KI-Forschung, die mit Persönlichkeiten wie Marvin L. Minsky, Seymour Papert und Joseph Weizenbaum verbunden ist.²

Bereits ab 1964 wurde in Zusammenarbeit mit den Bell Laboratorien und der *General Electric* versucht, die Ergebnisse des *Project MAC* in ein kommerziell verwertbares Produkt, das Betriebssystem Multics, zu überführen, dessen Fertigstellung ursprünglich für Ende 1965 angekündigt war. Wegen andauernder technischer Probleme, die ihre Ursache in der Komplexität der geplanten Software hatten, und wegen des Ausstiegs der Bell Labs aus dem gemeinsamen Projekt³ wurde erst im Oktober 1969 ein erster Prototyp für den internen Gebrauch am MIT fertiggestellt, die kommerzielle Verwertung begann sogar erst in den siebziger Jahren.⁴

¹ Norberg et al. 1996, S. 92–94.

² Zur Geschichte der KI siehe Newell 1983; McCorduck 1987; Kurzweil 1993 und Crevier 1994.

³ Die Entwicklung des erfolgreichen Betriebssystems UNIX für den DEC PDP-11 durch Ken Thompson und Dennis Ritchie (1969–1971) war eine unmittelbare Reaktion der Bell Labs auf den Ausstieg aus der Multics-Entwicklung. Vgl. dazu Salus 1994.

⁴ Fano and Corbató 1966; Fano 1979; Lee 1992a; Lee 1992b; Norberg et al. 1996, S. 94–102, 109–112. Obwohl Multics bis in die neunziger Jahre verwendet wurde, hatte es nie mehr als etwa 60 (institutionelle) Nutzer.

Neben den beiden großen Time-Sharing-Systemen bei SDC und am MIT förderte das IPTO insbesondere in der zweiten Hälfte der sechziger Jahre eine Reihe von kleineren Projekten. Dazu gehörte die Entwicklung des TENEX-Betriebssystems für den Großrechner PDP-10 von Digital Equipment durch BBN und das *Project Genie* an der Universität Berkeley, das die Entwicklung eines Time-Sharing-Betriebssystems für den wissenschaftlichen Computer SDS 930 von *Scientific Data Systems* beinhaltete. Beide Projekte hatten nur ein begrenztes Ziel und wurden bald in kom-

4. Mensch-Computer-Symbiose: Joseph Licklider und die Denkmaschine

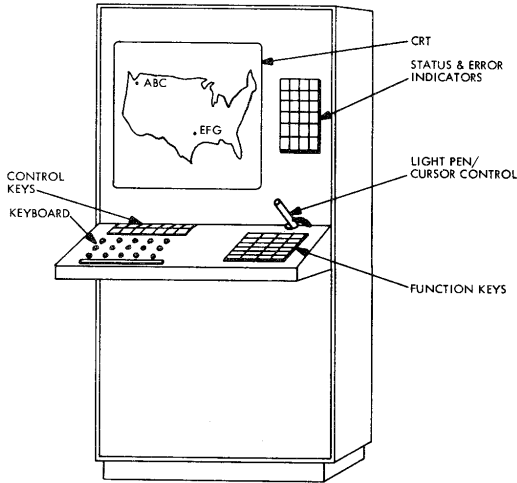


Abbildung 29. Computerterminal eines Time-Sharing-Computers mit Bildschirm, Lichtgriffel und Tastatur, um 1965

Time-Sharing-Betriebssysteme darf man allerdings nicht mit der von Licklider formulierten Symbiose gleichsetzen. Sie waren allein eine Folge der weiterhin hohen Computerpreise, durch die es immer noch unmöglich war, jedem Benutzer seinen eigenen Computer zur Verfügung zu stellen, wie es Licklider eigentlich für notwendig erachtete.¹

Wie sah nun die Arbeit mit einem Computer im Time-Sharing-Betrieb konkret aus und inwieweit konnten sie die von Licklider formulierten hohen Ansprüche erfüllen? Diese Fragen können schlaglichtartig durch zwei Beispiele beantwortet werden.

Abbildung 30 zeigt die Verwendung des SDC-Time-Sharing-Systems als »Desktop Calculator«. Der Benutzer saß hierbei an einem Terminal, das mit dem Q-32-Computer verbunden war. Solche Terminals waren zu Beginn der sechziger Jahre in der Regel mit einem Fernschreiber (Flexowriter) seltener mit einer Tastatur und einem Bildschirm ausgestattet (Abb. 29).

Im dargestellten Beispiel meldet sich der Benutzer mit dem LOGIN Kommando beim Computer an und lädt dann TINT, einen interaktiven JOVIAL-Interpreter, der sich auf dem Plattenspeicher des Computers befindet und jedem Benutzer zur Verfügung steht. Das Kommando GO startet einen Bildschirmeditor, mit dem

merzielle Produkte, das TOPS-20 Betriebssystem von DEC bzw. den SDS-940-Computer, überführt, die unter Computerwissenschaftlern sehr beliebt waren. Vgl. Norberg et al. 1996, S. 102ff., 116; Lampson et al. 1966; Bobrow et al. 1972.

¹ Licklider 1960, S. 7; Friedewald 1998.

Bildschirmdialog

```

LOGIN 0173 JDJ.25 ◀-----
$OK LOG ON 14
LOAD TINT 1796 ◀-----
$WAIT

$LOAD OK
GO
$MSG IN ◀-----
ITEM N F $ "NUMBER OF CASES"
ITEM SUMX F $ "SUM OF VALUES"
ITEM XBAR F $ "ARITH. MEAN"
ITEM SDEV F $ "STD. DEVIATION"
READ N, SUMX $
XBAR = SUMX/N $
PRINT 6H(MEAN =), XBAR $
SDEV = ((SUMX**2.0-N*XBAR**2.0)/(N-1))**0.5 $
PRINT 8H(S. D. =), SDEV $
TERM $ "CAUSES EXECUTION OF PROGRAM" ◀-----

      N = ? 12.0 ◀-----
      SUMX = ? 1478.0

      MEAN =      132.2 ◀-----
      S. D. =      10.3

      ILLT EXECUTION COMPLET ◀-----
!QUIT
$MSG IN

```

Erläuterung

Der Benutzer loggt sich in den Time-Sharing Computer ein. Er startet TINT, einen JOVIAL-Interpreter.

Ein Programm für statistische Berechnungen wird am Terminal eingegeben.

Ende der Programmeingabe und Beginn der Programmausführung
Interaktive Abfrage der zu verarbeitenden Daten

Unmittelbare Ausgabe der Ergebnisse.

Ende der Programmausführung

Abbildung 30. Beispiel einer Computersitzung am SDC-Time-Sharing-System. Dabei wird mit Hilfe des JOVIAL-Interpreters ein kurzes Statistikprogramm realisiert.

der Nutzer sein Programm, hier zur Berechnung von Mittelwert und Standardabweichung, eingeben kann. Unmittelbar danach startet er das Programm und führt seine Berechnungen durch. Dabei müssen die Daten für die Rechnung nun nicht mehr vollständig vor dem Programmstart vorliegen, sondern werden über die READ-Anweisung während des Programmlaufs abgefragt. Die Ergebnisse der Berechnung werden unmittelbar nach Ende der Datenabfrage auf dem Bildschirm angezeigt.¹

Die von John McCarthy und anderen formulierte Forderung nach dem unmittelbaren Zugriff zum Computer wurden bereits mit diesem frühen Time-Sharing-System prinzipiell erfüllt. Damit änderte sich auch bald das Berufsbild des Programmierers. So stellte das Branchenmagazin *Datamation* bereits im Februar 1966 fest, daß sich mit der Einführung von Time-Sharing-Systemen und höheren Programmiersprachen (wie JOVIAL, FORTRAN oder COBOL) der Schwerpunkt bei der Ausbildung von Computernutzern mehr auf die *Beherrschung des Computers* konzentrierte und weniger auf die Fähigkeit zur *Codierung* von Programmen.²

¹ Schwartz et al. 1964, S. 406; Shaw 1963, Schwartz and Weissman 1967.

² Rosenberg 1966, S. 74.

4. Mensch-Computer-Symbiose: Joseph Licklider und die Denkmachine

Bildschirmdialog	Erläuterung
*KILL. ? <-----	Der Benutzer versucht, eine "Löschen"-
*1, \$KILL. ?	Aktion zu starten und macht dabei einen Fehler.
*HELP. <-----	Er startet die Online-Hilfe
?HOW DO I CLEAR A BUFFER? <-----	und formuliert seine Frage.
DO YOU MEAN A STRING BUFFER OR THE MAIN TEXT BUFFER? <--	Auf Nachfrage des Systems
?MAIN TEXT BUFFER. <-----	präzisiert er seine Frage.
TO DELETE ONE OR MANY LINES IN THE MAIN TEXT BUFFER, <--	Das System liefert die gewünschte Information
USE THE "DELETE" COMMAND. FOR EXAMPLE:	
"1, \$DELETE." CLEARS THE ENTIRE BUFFER WHILE <-----	und gibt Beispiele.
"3DELETE." REMOVES LINE 3.	
?	
*1, \$DELETE. <-----	Der Benutzer gibt das richtige Kommando ein.
*	

Abbildung 31. Online-Hilfe des Berkeley Time-Sharing Systems

In diesem Zusammenhang gab es auch erste Bestrebungen, Computersysteme benutzerfreundlicher¹ zu gestalten. Dazu wurden erstmals sogenannte Online-Hilfen implementiert, bei denen der Benutzer im Dialog mit dem Computer Informationen über die Syntax und die Funktion von Systemkommando bzw. Programmanweisungen abrufen konnte. Abbildung 31 zeigt einen solchen Dialog, wie er im *Berkeley Time-Sharing System*² implementiert war. Dabei startet der Benutzer mit dem Kommando `HELP` die Online-Hilfe, nachdem er zuvor vergeblich versucht hatte, einen Pufferspeicher zu löschen. Er stellt seine Frage nach dem gesuchten Kommando in Form eines normalen (englischen) Satzes. Das Hilfe-Programm analysiert die Struktur der eingegebenen Frage, sucht nach Schlüsselwörtern und fragt die zur Beantwortung der Frage noch fehlende Informationen ab. Schließlich liefert die Online-Hilfe die gewünschte Information und gibt zum besseren Verständnis zwei Beispiele für die Verwendung des fraglichen Kommandos.³ So sehr Online-Hilfen dazu beitrugen, die Benutzerfreundlichkeit des Systems zu erhöhen, so stellten sie doch einen ersten Schritt zur »Verschwendung« von Rechenzeit dar, die während der siebziger Jahre zu einem Charakteristikum bei der Entwicklung von interaktiven Kleincomputern wurde.

¹ Der Begriff der Benutzerfreundlichkeit eines Computersystems wurde allerdings erst zu Beginn der siebziger Jahre geprägt. Was Benutzerfreundlichkeit konkret sein sollte, war seither allerdings stets umstritten. Mittlerweile fordert man, Computersysteme sollten nützlich und nutzbar sein. Vgl. dazu Nake 1994, S. 315f. und Shneiderman 1992. Dennoch wird im folgenden weiterhin von Benutzerfreundlichkeit gesprochen.

² Das *Berkeley Time-Sharing System* war das Ergebnis des vom IPTO finanzierten »Project Genie«, das von Edward Feigenbaum und David C. Evans geleitet wurde. Es wurde im April 1965 fertiggestellt und lief auf einem SDS-930-Computer von Scientific Data Systems. Vgl. Lichtenberger and Pirtle 1965.

³ Rosenberg 1966, S. 66.

Mit der Einführung von Time-Sharing-Systemen änderte sich aber nicht nur die Form und Qualität der Interaktion zwischen Mensch und Computer. Es ist auch zu beobachten, daß sich auch die Grundhaltung des Benutzers gegenüber dem Computer veränderte. Eine Reihe von Benutzern genossen es, Macht über den Computer auszuüben, indem sie »der Maschine auf dem Weg der Umsetzung eines vorher festgelegten Plans den eigenen Willen aufzwingen«.¹ Auf der anderen Seite gab es viele, vor allem junge Nutzer, die einen eher spielerischen Zugang zum Computer entwickelten. Ihnen machte es Freude, die Möglichkeiten und Grenzen des Computers im Dialog mit der Maschine zu entdecken und dabei unter Umständen sogar völlig neue Nutzungsmöglichkeiten zu entwickeln. Diese Tendenz manifestierte sich seit Beginn der sechziger Jahre durch eine wachsende Anzahl von Computerspielen, unter denen das von Steve Russell am MIT entwickelte »Spacewar« das wohl bekannteste ist. Während solche Spiele unter dem Gesichtspunkt einer möglichst ökonomischen Computernutzung vollkommen unsinnig waren, wurden sie gern zur Demonstration des Leistungsumfangs von Time-Sharing-Systemen verwendet und entwickelten sich im Laufe der siebziger Jahre zu einem bevorzugten Betätigungsfeld der zweiten Generation von Hackern, die bei der Entwicklung von sogenannten Homecomputern eine wichtige Rolle spielten (vgl. Kapitel 7).²

Die massive Förderung des Time-Sharings und die gezielte Verteilung der Fördermittel über eine Reihe von Institutionen in den ganzen USA zeigte schon bald Wirkung. Spätestens seit Mitte der sechziger Jahre nahm auch die Computerindustrie Notiz von der neuen Technik, die damit langsam an Eigendynamik gewann. Selbst IBM, ansonsten eher vorsichtig bei der Übernahme von Neuentwicklungen, entschloß sich schon 1966 zur Entwicklung eines eigenen Time-Sharing-Betriebssystems für sein System/360.³ Auf diese Weise wurde Time-Sharing zwischen Mitte der sechziger und Anfang der siebziger Jahre zu einer akzeptierten Technik des Computerbetriebs, die sich als technologische und kulturelle Wasserscheide in der Computergeschichte herausstellen sollte. Wie Licklider es vorhergesagt hatte, begann sich mit dem Time-Sharing nicht nur die Art und Weise zu verändern, wie Daten verarbeitet wurden, sondern auch die Denkweise der Menschen. Viele der Wissenschaftler, die diesen Umbruch aktiv mitgestaltet hatten, waren während der siebziger Jahre auch bei der Schaffung von wirklich persönlichen Computern beteiligt.

Mit der Durchsetzung von Time-Sharing-Betriebssystemen war allerdings erst ein erster Schritt zu Lickliders »Mensch-Computer-Symbiose« vollzogen. Vorerst war es lediglich die Arbeit der Programmierer, die durch den Computer in geeig-

¹ Turkle 1984, S. 128.

² Rosenberg 1966, S. 74; Brand 1974, S. 39ff.; Graetz 1981; Levy 1984, S. 59–66, passim.

³ O'Neill 1995; Pugh et al. 1991, S. 362ff.

4. Mensch-Computer-Symbiose: Joseph Licklider und die Denkmaschine

netter Weise unterstützt werden konnte. Es fehlte aber noch vollkommen an Werkzeugen, d. h. Computerprogrammen, die eine einfache Manipulation von symbolischer und bildlicher Information erlaubt hätten, die das Kennzeichen geistiger Arbeit darstellt. In einer Prioritätenliste für die vom IPTO zu fördernden Bereiche von Wissenschaft und Technik befand sich deshalb 1962 auch die Gestaltung von Werkzeugen zur besseren Kommunikation zwischen Mensch und Computer:

»[We have to devise] an electronic input-output surface on which both the operator and the computer can display, and through which they can communicate, correlated symbolic and pictorial information. The surface should have selective persistence plus selective erasability; the computer should not have to spend a large part of its time maintaining the displays. The entire device should be inexpensive enough for incorporation into a remote console.«¹

Abbildung 32 vermittelt einen Eindruck, wie ein Computerarbeitsplatz, der diese Anforderungen erfüllt, hätte aussehen können.

Die im Rahmen der Time-Sharing-Projekte entwickelten Systeme waren allerdings noch weit davon entfernt, eine natürliche und intuitiv beherrschbare Kommunikation zwischen Mensch und Computer zu ermöglichen. Dies lag in nicht unerheblichem Maße an der von den Programmentwicklern anvisierten Nutzergruppe. Es wurde bereits erläutert, daß es die Programmierer selbst waren, die eine Veränderung des Computerbetriebs für wünschenswert gehalten hatten. Diese Gruppe war es auch, die in verantwortlicher Position im IPTO die von ihr favorisierte Technik propagierte und finanziell unterstützte, und sie waren es auch, die während der sechziger Jahre Time-Sharing-Systemen gestalteten.

Time-Sharing war also eine Technik von Programmierern für Programmierer, die entstehenden Computersysteme wurden schließlich intensiv als Werkzeug zur Gestaltung von verbesserten Nachfolgesystemen verwendet. Diese in der Informatik auch als *Bootstrapping* bezeichnete Entwurfsmethodik, die seit Beginn der industriellen Revolution eine lange und erfolgreiche Tradition in der Technikentwicklung hat, wurde bei dem von Douglas C. Engelbart am Stanford Research Institute geleiteten *Augmentation Research Center* zur theoretischen Basis eines fortschrittlichen Computersystems zur Unterstützung auch von anderen intellektuellen Tätigkeiten, insbesondere im Team. Die Bemühungen Engelbarts und seiner Mitarbeiter zur Gestaltung eines diesen Aufgaben angemessenen Werkzeugs ist Gegenstand des nächsten Kapitels.

¹ Licklider and Clark 1962, S. 121; vgl. auch Sutherland 1965.

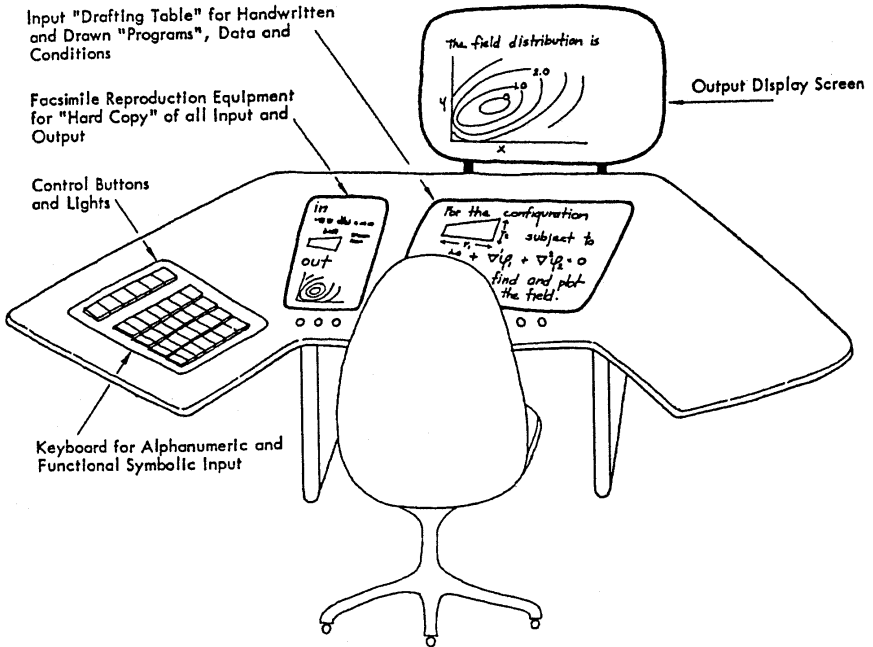


Abbildung 32. Herbert Teagers ambitionierter Entwurf eines Computerterminals von 1961. Teager leitete zu Beginn der sechziger Jahre eine der Time-Sharing-Forschungsgruppen am MIT, die aber bald zugunsten des pragmatischeren Project MAC aufgelöst wurde.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

... all we need to do is to put the proper elements together – at a reasonable expense – and we have a memex.

Vannevar Bush 1967

Douglas C. Engelbart las 1945 Vannevar Bushs programmatischen Artikel »As we may think«. 1951 entschloß er sich, Bushs Vision in die Realität umzusetzen. Am 9. Dezember 1968 stellte er das Ergebnis seiner Forschungs- und Entwicklungsarbeit einem staunenden Fachpublikum vor: Ein interaktives Computersystem zur Unterstützung intellektueller Tätigkeiten, mit dem sich Dokumente erstellen und manipulieren ließen, die Text und Grafik beinhalteten, mit dem man komplexe Forschungs- und Entwicklungsprojekte planen und überwachen konnte und das es ermöglichte, Dokumente über ein Computernetzwerk an entfernte Computer zu versenden. Es handelte sich um ein Computersystem, das alle Komponenten umfaßte, die heute einen leistungsfähigen Personal Computer ausmachen. Trotzdem wurde das Projekt Ende 1977 eingestellt, ohne daß das System eine größere Verbreitung gefunden hätte.

5.1 Memex, Radar und eine Vision

Im Frühherbst 1945 war der damals zwanzigjährige Douglas Carl Engelbart (* 1925) als Radarelektroniker bei der amerikanischen Marine im Südpazifik stationiert. Während der langen Monate zwischen dem Ende des Krieges und seiner Demobilisierung las er in der Bibliothek des Roten Kreuzes auf Lahiti unter anderem Vannevar Bushs Aufsatz »As we may think«, der ihn nach eigener Aussage zwar beeindruckte, aber zunächst keine unmittelbaren Folgen für seinen beruflichen Weg hatte.¹

Nach dem Krieg setzte Engelbart sein durch die Einberufung unterbrochenes Studium der Elektrotechnik an der Oregon State University fort, das er 1948

¹ Engelbart 1986/87, Interview 1, Tape 1, Side A.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

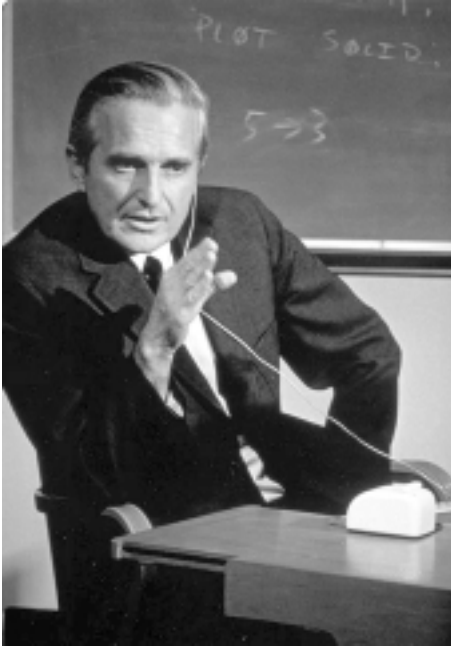


Abbildung 33. Douglas C. Engelbart um 1968

als *Bachelor of Science* abschloß. In den nächsten drei Jahren war Engelbart am *Ames Research Laboratory* in Mountain View, Ca. beschäftigt, das vom *National Advisory Committee for Aeronautics* (NACA) betrieben wurde, einer Vorgängerorganisation der NASA. Er war dort in einer Unterstützungs- und Serviceabteilung tätig, die nach Spezifikationen der Entwicklungsabteilung z. B. elektronische Steuerungen baute.¹

Schon nach wenigen Jahren spürte Engelbart, daß sich sein berufliches Leben in einer Sackgasse befand. Trotz der materiellen Sicherheit stellte er fest, daß seine Tätigkeit wenig aufregend war und keine neuen Herausforderungen bot. In dieser Situation wurde sich Engelbart bewußt, daß es genügend lohnendere Aufgaben für einen Ingenieur gab. Im Dezember 1950 entschloß er sich schließlich, nach neuen beruflichen Perspektiven Ausschau zu halten.²

Im folgenden halben Jahr widmete Engelbart seine Freizeit der Suche nach reizvollen beruflichen Zielen; dabei wollte er nicht weniger als einen Kreuzzug

¹ Lee 1995, S. 278; Engelbart 1986/87, Interview 1, Tape 1, Side A.

² Engelbart 1986/87, Interview 1, Tape 1, Side B – Obwohl diese heute von Engelbart gern erzählte Anekdote durchaus legendären Charakter hat, taucht sie bereits in der Einleitung eines Förderantrages von 1960 auf.

für eine bessere Welt führen.¹ Engelbart muß sich während dieser sechs Monate auch an Bushs »As we may think« erinnert haben², denn schließlich kam er zu der Einsicht, daß es ein lohnendes Ziel sei, ein Werkzeug zu entwickeln, mit dem sich die intellektuellen Probleme einer immer komplexeren Welt besser meistern lassen würden. Dies schien vor dem Hintergrund des Koreakriegs, mit dem sich die Welt nur sechs Jahre nach Ende des Zweiten Weltkrieges erstmals an der Schwelle einer nuklearen Auseinandersetzung befand, besonders dringlich.³ Engelbarts Vorstellungen über eine mögliche Realisierung eines solchen Werkzeugs orientierten sich an seinen Erfahrungen mit der Radartechnik während des Zweiten Weltkrieges. Er schrieb über die drei Aspekte seiner Motivation:

»I became motivated (committed) in 1951 to improving mankind's capability for dealing with its pressing problems, especially those over-taxing our collective capability to cope with complexity and urgency. I visualized people collaborating interactively on visual displays connected to a computer complex. I'm not »numerically oriented«; my vision has always facilitated discursive thinking and collaboration.«⁴

Innerhalb weniger Wochen entwickelte er das Konzept eines computerbasierten Systems zur Verstärkung der menschlichen Intelligenz (*augmentation of human intellect*)⁵, an deren Realisierung Engelbart mit unterschiedlicher Schwerpunktsetzung fortan arbeitete.

Ein erster Schritt auf diesem Wege war die Fortsetzung seiner Ausbildung, da Engelbart annahm, daß er seinen »Kreuzzug« innerhalb des etablierten Wissenschaftsbetriebs nur mit einem Dokortitel würde führen können. Engelbart schrieb sich bei der University of California in Berkeley ein, da diese im Gegensatz zur benachbarten Stanford University 1951 bereits über ein eigenes Computerprojekt

¹ Engelbart 1988, S. 188; Lindgren 1971, S. 54f.

² Obwohl es keine direkten Zeugnisse dafür gibt, wird in Engelbarts Schriften aus den sechziger Jahren gern und ausführlich aus »As we may think« zitiert. Auch stilistisch orientierte er sich an Bushs utopischer Darstellungsweise. Vgl. vor allem Engelbart 1962c, S. 47–56, 73–114 und Brief von Engelbart an Bush, 24. Mai 1962, SUL/EC, 6-15 (veröffentlicht in Nyce and Kahn 1991a, S. 235f.). Bushs Aufsatz hatte also – im Gegensatz zu Hans Dieter Helliges Einschätzung – zumindest in diesem Fall eine klare Leitbildfunktion. Vgl. Hellige 1996a, S. 208.

³ Engelbart 1988, S. 188f.

⁴ Engelbart 1995, S. 30; vgl. auch Engelbart 1960, S. 3.

⁵ Der von Engelbart benutzte Begriff läßt sich nicht unmittelbar ins Deutsche übertragen. Statt dessen wird der Begriff des »Intelligenzverstärkers« verwendet, wie ihn W. Ross Ashby 1956 definiert hat. Für Ashby war ein »Intelligenzverstärker« eine technische Einrichtung, die den Menschen bei umfangreichen und komplexen Selektionsvorgängen unterstützte, so daß die »Intelligenz« des Mensch-Maschine-Systems insgesamt höher lag als die des Individuums. Vgl. Ashby 1974, S. 253f. Engelbart hat diesen Begriff schon frühzeitig als für seine Ziele zutreffend bezeichnet. Vgl. Engelbart 1962c, S. 19.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

verfügte.¹ Engelbart mußte dort allerdings zum ersten Mal feststellen, daß die Computerentwickler wenig Interesse an seinen exotischen Ideen hatten, sie interessierten sich mehr für Computerarchitektur, Speichertechnik oder numerische Mathematik.² Notgedrungen ließ sich Engelbart darauf ein, Computerspeicher zu entwickeln, die auf der nichtlinearen Charakteristik von Gasentladungsröhren basierten und für die ihm 1956 der Doktorgrad in Elektrotechnik verliehen wurde. Die aus dieser Arbeit resultierenden Patente versuchte er 1956/57 mit wenig Erfolg in einem eigenen Unternehmen zu verwerten. Nach dem Scheitern seiner Firma nahm Engelbart im Oktober 1957 eine Position als Wissenschaftler bei der Ingenieurabteilung des *Stanford Research Institute* (SRI) in Menlo Park, Ca. an.³

Das Stanford Research Institute war 1946 von der Stanford University gegründet worden und führte angewandte Forschung durch, die für das akademische Umfeld der Universität als unpassend empfunden wurde.⁴ Damit gehört das SRI, ebenso wie das Lincoln Laboratory, die RAND Corporation oder SDC, zu den sogenannten *Think Tanks*, die während des Kalten Krieges einen Großteil der militärischen Forschung durchführten. Am SRI hatte man bereits während der fünfziger Jahre einige Erfahrung im Bereich der Datenverarbeitung sammeln können. Prominentestes Beispiel dürfte das ERMA-System zur maschinellen Erfassung von Schecks sein, das Mitte der fünfziger Jahre für die Bank of America entwickelt wurde.⁵ Engelbart hatte das Gefühl, daß eine solche Institution, die nicht den kommerziellen Zwängen der Industrie unterworfen war und ihn gleichzeitig nicht mit Lehraufgaben belastete, der richtige Ort wäre, um langfristig grundlegende Forschung zu betreiben.⁶

Nach einer Orientierungsphase, während der er sich mit den laufenden Forschungen des SRI vertraut machen konnte, beschäftigte sich Engelbart – in gewisser Fortsetzung seiner Doktorarbeit und seiner unternehmerischen Tätigkeit – mit der Entwicklung eines sogenannten *All-Magnetic Logic Systems*. Dabei wurden alle logischen Funktionen des Computers nicht in Form von elektronischen Schaltungen realisiert, sondern auf Grund der nichtlinearen Eigenschaften von magnetischen Materialien, die auch bei den sich damals gerade durchsetzenden

¹ Engelbart 1986/87, Interview 1, Tape 1, Side B und Tape 2, Side A. Dabei handelte es sich um die Entwicklung des CALDIC unter Leitung von Paul Morton. Vgl. Chapin 1963, S. 190.

² Engelbart 1986/87, Interview 1, Tape 1, Side B.

³ Engelbart 1986/87, Interview 1, Tape 2, Side B und Interview 2, Tape 1, Side A.

⁴ Reingold 1988, S. 292. – Die Verbindung zwischen der Stanford University und dem SRI blieben nach der Gründung nur locker. Insbesondere finanziell war das SRI vollständig unabhängig. Die Administration der Stanford University hatte lediglich Einfluß auf die Besetzung des Verwaltungsrates und der Universitätspräsident fungierte gleichzeitig als dessen Vorsitzender. 1970 wurden auch diese Verbindungen durchtrennt, mittlerweile trägt die Institution, die Niederlassungen in aller Welt unterhält, den Namen SRI International.

⁵ Fisher and McKenney 1993.

⁶ Engelbart 1986/87, Interview 2, Tape 1, Side A.

Magnetkernspeichern verwendet wurden. Diese Forschung wurde aber schon bald wieder aufgegeben, da sich der Transistor als leistungsfähiges, zuverlässiges und vergleichsweise billiges Grundbauelement des Computers Ende der fünfziger Jahre durchsetzte.¹ Während Engelbart im Jahre 1960/61 am Aufbau des »Computer Techniques Laboratory« mitwirkte, konnte er sich innerhalb des SRI einen gewissen Freiraum schaffen, da es ihm 1959 gelungen war, Fördergelder vom Büro für wissenschaftliche Forschung der amerikanischen Luftwaffe (*Air Force Office of Scientific Research*, AFOSR) einzuwerben.²

Die mathematische Abteilung des AFOSR unterstützte, anders als ihr Gegenstück beim *Office of Naval Research* während der vierziger Jahre, nur theoretische und konzeptionelle Projekte, die vor allem den Dialog zwischen Wissenschaftlern unterschiedlicher Fachbereiche anregen sollten. Dies hatte allerdings zur Folge, daß die Abteilung unter der Leitung von Harold Wooster und Rowena Swanson nur ein schmales Budget zur Verfügung hatte. Wooster und Swanson waren zwei außergewöhnliche Persönlichkeiten innerhalb der militärischen Forschungsförderungsverwaltung, ungemein belesen und für unkonventionelle Ideen aufgeschlossen, kurz die idealen Partner für Engelbart und seine Pläne.³ Die Fördersumme für sein Projekt war zwar mit etwa 25 000 \$ recht bescheiden, ermöglichte es ihm aber, die Hälfte seiner Arbeitszeit damit zuzubringen, über Möglichkeiten zur Intelligenzverstärkung mit Hilfe des Computers nachzudenken.⁴ In dieser konzeptionellen Phase, die bis 1962 dauerte, sichtete er die in unterschiedlichen wissenschaftlichen Disziplinen zu diesem Thema entwickelten Ansätze und untersuchte systematisch die für die Intelligenzverstärkung relevanten Faktoren.⁵

5.2 Die Quellen der Inspiration: Neue Theorien und altbekannte Maschinen

Bei dieser theoretischen Arbeit wurde Douglas Engelbart vor allem durch Erfahrungen mit einer dem *Rapid Selector* verwandten Maschine sowie durch Konzepte aus der »Künstlichen Intelligenz« und der Organisationswissenschaft beeinflusst.

¹ Notizen und Memos zum »All-Magnetic Logic System« (1958/59), SUL/EC, 3-3 und 3-4.

² Engelbart 1988, S. 190; Memorandum von Pat Conley an Torben Meisling über »Projects Relating to Man-Computer Program«, 21. Juli 1966, SUL/EC, 15-2. – Absender und Empfänger der internen SRI-Memoranden sind normalerweise mit ihren Initialen abgekürzt. Bei den Quellenangaben in diesem Kapitel wurden diese so weit wie möglich durch die kompletten Namen ersetzt.

³ Glasersfeld 1992; Forman 1987, S. 210.

⁴ Notizen und Memos über »Active Projects, Computer Techniques Laboratory«, ca. 1960, SUL/EC, 3-10.

⁵ Engelbart 1986/87, Interview 2, Tape 1, Side B; Memorandum von Pat Conley an Torben Meisling über »Projects Relating to Man-Computer Program«, 21. Juli 1966, SUL/EC, 15-2.

Mikrodokumentation

Im Oktober 1961 präsentierte Engelbart seine Ideen auf der Jahrestagung des *American Documentation Institute* erstmals der Öffentlichkeit. Er knüpfte dabei an das von Vannevar Bush beklagte Informationsproblem der Wissenschaft an, das – so die damalige Einschätzung – nach dem Sputnikschock noch dringlicher geworden war.¹ Er machte deutlich, daß es für die effektive Arbeit von Wissenschaftlern sowie von Entscheidungsträgern in Politik und Wirtschaft von entscheidender Bedeutung ist, daß die produzierte Information in geeigneter Weise erfaßt, klassifiziert und möglichst bald den Benutzern zugänglich gemacht wird, eine Aufgabe, die normalerweise von Bibliotheken übernommen wird. Damit war deren Dokumentationsystem ein wichtiges Mittel der Kommunikation zwischen Wissenschaftlern und ein mögliches Einsatzfeld für Engelbarts intelligenzverstärkende Verfahren.

Obwohl die Teamarbeit in der Wissenschaft immer wichtiger werde, so Engelbart, bleibe doch die problemorientierte Einzelperson von entscheidender Bedeutung. Wolle man also die Zusammenarbeit innerhalb eines Teams verbessern, so müsse man erst durch den Einsatz organisatorischer und technischer Verfahren den Einzelnen in die Lage versetzen, die für effiziente Teamarbeit notwendige Flexibilität zu erreichen.² Dabei unterstellte Engelbart zu dieser Zeit, daß sich die individuellen Verbesserungen auf höherem Niveau akkumulieren. Auch in dieser Ansicht knüpfte Engelbart an Ausführungen Bushs von 1945 an, die bereits auf Seite 65 diskutiert wurden.

Im folgenden stellte Engelbart einige Hypothesen über den Umgang eines Wissenschaftlers mit Information auf, die er durch Beobachtung seiner eigenen Arbeit gewonnen hatte. Demnach sind die Informationseinheiten, mit denen ein Wissenschaftler normalerweise umgeht, wesentlich kleiner als die Dokumente, für deren Verwaltung normale Dokumentationssysteme ausgelegt sind. Die Basis seiner Arbeit ist eine Sammlung von Fakten, Konzepten und Ideen, die nicht nur der Literatur entstammen, sondern auch der persönlichen Beobachtung, eigenen Überlegungen und der Kommunikation mit Kollegen.³

Ein technisches System, das die Effizienz des wissenschaftlichen Arbeitens unterstützen soll, müßte diese Faktoren berücksichtigen. Engelbart beschrieb die Eigenschaften dieses Systems folgendermaßen:

»The time constants from storage to retrieval would be shorter, the package size for information would be smaller, the number of different people involved as generator, retriever, and user would be much smaller, and the total amount of information would be smaller, but ... [it] would be still

¹ Bourne and Engelbart 1958; Kastens 1958; DuShane 1957; Anonymous 1957.

² Engelbart 1961c, S. 121f., 124.

³ Engelbart 1961c, S. 122.

large enough to require a respectable amount of sophistication in developing a good solution to this problem.«¹

Es ist nicht schwierig, in diesem System, das Engelbart als »Mikrodokumentationssystem« bezeichnete, Bushs Memex wiederzuentdecken, die persönliche Maschine, die dem Wissenschaftler helfen soll, sein individuelles Informationsproblem zu lösen.

Wie ein solches System zu realisieren sei, beschrieb Engelbart anhand einer von ihm seit 1954 verwendeten Technik. Dabei notierte er bibliographische Verweise, Ideen und Konzepte auf IBM-Karten, die er mit Hilfe von Randlochungen klassifizierte. Bei der Recherche nach bestimmten Informationen konnte er mit Hilfe einer einfachen mechanischen Vorrichtung die gesuchten Informationseinheiten wieder ausfindig machen. Dieses Verfahren ähnelte Calvin Mooers' *Zatocoding System*, einer Weiterentwicklung der bei Bushs *Rapid Selector* verwendeten Methoden.²

Zusammenfassend wies Engelbart darauf hin, daß zwar mit dem Digitalcomputer bereits eine für die Realisierung passende Technologie zur Verfügung stehe, daß aber zunächst die organisatorischen und prozeduralen Aspekte des Problems genauer untersucht werden müßten.³ Damit sind auch die beiden Bereiche genannt, die Bush bei seinen Entwürfen vernachlässigt hatte.

Engelbarts Vortrag hatte allerdings nicht den gewünschten Zuspruch, insbesondere fanden sich keine neuen Sponsoren.⁴ In der Diskussion wurde Engelbart gar vorgeworfen, nur ein weiteres System zur Informationswiedergewinnung entworfen zu haben, ohne über den Stand der Technik hinauszugehen. Ein Teilnehmer äußerte sogar, Engelbart tue sich mit seinen Ideen wichtig, anstatt ernsthaft zur Forschung beizutragen. Diese Episode wurde typisch für die Rezeption von Engelbarts Arbeit bis weit in die siebziger Jahre. Anstatt sich auf die *Gesamtheit* von Engelbarts Konzept einzulassen, nahmen Wissenschaftler häufig nur die Elemente wahr, die in ihr eigenes Fachgebiet fielen. Diese aus dem Zusammenhang gerissenen Aspekte waren dann tatsächlich bekannt oder gar trivial.⁵

Diese Episode deutet bereits eine grundlegende Charakteristik von Engelbarts Forschungs- und Managementstil an. Obwohl er weitreichende Ziele verfolgte und dabei anspruchsvolle theoretische und praktische Lösungsansätze verfolgte, gelang es ihm nur selten, seinen Mitarbeitern, Kollegen und Geldgebern die Bedeutung seiner Forschungen zu vermitteln. Häufig blieb es bei der geschilderten

¹ Engelbart 1961c, S. 122f.

² Engelbart 1961c, S. 123f.; Mooers 1950.

³ Engelbart 1961c, S. 124.

⁴ Memorandum von Engelbart an Reid Anderson, Jerre Noe, James Norton und Robert Wing über »Individual's Information Handling«, 26. Juli 1960, SUL/EC, 17-7.

⁵ Engelbart 1988, S. 190; Lindgren 1971, S. 56.

Wahrnehmung von oberflächlichen Details seines Forschungsprogramms oder gar bei völligem Unverständnis. Dies hatte unter anderem zur Folge, daß Engelbarts Arbeit nie wirklich in das Konzept des SRI integriert werden konnte. In späteren Jahren wurde das Problem dadurch abgemildert, daß bestimmte Mitarbeiter als Vermittler zwischen Engelbart, seinen Mitarbeitern, dem SRI-Management und der Fachöffentlichkeit auftraten.¹

Die Organisation wissenschaftlicher Arbeit

Durch seine Arbeit am *Computer Techniques Laboratory* lernte Engelbart Ende der fünfziger Jahre auch einige prominente Vertreter der KI-Forschung kennen: Allen Newell, Herbert Simon und Edward Feigenbaum von der Carnegie Mellon University in Pittsburgh, Clifford Shaw von der RAND Corporation sowie Daniel Bobrow vom MIT und Frank Rosenblatt vom *Cornell Aeronautical Laboratory* in Buffalo, NY. Gleichzeitig formierte sich auch am SRI eine KI-Forschungsgruppe, zu der Bertram Raphael und Charles Rosen gehörten.²

Mit Beginn seiner Forschungen zur Verstärkung der menschlichen Intelligenz begann sich Engelbart auch verstärkt für Fragen der KI zu interessieren. Ein lohnender Ausgangspunkt für seine eigenen Arbeiten schienen ihm die Arbeiten von Newell, Simon und Shaw zu sein, die Ende der fünfziger Jahre an einem *General Problem Solver* arbeiteten, der vor allem mit heuristischen Methoden, die dem menschlichen Problemlöseverhalten nachempfunden waren, unterschiedliche Probleme – mathematische Sätze, kryptologische Rätsel oder Denksportaufgaben – lösen konnte. Um eine möglichst breite Problempalette bearbeiten zu können, waren die verwendeten Verfahren aufgabenunabhängig implementiert. Außerdem mußte aber auch jedes Problem in einer formalen und damit für den Computer handhabbaren Weise formuliert werden.³ Obwohl dieser Ansatz schon bald als Irrweg in die Kritik geriet, schärfte er doch Engelbarts Sinn für die Bedeutung der menschlichen Problemlösestrategien und für die zentrale Bedeutung der Sprache für seinen geplanten Intelligenzverstärker.⁴

¹ Email von James C. Norton an den Autor, 29. Januar 1998 und Email von Dirk H. van Nouhuys an den Autor, 26. Januar 1998. Diese Tendenz ist auch in Engelbarts Schriften erkennbar: Die Mehrzahl der von Engelbart allein verfaßten Texte sind in ihrer Aussage weniger konkret als solche, die einen zweiten Autor hatten.

² Engelbart 1986/87, Interview 1, Tape 2, Side B, Interview 2, Tape 1, Side B. – Auch das SRI gehörte während der sechziger und siebziger Jahre zu den von der ARPA geförderten Centers of Excellence im Bereich der Künstlichen Intelligenz. Vgl. McCorduck 1987, S. 112.

³ McCorduck 1987, S. 203ff.; Newell et al. 1957; Newell et al. 1960.

⁴ Memorandum von Engelbart an H. Zeidler, R. Anderson, J. Noe, T. Morrin und W. McGuigan über »Program at SRI on »Individual Information-Handling Problems««, 21. Januar 1960, SUL/EC, 5-14; Brief von Engelbart an Fred Dion, Rome Air Development Center, 6. Juli 1961, SUL/EC, 5-2; Engelbart 1962b, S. 4.

Schließlich wurde Engelbart 1961 auf einen bereits 1956 erschienenen RAND-Forschungsbericht aufmerksam, der seiner Arbeit nach Monaten der Stagnation einen neuen An Schub gab. Die Autoren stellten bei ihrer organisationspsychologischen Untersuchung heraus, daß ein Hauptproblem interdisziplinärer Arbeitsgruppen darin bestehe, daß Wissenschaftler unterschiedlicher Disziplinen nur schwer miteinander kommunizieren könnten. Jede etablierte Disziplin besitze einen eigenen konzeptionellen Rahmen, der nach ihren individuellen Zielen und Werten gestaltet ist und eine effektive Fachkommunikation erst ermöglicht. Aus diesem Grund sei die Arbeit bei neuen – und deshalb meist interdisziplinären – Forschungsgebieten solange wenig ertragreich, bis ein neuer konzeptioneller Rahmen entstanden sei.¹

Zu ähnlichen Schlüssen kam Ende der fünfziger Jahre auch Thomas S. Kuhn, der zu dieser Zeit als junger Dozent in Berkeley an seinem Buch »The Structure of Scientific Revolutions« arbeitete. Der darin geprägte Begriff des Paradigma umfaßt auch Strukturen, die Kuhn später als »disziplinäres System« bezeichnet hat. Dieser bei Kuhn durchaus kritisch gemeinte Begriff kennzeichnet einen Mechanismus, der für das Funktionieren der »normalen Wissenschaft« eine zentrale Bedeutung hat, aber auch die Gefahr einer dogmatischen Erstarrung in sich trägt. Er bezeichnet ebenfalls nichts anderes als einen konzeptionellen Rahmen.²

Diese auffällige Häufung ähnlicher Ansätze zur Erklärung des menschlichen Verhaltens in Organisationen, zu denen auch noch Herbert Simons klassisches Buch über das Entscheidungsverhalten in Organisationen von 1947 zählt³, ist ein Hinweis, daß Engelbarts Überlegungen um 1960 durchaus nicht ungewöhnlich waren. Herbert Simon gab dem Autor sogar zu bedenken, daß nach seinem Empfinden solche Ideen in der wissenschaftlichen Kultur jener Zeit weit verbreitet waren.⁴

Trotz der scheinbaren Ähnlichkeiten zwischen dem Kuhnschen Paradigmenbegriff und Engelbarts konzeptionellen Rahmen besteht ein fundamentaler Unterschied zwischen beiden Konzepten. Während sich die Ingenieurwissenschaften nämlich damit beschäftigen, wie Dinge sein sollen, stellen sich die Naturwissenschaften die Frage, wie die Dinge sind.⁵ Eine Folgerung aus dieser Unterscheidung besteht nun darin, daß der Ingenieur stets subjektiv sein *muß*, da er selbst ein Bestandteil des Entwurfsprozesses ist. Demnach sind auch die Ziele, die er sich

¹ Kennedy and Putt 1956, S. 4ff.; auch Chapman et al. 1959.

² Kuhn 1984, S. 187ff., 194 und Kuhn 1992.

³ Simon 1981.

⁴ Email von Herbert A. Simon an den Autor, 4. August 1997; vgl. auch Hounshell 1997, insb. S. 258ff.

⁵ Simon 1994, S. 95.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

selbst für seine Arbeit setzt, gestaltbar und dabei der gleichen Entwurfsmethodik unterworfen.¹

Vor diesem Hintergrund interessierte sich Kuhn (als studierter Physiker) lediglich für die Beobachtung und Analyse der Bildung bzw. Ablösung eines Paradigmas, das aus der täglichen Arbeit von Wissenschaftlern entsteht. Für den Systemingenieur Engelbart war ein konzeptioneller Rahmen hingegen nicht nur veränderlich, sondern selbstverständlich auch gestaltbar.

Jedenfalls entschloß sich Engelbart, in einer »Suchphase« zunächst den konzeptionellen Rahmen, also den theoretischen Unterbau für seine Arbeit zu schaffen.² Engelbart begann daher, bei SRI und an der Stanford University interdisziplinäre Seminare abzuhalten, die sich mit der Frage befaßten, wie man die Effektivität eines Individuums oder einer Gruppe mit organisatorischen und technischen Verfahren steigern könne.³

An diesen Veranstaltungen nahmen auf Engelbarts Einladung hin auch Vertreter von Firmen und Organisationen wie IBM, Burroughs, 3M, General Electric, Lockheed oder der NASA teil, die nicht nur ihr Wissen über die Organisation geistiger Arbeit und ihre praktischen Erfahrungen einbringen sollten, sondern auch gezielt auf eine finanzielle Beteiligung an Engelbarts Projekt angesprochen wurden. Die Resonanz blieb freilich erneut gering. Der NASA war beispielsweise Engelbarts auf zehn Jahre angelegtes Programm für eine Beteiligung prinzipiell zu unspezifisch.⁴

Trotzdem schienen die Perspektiven für Engelbarts Projekt 1961 durchaus positiv zu sein. Durch die Förderung des Büros für wissenschaftliche Forschung der Luftwaffe war die Arbeit am konzeptionellen Rahmen sichergestellt. Außerdem hatten einige der Unternehmen, die an den SRI-Seminaren teilgenommen hatten, durchaus Interesse an einer Zusammenarbeit erkennen lassen. Die Industrie hatte aber mit der Förderung von Grundlagenforschung, zumal solcher mit einem recht diffusen Ziel, offenbar Schwierigkeiten. Obwohl Engelbart anbot, den industriellen Partnern etwaige Patente zur Verwertung zu überlassen, kam letztlich keine Kooperation zustande.⁵

¹ Evans 1969, S. 109f.

² Engelbart 1960, Anhang A, S. 2f.; Engelbart 1988, S. 189f.

³ Group Work SRI Seminars, »On Augmenting the Human Intellect«, 1960/61, SUL/EC, 2-17; Materialien, Memos und Protokolle zu den »Augmented Human Seminars« und »Artificial Intelligence Seminars«, 1961, SUL/EC, 5-8 bis 5-12.

⁴ Engelbart 1960, Anhang 1, S. 1; Engelbart, D. C., »Technical Status Report for the Air Force Office of Aero Space Research«, 28 August 1961, SUL/EC, 5-14; Engelbarts Aufzeichnungen über Reaktionen der NASA, ca. 1960, SUL/EC, 6-2.

⁵ Briefe von Engelbart an Fred Dion, Rome Air Development Center, 25. Juli und 25. August 1961, SUL/EC, 5-2; Engelbart, D. C., »Proposal for Participation in the Program of Human Effectiveness – Draft«, 24. August 1961, SUL/EC, 5-8.

Schließlich war im Juni 1961 auch Frederick A. Dion vom *Rome Air Development Center* (RADC) der Luftwaffe in Rome, NY auf Engelbarts Arbeiten aufmerksam geworden. Dion stellte weitere 20 000 \$ in Aussicht, mit denen das Forschungsprogramm ausgeweitet werden sollte. Dabei sollten Engelbarts *Bootstrap-Programm* (siehe Abschnitt 5.3.3) sowie Untersuchungen zur Vermittlung von psychomotorischen Fertigkeiten mit Hilfe des Computers (siehe Abschnitt 5.5) gefördert werden. Nachdem der Vertrag mit dem RADC fast unterschriftsreif war, wurde der Antrag Ende November 1961 überraschenderweise mit dem Hinweis abgelehnt, das RADC fördere bereits ein ähnliches Projekt.¹ Nachdem auch ein weiterer Förderantrag vom amerikanischen Bundesgesundheitsamt (*National Institute of Health*) abgelehnt worden war², mußte Engelbart seine Ambitionen zunächst wieder zurückschrauben. Rückblickend äußerte er, die Jahre 1961 und 1962 seien »nicht nur schweißtreibend, sondern auch einsam« gewesen.³

Und so blieb es auch noch einige Zeit. In den Jahren bis 1965 war Engelbart der einzige Wissenschaftler, der sich ganztägig mit der Konzeptionierung und Realisierung des Intelligenzverstärkers beschäftigte. Erst nachdem eine massive Förderung durch die ARPA und andere finanzkräftige (staatliche) Organisationen einsetzte, konnte das sogenannte *Augmentation Research Center* (ARC) die Dynamik entwickeln, die sich Engelbart wünschte.⁴

5.3 »Augmenting Human Intellect: A Conceptual Framework«

Das Ergebnis seiner Arbeit in den Jahren 1961/62, seiner Literaturrecherche, seiner Gespräche mit Wissenschaftlern aus verschiedensten Disziplinen und der SRI-Seminare, stellte Engelbart im Oktober 1962 in seinem 130seitigen Abschlußbericht an das Büro für wissenschaftliche Forschung vor.⁵ Im Frühjahr 1963 wurde

¹ Korrespondenz zwischen Engelbart und Fred Dion, RADC, 20. Juni bis 13. Dezember 1961, SUL/EC, 5-2 und 6-11.

² Engelbart 1962a.

³ Engelbart 1988, S. 191.

⁴ Engelbart and SRI-ARC Staff 1968 – Im Lauf der Jahre hat sich der Name für Engelbarts Projekt, die daraus hervorgegangene Gruppe bzw. die spätere SRI-Abteilung mehrfach verändert. Während das Projekt bis Mitte der sechziger Jahre zunächst keinen feststehenden Namen hatte, wurde es nach den beiden damaligen Sponsoren ARPA und NASA zunächst ARNAS genannt. 1967 wurde die Bezeichnung »Augmented Human Intellect Research Center« (AHIRC) geprägt, die Ende der sechziger Jahre zu »Augmentation Research Center« (ARC) abgekürzt wurde. Während der siebziger Jahre wandelte sich der Name wegen der veränderten Forschungsschwerpunkte zum »Augmented Knowledge Workshop« (AKW). Aus Gründen der Einheitlichkeit wird im folgenden die Bezeichnung »Augmentation Research Center« bzw. ARC verwendet. Vgl. Engelbart 1986/87, Interview 3, Tape 1, Side A.

⁵ Engelbart 1962c.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

er in gekürzter Form als Beitrag in einem Sammelband auch einer breiteren Öffentlichkeit zugänglich gemacht.¹

Engelbarts »Augmenting Human Intellect: A Conceptual Framework« ist ein problematischer Text, intellektuell anspruchsvoll und vielfach nur schwer zu lesen, in zentralen Punkten häufig widersprüchlich und angreifbar. Vor allem aber ist der Text gerade so vage und ungenau bei der Definition konkreter Realisierungsvorschläge, daß er sich nach vielen Seiten hin auslegen läßt. Es verwundert daher nicht, daß selbst im ARC der sechziger Jahre Engelbarts eigene grundlegende Arbeit häufig nicht verstanden wurde.² Man erhält vielmehr den Eindruck, daß die Arbeit an seinem konzeptionellen Rahmen vor allem für Engelbarts eigene Orientierung und Zieldefinition wichtig war und weniger als unmittelbare Grundlage der späteren konkreten Entwicklungsarbeiten.

Engelbart selbst hält nach über dreißig Jahren seine damaligen Ideen für so gut und aktuell wie 1962, und seine Zielsetzung liest sich – ähnlich wie die programmatischen Arbeiten der Künstlichen Intelligenz dieser Zeit – auch heute noch wie ein Versprechen über die grenzenlosen Möglichkeiten des Computers zum Nutzen der Allgemeinheit:³

»By ›augmenting human intellect‹ we mean increasing the capability of a man to approach a complex problem situation, to gain comprehension to suit his particular needs, and to derive solutions to problems. Increased capability in this respect is taken to mean a mixture of the following: more-rapid comprehension, better comprehension, the possibility of gaining a useful degree of comprehension in a situation that previously was too complex, speedier solutions, better solutions, and the possibility of finding solutions to problems that before seemed insoluble. And by ›complex situations‹ we include the professional problems of diplomats, executives, social scientists, life scientists, physical scientists, attorneys, designers – whether the problem situation exists for twenty minutes or twenty years. We do not speak of isolated clever tricks that help in particular situations. We refer to a way of life in an integrated domain where hunches, cut-and-try, intangibles, and the human ›feel for a situation‹ usefully co-exist with powerful concepts, streamlined terminology and notation, sophisticated methods, and high-powered electronic aids.«⁴

¹ Engelbart 1963.

² Evans 1969, S. 127. David Evans, Engelbarts vielleicht intellektuellster Mitarbeiter zu Ende der sechziger Jahre, drückt diese Kritik in seiner Dissertation deutlich aus: Obwohl zentrale Begriffe ausgiebig definiert werden, werden sie doch in der Folge nicht immer konsistent verwendet. Einige Teile des konzeptionellen Rahmens würden einfach schlecht präsentiert.

³ Engelbart 1988, S. 190f.

⁴ Engelbart 1962c, S. 1.

Am Anfang von Engelbarts Arbeit stand – wie schon bei Vannevar Bush – der Wunsch nach einem universellen Werkzeug für intellektuelle Aufgaben, denen sich der Systementwurf anzupassen hat. Dabei nahm er an, daß es eine (kleine) Anzahl von grundlegenden Verfahren und Werkzeugen gibt, die sich so kombinieren lassen, daß sich schließlich beliebige Aufgaben bewältigen lassen. Engelbarts Ziel war es, letztlich einen Satz von Verfahren und Werkzeugen zu entwickeln, die sich des Digitalcomputers bedienen und der bisherigen Technik zur Unterstützung von geistiger Arbeit überlegen sind. So ging Engelbart auch bewußt über die Ziele der damaligen KI-Forschung hinaus, die sich zunächst die Lösung von sehr speziellen Problemen zur Aufgabe gemacht hatte und dazu auf heuristische Verfahren zurückgriff.

Engelbart fand ein Metapher für die Rolle, die sein System für den Benutzer spielen soll. Für Engelbart war dies der »Sekretär« (engl. *clerk*), der über die Arbeiten seines Chefs genau Bescheid weiß, ihm die dafür notwendigen Informationen aufbereitet und Routinarbeiten übernimmt. Bezeichnenderweise ist der Benutzer, den Engelbart in seinem Beispiel anführt, kein Naturwissenschaftler wie bei Bush oder Licklider, sondern ein Architekt, dessen Tätigkeiten breiter gestreut sind. Er muß nicht nur Daten analysieren und daraus physikalische oder mathematische Modelle ableiten, sondern mit Hilfe des *clerk* auch Entwürfe erstellen, Kosten berechnen, Personal koordinieren und Termine verwalten, kurz er benutzt den Computer als Managementwerkzeug.¹ Damit wird auch klar, daß Engelbart einen Typus von Nutzer im Auge hatte, den Peter F. Drucker Ende der sechziger Jahre als Informationsarbeiter (*Knowledge Worker*) bezeichnete.²

Um seine Prämissen theoretisch weiter zu untermauern und ein Konzept zur technischen Realisierung vorschlagen zu können, stellte Engelbart Hypothesen über das Wesen und die Quelle der Intelligenz auf, die über die angeborene oder natürliche Intelligenz des Menschen hinausreicht.

Engelbart bediente sich dazu einer kybernetischen Sichtweise von soziotechnischen Systemen, d. h. Organisationen, die sich aus Menschen und technischen Artefakten so zusammensetzen, daß sie bestimmte Aufgaben bestmöglich bewältigen können. Obwohl die einzelnen Systemkomponenten allein nur eine geringe Leistungsfähigkeit bzw. Intelligenz besitzen, ist die Leistungsfähigkeit bzw. Intelligenz des Gesamtsystems durch die Wirkung der Synergetik größer als die Summe der Einzelleistungen. Folglich, so die Einschätzung Engelbarts, sei die Organisation die eigentliche Quelle aller Intelligenz, die über die angeborene oder natürliche Intelligenz des Menschen hinausreiche. Schließlich unterstellte er, daß die reibungslose Zusammenarbeit aller Menschen und technischen Artefakte innerhalb des soziotechnischen Systems von einem möglichst verlustlosen Austausch

¹ Engelbart 1962c, S. 4ff.

² Drucker 1969; Engelbart et al. 1973, S. 9.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

von Energie und Informationen abhängig sei. Deshalb nehme die Sprache eine zentrale Rolle beim Systementwurf ein.¹

5.3.1 Der Computer als Medium

Es wurde bereits im vorigen Kapitel angedeutet, daß die Vertreter der KI den Computer als ein neues Medium betrachteten, nämlich als Erweiterung des menschlichen Gehirns. Dabei entwickelten sich rasch zwei unterschiedliche Sichtweisen. Der Großteil der KI-Wissenschaftler betrachtete den Computer als eine autonome Einheit, die im wesentlichen mit der Außenwelt kommuniziert. Engelbart – wie auch Licklider – war hingegen Vertreter einer anderen Sichtweise und ging davon aus, daß der Computer und sein Benutzer gleichberechtigte Partner sein müßten und die notwendige Kommunikation deshalb vor allem zwischen den Partnern abläuft. Für beide Sichtweisen ist es zentral, daß der Computer eine grundlegende menschliche Fertigkeit beherrscht: die Kommunikation durch Sprache. Ob dadurch die Grenzlinie zwischen Mensch und Maschine neu definiert werden müsse, wurde zum Gegenstand einer lang andauernden philosophischen Grundsatzdebatte.² Unabhängig vom Standpunkt des einzelnen Wissenschaftlers brachte diese Debatte das neue Paradigma der Mensch-Computer-Interaktion als kommunikativen Akt hervor:

»Prior styles of interaction between people and machines – such as driver and automobile, secretary and typewriter, or operator and control room – are all extremely lean: there is a limited range of tasks to be accomplished and a narrow range of means (wheels, levers and knobs) for accomplishing them. The notion of the *operator* of a machine arose out of this context. But the user is not an operator. He does not operate the computer, he communicates with it to accomplish a task. Thus we are creating a new arena of human action: communication *with* machines rather than operation *of* machines.«³

Die grundlegende Debatte über das Wesen des Computers eröffnete allerdings interpretative Freiräume in der Frage, wie diese Kommunikation aussehen sollte. Neben das einfache Konversationsmodell trat dabei zunehmend die Vorstellung, daß der Computer ein beliebig gestaltbarer Werkstoff sei, dessen Wesen zwischen

¹ Engelbart 1962c, S. 17f.; Engelbart 1963, S. 9f.; Evans 1969, S. 129.

² Bardini and Horvath 1995, S. 58f.; Winograd 1991; Newell 1991b. Vgl. auch Memorandum von Engelbart an H. Zeidler, R. Anderson, J. Noe, T. Morrin und W. McGuigan über »Program at SRI on ›Individual Information-Handling Problems‹«, 21. Januar 1960, SUL/EC, 5-14.

³ Card et al. 1983, S. 7. Hervorhebung im Original.

Intelligenz und Mechanik angesiedelt sei. Damit ist die von Licklider beschworene symbiotische Verbindung von Mensch und Maschine das Gegenbild zur mechanistischen Vorstellung des Cyborg, des mechanischen Menschen. Nachdem dieser interpretative Freiraum einmal erkannt war, stellte sich – ganz unabhängig vom Wesen der Mensch-Computer-Beziehung – die Frage nach der Gestaltung der Kommunikation von Mensch und Computer. Dies führte dazu, daß man den Computer als Medium zu begreifen begann. J. C. R. Licklider und Robert Taylor beschrieben dies folgendermaßen:

»Creative, interactive communication requires a plastic or moldable medium that can be modelled, a dynamic medium in which premises will flow into consequences, and above all a common medium that can be contributed to and experimented with by all.

Such a medium is at hand – the programmed digital computer. Its presence can change the nature and value of communication even more profoundly than the printing press and the picture tube, for as we shall show, a well-programmed computer can provide access both the informational and to the *processes* for making use of the resources.«¹

Den Computer als Medium zu betrachten, beendete freilich nicht die Debatte über das Wesen des Computers, sondern hob sie nur auf eine andere Ebene. Wenn der Computer ein Medium ist, kann man sich ihn als Mittel zur Kommunikation zwischen Menschen vorstellen: Der kommunikative Akt, der ursprünglich zwischen Mensch und Maschine stattfand, wird durch eine rein menschliche Kommunikation ersetzt, die durch die Technik nur noch *vermittelt* wird. Ähnlich wie bei anderen Medien, erkannte man nur langsam das Potential des Computer-Mediums, das sich erst durch die beginnende Computer-Vernetzung seit Anfang der siebziger Jahre voll entfalten konnte.

In dieser Perspektive ermöglicht der Computer den Zugang zu einem Netzwerk ähnlich ausgestatteter Individuen. Die kommunikative Kopplung wird zur Schnittstelle, einem Mittel zur Eingabe, Übersetzung und Verteilung von Informationen über das Netzwerk. Der Computer wird damit zu einem Werkzeug zur Kodierung und Dekodierung einer zeitlich oder räumlich verteilten Kommunikation. Die Stärke des neuen Mediums liegt in seiner beliebigen Gestaltbarkeit; sie ermöglicht, daß der Mensch so mit dem Computer interagieren kann, als würde er direkt (unvermittelt) mit seinem Gegenüber kommunizieren.²

¹ Licklider et al. 1968, S. 22. Hervorhebung im Original.

² Ong 1982, passim.

5.3.2 Eine universelle Sprache für die Mensch-Computer-Kommunikation

Douglas Engelbart war um 1960 einer der ersten Computerwissenschaftler, die den medialen Charakter des Computers betonten und dabei zeitgenössische linguistische Forschungsansätze für die Computerentwicklung fruchtbar machten. Er war sich schon 1960 sicher, daß die Sprache für den von ihm geplanten Intelligenzverstärker zentrale Bedeutung besitzen würde. Eines seiner ersten Ziele war deshalb, »[to] construct a general purpose signal language that makes our various directions convenient to communicate.« Da die Flexibilität des *clerk* von der Mächtigkeit dieser Sprache abhängig sei, müßte diese einen Umfang besitzen, die die Formulierung eines großen Repertoires an Problemen erlaubt, ohne den Benutzer allzuviel an Aufwand und Zeit zu kosten. Im Idealfall sollte der *clerk* wissen, was der menschliche Nutzer *wirklich* will, er sollte unscharf formulierte Probleme in den Zusammenhang der laufenden Arbeiten einordnen und mit ungenauen bzw. unvollständigen Daten umgehen können.¹

Diesen Forderungen stand der typische Digitalcomputer des Jahres 1960 gegenüber, mit dem der Benutzer nur über wenige Maschinenbefehle kommunizieren konnte, die zudem an den grundlegenden Funktionen der Hardware orientiert waren. Mit FORTRAN (1957), ALGOL und COBOL (beide 1960) waren erst seit kurzem problemorientierte Programmiersprachen für mathematische und kaufmännische Zwecke in Verwendung. Für Probleme der Künstlichen Intelligenz waren etwa zur gleichen Zeit zwei listenverarbeitende Sprachen entwickelt worden, LISP (1958) von John McCarthy und IPL (1957) von Simon, Newell und Shaw, die sich besonders für nichtnumerische Probleme eigneten. Engelbart hatte diese Entwicklung Ende der fünfziger Jahre aufmerksam verfolgt und sich im Rahmen der SRI-Seminare intensiv mit IPL beschäftigt. So erklärt sich sein Optimismus, vom schnellen Fortschritt bei der Entwicklung höherer Programmiersprachen profitieren zu können.²

Engelbart über Sprache

Nach Engelbarts Vorstellung ist die Interaktion des Menschen mit dem Computer durch den Austausch von Energie und Information gekennzeichnet. Diese Austauschprozesse finden innerhalb eines soziotechnischen Systems statt, das Engelbart schwerfällig als »H-LAM/T-System« bezeichnete, was für »Human using

¹ Memorandum von Engelbart an H. Zeidler, R. Anderson, J. Noe, T. Morrin und W. McGuigan über »Program at SRI on ›Individual Information-Handling Problems««, 21. Januar 1960, SUL/EC, 5-14. Zitat auf S. 2.

² Engelbart 1962c, S. 64f.; Engelbart, D. C., »Technical Status Report for the Air Force Office of Aero Space Research«, 28. August 1961, SUL/EC, 5-14. Zu den frühen Vorstellungen der KI-Wissenschaftler zur Bedeutung der Sprache siehe u. a. McCarthy 1958.

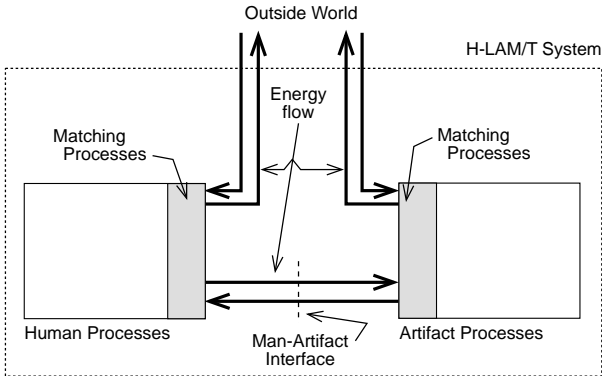


Abbildung 34. Struktur des H-LAM/T-Systems (1962)

Language, Artifacts and Methodology in which he is trained« stand.¹ Abbildung 34 zeigt, wie sich Engelbart die Struktur dieses Systems und die darin stattfindenden Austauschvorgänge vorgestellt hat.

In dieser schematischen Darstellung repräsentieren die Pfeile den Energie- bzw. Informationsfluß zwischen den unterschiedlichen Systembereichen und der Außenwelt, wobei die grauen Bereiche die an der Kommunikation beteiligten Prozesse darstellen. Für Engelbart können diese explizit dem Menschen bzw. der Maschine zugeordneten Prozesse durch die Verwendung einer beiden gemeinsamen Sprache »kompatibel« gemacht werden. Erst durch die gemeinsame Sprache wird der Computer je nach Sichtweise zur Erweiterung des menschlichen Gehirns oder zum Medium. Engelbart äußerte dazu in einem Interview:

»I remember the revelation to me when I was saying, »Let's look at all the other things that probably are out there in the form of tools,« and pretty soon focusing on language; realizing how much there was already that is added to our basic capability (...) It amounts to an immense system that you essentially can say augments the basic human being.«²

Im gleichen Interview machte Engelbart auch deutlich, daß während der Entstehung seines konzeptionellen Rahmens eine wichtige Veränderung seiner Vorstellungen über intelligenzverstärkende Methoden stattgefunden habe. Er kam nämlich zu der Erkenntnis, daß Sprache mehr als die Manipulation von Symbolen nach bestimmten Regeln ist und daß das elementarste Mittel zur Intelligenzverstärkung in einer Koevolution, also der gegenseitigen Anpassung beider Systembereiche liegt. Welche Bedeutung der »Sprache« dabei zukam, formulierte Engelbart folgendermaßen:

¹ Engelbart 1962c, S. 15f.; Engelbart 1963, S. 8f.

² Engelbart 1986/87, Interview 3, Tape 1, Side B.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

»*Language* – the way in which the individual parcels out the picture of his world into the concepts that his mind uses to model the world, and the symbols that he attaches to those concepts and uses in consciously manipulating the concepts (›thinking‹).«¹

Diese Definition liefert zwei mögliche Ebenen, um die Bedeutung der »Sprache« zu verstehen. Sprache kann entweder dazu verwendet werden, bereits existierende Konzepte abzubilden. Sie kann aber auch dazu verwendet werden, um ein neues bestimmtes Weltbild zu modellieren *und* gleichzeitig darzustellen. Der Computer wird so zu einem eigenständigen Medium, das ganz spezifische Eigenschaften und eine eigene »Grammatik« besitzt. Etwa zur gleichen Zeit brachte der kanadische Medientheoretiker Marshall McLuhan diese Erkenntnis in seinem bekannten Slogan »Das Medium ist die Botschaft« zum Ausdruck, die besagt, daß »der ›Inhalt‹ jedes Mediums immer ein anderes Medium ist.«² Eine ähnliche Vorstellung brachte Engelbart im folgenden Zitat auf den Punkt:

»A natural language provides its user with a ready-made structure of concepts that establishes a basic mental structure, and that allows relatively flexible, general-purpose concept structuring. Our concept of ›language‹ as one of the basic means for augmenting the human intellect embraces all of the concept structuring which the human may make use of... The other important part of our ›language‹ is the way in which concepts are represented – the symbols and *symbol structures*.«³

Linguistisches Relativitätsprinzip und whorfsche Hypothese

Die wohl wichtigste Quelle für Engelbarts Ansichten über die Bedeutung der Sprache für das menschliche Denken im allgemeinen und für die Entwicklung eines Intelligenzverstärkers im besonderen waren die ethnolinguistischen und sprachphilosophischen Arbeiten von Benjamin Lee Whorf (1897–1941). Whorfs wichtigste Schriften waren 1956 erstmals in gesammelter Form erschienen und erfreuten sich für einige Jahre bei Wissenschaftlern an der Grenze zwischen Psychologie, Kybernetik und Computerwissenschaft großer Beliebtheit.

Whorf hatte sich in den dreißiger und vierziger Jahren mit dem Zusammenhang zwischen Sprache, Denken und Wirklichkeit beschäftigt. Dieses Thema war alles andere als neu, es gehört zu den ältesten philosophischen Fragen der Menschheit und ist auch heute nicht beantwortet. In langen Wellenzügen wurde immer wieder behauptet, daß entweder das Denken der Sprache vorangehe

¹ Engelbart 1962c, S. 9. Hervorhebung im Original.

² McLuhan 1992, S. 18. In einem Interview mit dem kanadischen Technikhistoriker Thierry Bardini (15. Dezember 1992 in Fremont, CA) erklärte Engelbart, er habe bereits zur Entstehungszeit seines konzeptionellen Rahmens Kenntnis über McLuhans Thesen gehabt.

³ Engelbart 1962c, S. 35. Hervorhebung im Original.

oder aber die Sprache das Denken determiniere.¹ Whorf glaubte, daß die zentralen Konzepte einer Kultur in der Sprache eingebettet sind und daß alle Mitglieder diese Konzepte beim Erlernen ihrer Sprache verinnerlichen. Mit anderen Worten beinhaltet der Spracherwerb auch das Lernen einer bestimmten Denkweise und Weltsicht.² Daraus leitete Whorf das sogenannte »linguistische Relativitätsprinzip« ab, das besagt, daß die Gedanken des Menschen »von unerbittlichen Strukturgesetzen gesteuert werden, die ihm unbewußt bleiben.« Diese würden erst sichtbar, wenn man verschiedene Sprachen miteinander vergleicht.³

Die von Whorf beeinflussten Kognitionswissenschaftler kamen zu dem Ergebnis, daß die Sprache eine vom Inhalt unabhängige hierarchische Baumstruktur aufweise. Damit, so die Vorstellung, könne man menschliche Sprache grundsätzlich auch für den Computer zugänglich und verständlich machen.⁴

Ferner machten sie sich bewußt, daß Sprache nicht nur ein Mittel ist, um das menschliche Wissen zusammenhalten zu können, sondern auch ein Metawerkzeug zur Planung von Handlungsplänen darstellt.⁵ Den zweiten Gedanken griff Engelbart für seinen konzeptionellen Rahmen auf und erweiterte ihn dahin, daß er ihn nicht nur auf die Sprache, sondern auch auf technische Artefakte wie den Computer anwendete:

»The Whorfian hypothesis states that ›the world view of a culture is limited by the structure of the language which this culture uses.‹ But there seems to be another factor to consider in the evolution of language and human reasoning ability. We offer the following hypothesis, which is related to the Whorfian hypothesis: Both the language used by a culture, and the capability for effective intellectual activity are directly affected during their evolution by the means by which individuals control the external manipulation of symbols.«⁶

¹ Bolter 1990, Kapitel 8.

² Whorf 1997, S. 7–18. Als Beleg führte Whorf an, daß nach seiner Beobachtung die Hopi-Indianer über keine Wörter oder andere linguistische Konstrukte verfügten, mit der sich zeitliche Beziehungen ausdrücken ließen. Whorf behauptete, daß deshalb die Weltanschauung der Hopis das Konzept der Zeit, das für die westliche Kultur von eminenter Bedeutung ist, nicht kennt. Diese ethnolinguistische Beweisführung gilt mittlerweile als widerlegt. Zu diesem Thema sowie zur Rezeptionsgeschichte von Whorfs Arbeiten vergleiche Dürbeck 1975 und Lee 1996.

³ Whorf 1997, S. 52. Damit kommt Whorf zu ähnlichen Ergebnissen wie der Philosoph Ludwig Wittgenstein (1989), in dessen »Tractatus logico-philosophicus« es heißt: »5.6 Die Grenzen meiner Sprache bedeuten die Grenzen meiner Welt«.

⁴ Chomsky 1956; Miller et al. 1991, S. 140ff; Wessells 1990, S. 36ff., 297f. – Heike Stach hat darauf hingewiesen, daß die Analogien, die Mitte der fünfziger Jahre zwischen der menschlichen Sprache und den damaligen »Codiersystemen« hergestellt wurden, ein auslösendes und prägendes Element bei der Entwicklung von höheren Programmiersprachen darstellte. Vgl. Stach 1996, S. 51f.

⁵ Miller et al. 1991, S. 139, 153f.

⁶ Engelbart 1962c, S. 24.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

Engelbarts Auslegung der whorfschen Hypothese steht sowohl in Einklang mit den Hauptströmungen der klassischen europäischen Philosophie als auch mit der modernen soziologischen Theorie.¹ Insbesondere seine Verwendung des Begriffs »Weltsicht« verweist stark auf die geistigen Vorgänger Whorfs, insbesondere auf Wilhelm von Humboldt, der, wie Whorf, sein Wissen über nicht-indoeuropäische Sprachen mit einem breiten philosophischen Hintergrund kombiniert hatte.²

Durch seine Interpretation der whorfschen Hypothese postulierte Engelbart in der Tat ein dialektisches Verhältnis zwischen den beiden bereits erwähnten untergeordneten Sprachebenen: Die symbolische *Repräsentation* eines Gedanken beeinflusst demnach die durch sie begründete Weltsicht. Deshalb müßte, so Engelbarts Schlußfolgerung, auch die Computerdarstellung eine neue Weltsicht hervorbringen:

»If [our neo-whorfian] hypothesis could be proved readily, and if we could see how our means of externally manipulating symbols influence both our language and our way of thinking, then we would have a valuable instrument for studying human-augmentation possibilities.«³

Während Whorf die Auffassung vertrat, daß die Sprache das Denken strukturiere, stellte Engelbart die These auf, daß auch technische Artefakte in ähnlicher Weise wirken können. Sollte sich diese These verifizieren lassen, wäre es möglich »Denkwerkzeuge« zu konstruieren, die keine kognitiven Prozesse *nachahmen*, sondern die natürliche Intelligenz des Menschen *verstärken* können, so wie Werkzeuge die manuellen Fähigkeiten des Menschen zu verstärken vermögen.

Diese Aussagen deuten bereits an, daß Engelbart sein Forschungsprogramm in mehreren Phasen durchzuführen plante. Zunächst sollte nachgewiesen werden, daß die neo-whorfsche Hypothese zutreffend ist, um dann einzelne Verfahren und schließlich ein komplettes System zur Intelligenzverstärkung aufzubauen. Diese Vorgehensweise macht gleichzeitig klar, daß Engelbart eine evolutionäre Entwurfsmethodik anwenden wollte, bei der stets die postulierten mit den realen Effekten einer Entwicklung verglichen werden. Nur so konnte das Risiko verringert werden, das ambitionierte Forschungsziel durch eine falsche Entwicklungsrichtung zu verfehlen. Diese Methodik nannte Engelbart *Bootstrapping*.

¹ Fishman 1980. – In einem früheren Aufsatz stellte Fishman unter anderem fest »[that] there is no doubt in my mind that the Whorfian hypothesis and its stress on the linguistic relativism of cognitive processes was one of the <cutting edges> of the language sciences in the 1950s and early 1960s (...) No matter how limited the validity of his claim, and several decades of sympathetic research have produced almost no convincing confirmatory evidence, Whorf deserves to be seen in the illustrious company of the other great detractors, among whom the other social and behavioral scientists (Marx and Freud) may have been no more correct, even though they became much more famous than Whorf.« Vgl. Fishman 1977, S. 52f.

² Dürbeck 1975, S. 5ff.

³ Engelbart 1962c, S. 24.

5.3.3 Bootstrapping: Eine evolutionäre Entwicklungsmethodik

Es wurde Engelbart schnell bewußt, auf was für ein risikoreiches und langwieriges Projekt er sich eingelassen hatte, als er sich entschloß, aus dem Nichts ein vollständiges System zur Unterstützung von geistigen Tätigkeiten zu entwerfen. Nicht nur, daß sein komplettes Programm auf einer bislang unbewiesenen Hypothese basierte, seine Studie hatte auch eine Vielzahl von möglichen Einflußfaktoren aufgezeigt, die zu analysieren und beim Systementwurf zu berücksichtigen waren. Weiterhin reichte das Spektrum der anvisierten Nutzungsbereiche von der Arbeit des Naturwissenschaftlers und Ingenieurs bis hin zu Managementaufgaben. Alle diese Bereiche sollten mit einem einzigen technischen System unterstützt werden können. Schließlich war die Entwicklung der Computertechnik zu Beginn der sechziger Jahre – wie die Bemerkungen zu den Computersprachen angedeutet haben – noch weit davon entfernt, für die sich stellenden Aufgaben adäquate Hard- und Software bereitstellen zu können.

Aus diesen Gründen entschloß sich Engelbart 1961, eine evolutionäre und rekursive Entwurfsmethodik zu wählen, die er *Bootstrapping* nannte. Ihr Wesen besteht darin, Werkzeuge und Methoden zu entwickeln, die dazu verwendet werden können, bessere Werkzeuge und Methoden zur Problemlösung zu entwickeln.¹ Dabei ist die Methode des *Bootstrapping* weder auf Engelbarts Arbeiten noch auf die Computertechnik beschränkt. So weist beispielsweise Karl Marx darauf hin, daß Werkzeugmaschinen für die industrielle Revolution im 19. Jahrhundert eine entscheidende Rolle spielen, weil es mit ihnen möglich war, bessere und leistungsfähigere Werkzeuge und Werkzeugmaschinen herzustellen, die für eine expandierende industrielle Produktion benötigt wurden.²

Um zur Lösung eines Problems zu gelangen, wird dieses zunächst in weniger allgemeine Teilprobleme aufgeteilt, die mit den zur Verfügung stehenden Werkzeugen und Methoden einfacher lösbar sind als das Hauptproblem. Auch bei der Lösung der Teilprobleme kann wieder nach dem gleichen Schema vorgegangen werden. Die Lösung der Teilprobleme und die während der Lösung gewonnenen methodischen Erfahrungen können dann dazu genutzt werden, übergeordnete Probleme zu lösen, bis schließlich das Oberziel erreicht ist. Können Zwischenergebnisse nicht zufriedenstellend zur Lösung übergeordneter Probleme beitragen, wird versucht, mit der gleichen Methode eine alternative bzw. bessere Lösung zu finden.³

¹ *Bootstraps* sind die Schnürriemen, an denen man sich, ähnlich wie Münchhausen am eigenen Zopf, sprichwörtlich aus dem Sumpf zieht.

² Marx 1967, Bd. 1, S. 401ff.

³ Brief von Engelbart an Fred Dion, Rome Air Development Center, 25. Juli 1961, SUL/EC, 5-2.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

Der Top-down-Ansatz des *Bootstrapping* kann seinen Ursprung in der Kybernetik mit seiner Betonung von Funktionshierarchien, Wirkungsinterdependenzen und Regelmechanismen nicht verbergen. Dies dürfte einer der Gründe sein, warum er für den Systemingenieur Engelbart so naheliegend und attraktiv war.¹

Ganz aus technischer Sicht machte er sich auch Gedanken über die Praktikabilität des *Bootstrapping*. Die Rekursion hätte dazu führen können, daß man die Zyklen des Regelkreises hätte durchlaufen können, ohne jemals auf ein Teilprobleme zu stoßen, das sich mit einem technischen Hilfsmittel lösen ließe. Engelbart hielt dieses Risiko jedoch für vernachlässigbar, da durch die Verwendung eines interaktiven Computersystems mit Bildschirm und diversen Eingabeinstrumenten bereits ein erster Lösungsansatz existierte. Ausgehend von diesem »Anfangswert« sollten mittels *Bootstrapping* kontinuierliche Weiterentwicklungen erfolgen.²

Das Problem der positiven Rückkopplung innerhalb des *Bootstrapping*-Zyklus und der dadurch denkbaren Instabilität des Regelkreises war weniger leicht auszuräumen. Engelbart hoffte aber, daß dieses Problem nicht auftreten werde, wenn die neuentwickelten Werkzeuge und Methoden möglichst rasch in die eigene Arbeit integriert werden. So könnten schon in einem sehr frühen Stadium die Vor- und Nachteile getestet werden. Besonders wichtig war Engelbart auch die Berücksichtigung möglicher *Wechselwirkung* zwischen den einzelnen Komponenten.³ In seinen eigenen Worten besteht *Bootstrapping* (Abb. 35) in:

»Using the new tools and techniques as they evolve, for as much of [the developer's] work as possible – living with the stress of awkwardness, system bugs, new ways of doing things, etc., as part of his contribution to the research.«⁴

Eine erste Konsequenz des *Bootstrapping* war, daß sich Engelbart zunächst das Ziel setzte, Methoden zur effizienteren Entwicklung eines komplexen Computersystems zu entwickeln. Engelbart betrachtete also die eigene Arbeit bzw. die seiner Mitarbeiter als repräsentativ für intellektuelle Tätigkeiten im allgemeinen. Dieses Vorgehen barg freilich auch ein fundamentales Problem in sich. So könnte es dazu kommen, daß sich aufgrund der rekursiven Methodik die Entwicklungsgruppe so sehr mit sich selbst beschäftigte, daß sich ihre Arbeiten rasch von den

¹ Tatsächlich ist das *Bootstrapping* aus der rückblickenden Perspektive Engelbarts wichtigstes Anliegen. Nachdem viele seiner technischen Ziele heute erreicht scheinen, versuchte Engelbart, seine Methodik vor allem im unternehmensorganisatorischen Umfeld anzuwenden. Vgl. Engelbart 1991. Nicht zufällig hat er dem Institut, das er seit 1989 leitet, den Namen »Bootstrap Institute« gegeben.

² Brief von Engelbart an Fred Dion, Rome Air Development Center, 25. Juli 1961, SUL/EC, 5-2.

³ Engelbart 1962a, S. 7f.

⁴ Engelbart, D. C., »Study for the Development of Human Intellect Augmentation Techniques«, Quarterly Technical Letter Report No. 2 to NASA Langley Research Center, 15. August 1966, SUL/EC, 2-6.

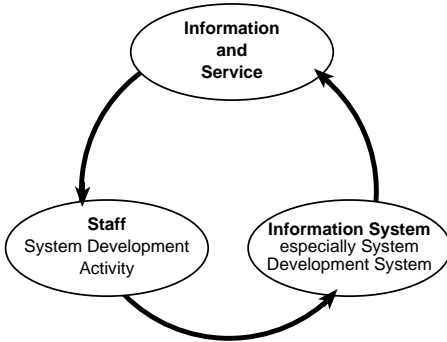


Abbildung 35. Regelkreis beim Bootstrapping

Anforderungen realer Anwendungen entfernte. Trotzdem war es ein sinnvoller Ausgangspunkt, um bei einem so ambitionierten Fernziel wie dem Intelligenzverstärker überhaupt in absehbarer Zeit erste vorzeigbare Ergebnisse zu erzielen – und damit auch die weitere Finanzierung des Projekts sicherstellen zu können.¹

Daneben besaß das *Bootstrapping* die zusätzliche Tugend, die Mitglieder der Forschungsgruppe bei ihrer Arbeit zu motivieren, da sie im Alltag kein in ferner Zukunft liegendes abstraktes Ziel verfolgten, sondern Systemkomponenten (Ein- und Ausgabegeräte, Software) entwickelten, die ihnen unmittelbar zugute kamen. Damit war auch sichergestellt, daß die Entwicklungsarbeit, zumindest für den Bereich der Programmierung bzw. Softwareentwicklung, anwendungsnah blieb. Schließlich stellte sie ein Regulativ gegen zu hochfliegende Pläne und gegen die Selbstüberschätzung des einzelnen Entwicklers dar.² Die Frage nach einem potentiellen späteren Benutzer des Intelligenzverstärkers war damit an dieser Stelle überflüssig. Er sollte, genau wie die Technik, durch *Bootstrapping* immer wieder neu entdeckt und definiert werden.³

Es galt allerdings nicht nur zu entscheiden, *wessen* Tätigkeit als erstes unterstützt werden sollte, sondern auch *welche* konkreten Tätigkeiten. Engelbart stellte hier pragmatisch fest, daß der größte Teil jeder intellektuellen Arbeit mit Hilfe von Text – sei es in Form von Programmtexten oder technischen Berichten – erfolge, der wiederum ein Abbild des menschlichen Denkens sei. Deshalb sei die Entwicklung von Computerprogrammen zur *Textverarbeitung* ein naheliegender erster Entwicklungsschritt, zumal diese wirkungsvollen Werkzeuge relativ einfach zu realisieren seien.⁴

¹ Engelbart 1962a, S. 8; Engelbart 1962c, S. 116ff.

² Engelbart 1962a, S. 8f.

³ Bardini and Horvath 1995, S. 49.

⁴ Engelbart 1962a, S. 13ff.; Memorandum von Engelbart an John H. Wensley, Tom L. Humphrey, Donald C. Lincicome, Elmer B. Shapiro und Roy C. Amara über »Introductory thoughts on a

5. Douglas Engelbart und der Computer als Intelligenzverstärker

Die weiteren Ausführungen werden zeigen, daß das *Bootstrapping* eine effektive Vorgehensweise darstellte, solange man sich im Rahmen des Time-Sharing-Paradigmas bewegte, das allerdings in dem Moment zum Hindernis für den weiteren Fortschritt wurde, als sich mit der Durchsetzung preiswerter integrierter Schaltkreise zu Beginn der siebziger Jahre die Rahmenbedingungen erheblich veränderten.

5.3.4 Kurzfristige und langfristige Ziele

Auf der Grundlage seines konzeptionellen Rahmens formulierte Engelbart im Oktober 1962 seine Pläne für die nächsten Jahre. In einem ersten Schritt sollten Untersuchungen über mögliche Formen der Mensch-Computer-Interaktion untersucht werden, durch die man auch die neo-whorfsche Hypothese erhärten könnte. Im zweiten Schritt sollten erste Werkzeuge entstehen, mit denen dann im dritten Schritt die Entwicklung des eigentlichen Intelligenzverstärkers, also eines Anwendersystems unterstützt werden sollte.¹ Gleichzeitig grenzte sich Engelbart deutlich gegen die Vorgehensweise der KI-Forschung im engeren Sinne ab, deren Ziele ja durchaus ähnlich waren. Insbesondere lehnte er es ab, die (analytische) Untersuchung der menschlichen Informationsverarbeitung von der Entwicklung des Intelligenzverstärkers abzukoppeln:

»By our view, we *do not* have to wait until we learn how the human mental processes work, we *do not* have to wait until we learn how to make computers more »intelligent« or bigger or faster, we *can* begin developing powerful and economically feasible augmentation systems on the basis of what we know and have.«²

Um die im ersten Schritt geplanten Experimente überhaupt durchführen zu können, mußte zunächst die notwendige Hardware beschafft werden. Engelbart benötigte einen leistungsfähigen Computer, der sich für den interaktiven Betrieb eignete. In einem weiteren Schritt mußten auf dem Markt befindliche Ein- und Ausgabegeräte getestet und neu konstruiert werden.³ Dafür mußte sich Engelbart erneut auf die Suche nach Geldgebern machen.

Transcription Central – a computer-aided, multiple-access, time-shared transcription system«, 29. Oktober 1963, SUL/EC, 17-6; English et al. 1965, S. x. Zur Bedeutung der Textverarbeitung als Werkzeug für den Computerbenutzer vergleiche auch Brooks 1975, S. 134.

¹ Engelbart 1962c, S. 115.

² Engelbart 1962c, S. 131, Hervorhebungen im Original.

³ Engelbart 1962c, S. 115f.; Memorandum von Engelbart an Reid Anderson, Jerre Noe und James Norton über »Individual's Information Handling«, 26. Juli 1960, SUL/EC, 17-7.

5.4 Geld von ARPA und NASA

Für Engelbart stand schon sehr früh fest, daß der Online-Betrieb für einen Intelligenzverstärker unbedingt notwendig sei und Time-Sharing die geeignete technische Realisierung. Schon kurz nach der Veröffentlichung von Lickliders »Man-Computer Symbiosis« erklärte Engelbart, daß er eine erstaunliche Nähe zwischen seinen und Lickliders Zielen sehe.¹ Und schon im Januar 1960 äußerte er in einem Memorandum die Idee, einen »time-shared clerk« zu realisieren.² Bis Ende 1961 entwickelte sich aus diesem Gedanken die Gewißheit, daß nur ein Time-Sharing-Computer die Möglichkeiten biete, die er sich vorstellte. Darunter fiel nicht nur der Dialogbetrieb – diesen hätte man auch durch den Einbenutzerbetrieb eines herkömmlichen Computers realisieren können – sondern auch die Möglichkeiten zur zentralen Speicherung aller Benutzerdokumente und zur computerunterstützten Kommunikation der Benutzer untereinander.³

Im Dezember 1961 waren die Überlegungen schließlich so weit gereift, daß Engelbart ernsthaft die Anschaffung eines Computers plante. Zu diesem Zweck fuhr er zusammen mit Roy Amara, dem Manager der SRI »System Engineering Division«, nach Washington, um diese Anschaffung mit Rowena Swanson vom Büro für wissenschaftliche Forschung der Luftwaffe zu diskutieren. Während der Reise wollten sie auch Kontakt mit J. C. R. Licklider und Edward Fredkin aufnehmen, um sich über die Time-Sharing-Experimente mit dem PDP-1 am MIT und bei BBN berichten zu lassen, da auch Engelbart die Anschaffung dieses Computers in Betracht zog.⁴

Obwohl es wegen der immer noch relativ geringen Förderung von Engelbarts Projekt zunächst nicht zur Anschaffung eines Computers kam, war auf diese Weise ein Kontakt zu Licklider, dem designierten IPTO-Direktor hergestellt. Bereits im August 1962 fanden Sondierungsgespräche zwischen dem SRI und der ARPA statt, in denen von Lickliders künftigem Stellvertreter das grundsätzliche Interesse bekundet wurde, Engelbarts Arbeit finanziell zu unterstützen. Konkretere Schritte konnten allerdings erst nach Lickliders offiziellem Dienstantritt im Oktober 1962 in die Wege geleitet werden.⁵ Schon Anfang Oktober nahm Engelbart persönlich

¹ Brief von Engelbart an Franklin S. Cooper, Haskins Laboratories, 25. Juli 1960, SUL/EC, 17-7.

² Memorandum von Engelbart an Reid Anderson, Jerre Noe und James Norton über »Individual's Information Handling«, 26. Juli 1960, SUL/EC, 17-7.

³ Engelbart 1962c, S. 3, 70.

⁴ Engelbart 1961b; Brief von Engelbart an J. C. R. Licklider, Bolt Beranek and Newman, 3. Januar 1962, SUL/EC, 6-12; Brief von Swanson an Edward Fredkin, Bolt Beranek and Newman, 19. Januar 1962, SUL/EC, 6-12; Brief von Engelbart an Rowena Swanson, Air Force Office of Scientific Research, 29. Januar 1962, SUL/EC, 6-12.

⁵ Memorandum von Roy Amara an Jerre Noe, Robert Wing, Cone und Douglas C. Engelbart über »Meeting with Mr. Floyd Van Hoosen, 2. August 1962«, 9. August 1962, SUL/EC, 5-2.

Kontakt mit Licklider in Washington auf und vereinbarte ein baldiges Treffen, bei dem er ihm seine Ziele, seine bisherigen Forschungsergebnisse und seine Pläne vorstellen wollte.¹ Engelbarts Beharrlichkeit wurde schließlich belohnt:

»When J. C. R. Licklider ... came from Cambridge to take over ARPA's newly formed Information Processing Techniques Office ... I was figuratively standing at the door with the ›Conceptual Framework‹ report and a proposal. There the unlucky fellow was, having advertised that ›man-computer symbiosis,‹ computer time-sharing, and man-computer interfaces were the new directions. How could he in reasonable consistency turn this down, even if it was way out there in Menlo Park?«²

Aufgrund der besonderen Organisationsstruktur des IPTO konnte Licklider schnell reagieren, so daß schon Anfang 1963 die Förderung durch die ARPA zugesagt wurde. Es wurden allerdings nicht die eher allgemeinen, konzeptionellen Arbeiten gefördert, die auch weiterhin vom Büro für wissenschaftliche Forschung finanziert wurden, sondern ein Projekt zur Entwicklung von »Programmierhilfen«, das besser in das Forschungsprogramm der ARPA paßte. Es ist nicht verwunderlich, daß selbst Licklider zu diesem Zeitpunkt nicht allzuviel Vertrauen in Engelbarts Pläne hatte. Engelbarts Visionen waren sogar für Licklider etwas zu grandios und seine Pläne zu nebulös, und so war die Förderung für ihn eine Art Glücksspiel: Sollte Engelbart keinen Erfolg haben, so wäre es die ganze Idee vermutlich ohnehin nicht wert, weiter verfolgt zu werden.³ Die Förderung hatte deshalb nur den vergleichsweise geringen Umfang von 195 000 \$ und war vorerst auf ein Jahr befristet. Damit schienen aber Engelbarts finanzielle Schwierigkeiten bei der Beschaffung eines Computers und der Rekrutierung von Personal gelöst zu sein.⁴ Im Vergleich zu den großen Time-Sharing-Projekten am MIT und bei der SDC, die Licklider mit Millionenetats versah, handelt es sich bei Engelbarts ARC allerdings um ein eher kleines ARPA-Projekt. Dies hatte nicht nur damit zu tun, daß Engelbarts Arbeitsprogramm nicht recht in die von der ARPA geförderten Forschungsschwerpunkte paßte, sondern vor allem mit Engelbarts Persönlichkeit und kommunikativen Fähigkeiten. Engelbart war nicht nur »ein schlechter Erklärer seiner eigenen Ideen«, sondern war auch nicht bereit, Kompromisse bei der Ausrichtung seiner Forschung einzugehen. Schon bald galt er deshalb als »Ein-

¹ Brief von Engelbart an J. C. R. Licklider, Advanced Research Projects Agency, 3. Oktober 1962, SUL/EC, 6-15.

² Engelbart 1988, S. 191.

³ Roy Amara in einem Interview mit Stan Augarten, zitiert in Augarten 1995.

⁴ Memorandum von Pat Conley an Torben Meisling über »Projects Relating to Man-Computer Program«, 21. Juli 1966, SUL/EC, 15-2; Brief von Engelbart an Robert W. Taylor, Advanced Research Projects Agency, 25. August 1967, SUL/EC, 1-29.

zelgänger, der seltsame Dinge tut.«¹ Engelbart spielte also innerhalb der elitären Gemeinschaft der ARPA-Wissenschaftler nicht nur finanziell, sondern auch inhaltlich nur eine marginale Rolle.

Auch innerhalb des SRI blieb Engelbarts Stellung trotz der gesicherten Finanzierung weiterhin problematisch. Selbst seine unmittelbaren Vorgesetzten begegneten ihm und seinen Ideen mit wenig Verständnis. Vor allem hielt man einen »Besessenen« nicht für den geeigneten Manager eines Projekts, das mit begrenzten Mitteln in relativ kurzer Zeit zu konkreten Ergebnissen führen sollte. Dies führte dazu, daß Roy Amara Engelbart *nicht* die Leitung des Projektes übertrug, sondern dem jungen John H. Wensley »eine Chance geben« wollte. Engelbart sollte nach der Auffassung seiner Vorgesetzten weiterhin Ideen entwickeln, Projektanträge schreiben und Gelder einwerben. Allerdings hatte Wensley kein Gespür für das Potential des interaktiven Computerbetriebs und die von Engelbart verfolgten Ziele. Auch hier muß man annehmen, daß es Engelbart nicht gelang, Wensley die für eine erfolgreiche Durchführung des Projekts notwendigen Ideen und Konzepte zu vermitteln.² Man kann sogar unterstellen, daß Engelbart einen solchen Versuch erst gar nicht unternommen hat, nachdem ihn das Management bei der Projektleitung übergangen hatte. Engelbart selbst erklärte rückblickend, die Situation sei für ihn so frustrierend gewesen, daß er in Erwägung zog, das SRI zu verlassen und zum MIT zu wechseln, wo er bereits als technischer Berater für das gerade anlaufende *Project MAC* tätig war. Engelbart übernahm die Leitung des Projekts erst wieder, als Licklider dies Ende 1963 zur expliziten Bedingung für die Vertragsverlängerung machte.³

Auch auf technischer Seite hatte das Projekt mit unerwarteten Problemen zu kämpfen, weil Lickliders Förderungszusage an Bedingungen geknüpft war. Anstatt einen eigenen Computer mit Time-Sharing-Betriebssystem anzuschaffen, sollte das SRI den AN/FSQ-32-Computer nutzen, der bei der *System Development Corporation* in Santa Monica stand und für den mit ARPA-Mitteln ein Time-Sharing-Betriebssystem entwickelt wurde (vgl. Abschnitt 4.3). Dabei stellte sich aber die Schwierigkeit ein, daß sich das Q-32 TSS während des Jahres 1963 noch in der Entwicklungsphase befand und nicht stabil lief. Wegen des fehlenden eigenen Rechners mußten Engelbart und sein Programmierer für jeden Programm-

¹ Robert Taylor in einem Interview mit Thierry Bardini, Palo Alto, 18. März 1993. – John Backus, der Entwickler der Programmiersprache FORTRAN, hat auf die Bedeutung von starken Persönlichkeiten in der Frühphase der Computerentwicklung hingewiesen. Er schreibt: »Programming in America of the 1950s had a vital frontier enthusiasm virtually untainted by either the scholarship or the stuffiness of academia (. . .) Recognition in the small programming fraternity was more likely to be accorded for a colorful personality, an extraordinary feat of coding, or the ability to hold a lot of liquor well than for intellectual insight.« Vgl. Backus 1980, S. 126f.

² Email von Dirk H. van Nouhuys an den Autor, 26. Januar 1998.

³ Engelbart 1986/87, Interview 2, Tape 2, Side A; Augarten 1995.

lauf nach Santa Monica fliegen. Außerdem konnte SDC ihnen zu dieser Zeit nur einen kleinen Bildschirm und eine Tastatur zur Verfügung stellen. Die Durchführung von Experimenten zur Mensch-Computer-Kommunikation, bei denen unterschiedliche Ein- und Ausgabeformen getestet werden sollten, war unter diesen Umständen sehr schwierig, und *Bootstrapping* war auf diese Weise überhaupt nicht zu realisieren.¹

Auch Licklider begann einzusehen, daß Engelbart auf diese Weise keine Forschung betreiben konnte und stimmte der Anschaffung eines CDC-160-A-Computers von der *Control Data Corporation* zu. Der CDC 160-A war ein 12-bit-Minicomputer mit 8 192 Worten Hauptspeicher, einer Zykluszeit von 6 μ s pro Instruktion und einem Trommelspeicher von 32 000 Worten. Damit war er zwar nicht so leistungsfähig wie der PDP-1, insbesondere erlaubte er keinen Time-Sharing-Betrieb, mit knapp 100 000 \$ war er aber auch deutlich billiger.

Zunächst wurde der CDC 160-A als Kommunikationsrechner verwendet, der über selbstkonstruierte Telekommunikationsgeräte mit dem Großrechner in Santa Monica verbunden wurde. Die Ein- und Ausgaben wurden am CDC 160 durchgeführt, während die Programme auf dem FSQ-32-Computer liefen.² Dieser Fernbetrieb erwies sich jedoch auch als problematisch. Die für den Dialogbetrieb benötigte fehlerfreie Datenübertragung in Echtzeit war über die zur Verfügung stehenden Telefonleitungen ohne aufwendige Verfahren zur Erkennung von Übertragungsfehlern nicht möglich.³ Außerdem beliefen sich die Kosten für die Verbindung zwischen Menlo Park und Santa Monica auf monatlich 1 500 \$. Aus diesen Gründen bat Engelbart im Frühjahr 1964 um die Änderung der Vertragsbedingungen. Dabei führte er weniger die praktischen Probleme seiner Arbeit an, als vielmehr den Wunsch, endlich sein eigenes System zur Verwaltung von bibliographischen Daten und zur Textverarbeitung entwerfen zu können, mit dem er dann *echte* Arbeiten durchführen wollte. Aus diesem Grund schlug er vor, alle weiteren Arbeiten nur noch auf dem lokalen Computer durchzuführen, dessen Rechenzeit frei verfügbar wäre. Nur so könne man die auch für die ARPA wichtigen Erfahrungen mit dem *Bootstrapping* sammeln. Da Licklider merkte, daß unter den bisherigen Umständen selbst Engelbarts bescheidenste Ziele nicht erreichbar wa-

¹ Engelbart 1986/87, Interview 2, Tape 2, Side A.

² Engelbart 1988, S. 192; Engelbart 1986/87, Interview 2, Tape 2, Side A; English et al. 1965, S. 88f.

³ »SRI-SDC Experience in Use of 2kc Data Transmission Lines from Menlo Park to Santa Monica, California«, Joint Report by SRI and SDC, 12. Mai 1964, SUL/EC, 15-10; Memorandum von Engelbart an John H. Wensley, Tom L. Humphrey, Donald C. Lincicome, Roy C. Amara über »Connection between Our Secondary Q-32 Programs and the TSS Exec«, 8. April 1964, SUL/EC, 15-2; Newell 1991a.

ren, wurde der Vertrag zwischen der ARPA und dem SRI schließlich in Engelbarts Sinne geändert.¹

Während die Förderung durch die ARPA lediglich das Fortbestehen des Projekts gesichert hatte, kamen 1964 auch die *Electronic Systems Division* (ESD) der amerikanischen Luftwaffe und vor allem die NASA als weitere Sponsoren hinzu. Engelbart hatte bereits im März 1963 Kontakt mit Robert W. Taylor (* 1932) aufgenommen, der zu dieser Zeit beim NASA *Office of Advanced Research and Technology* arbeitete. Obwohl Taylor an der University of Texas studiert hatte, also nicht der Bostoner Wissenschaftlergemeinschaft angehörte, wurde er im Laufe der sechziger Jahre zu einer Schlüsselfigur bei der Entwicklung interaktiver Computersysteme, zunächst bei der NASA, dann als Vizedirektor bzw. Direktor des IPTO und schließlich ab 1970 am Xerox Palo Alto Research Center (PARC) (vgl. Kapitel 6). 1963 sollte Taylor für die NASA den Aufbau eines eigenen Forschungszentrums für Elektronik vorbereiten und suchte Kontakt zu Projekten mit vergleichbaren Zielen. Die von Taylor zugesagte Förderung begann nach einiger Verzögerung im Juni 1964, weil es ihm einige Mühe bereitete, Engelbarts Projekt im Forschungsschwerpunkt Flugsimulatoren unterzubringen.²

Nachdem Engelbart im Mai 1965 auf der halbjährlichen ARPA-Tagung eine Vorführung des ersten Prototypen seines Offline-Text-Systems (vgl. Abs. 5.7) gegeben hatte, die Robert Taylor, damals IPTO-Vizedirektor, sehr beeindruckte, wurden die Fördermittel weiter aufgestockt und gebündelt.³ So sollten die Reibungsverluste vermieden werden, die durch die Beteiligung einer Vielzahl von Institutionen auftraten, zumal diese ohne Ausnahme dem amerikanischen Verteidigungsministerium unterstanden. Seit 1966 erhielt das ARC seine finanziellen Mittel nur noch über die NASA und das Rome Air Development Center, das 1967 doch noch zu Engelbarts Geldgebern hinzukam.⁴ Damit besaß Engelbarts Labor Mitte der sechziger Jahre endlich eine solide finanzielle Basis.

Mit der zunehmenden Finanzausstattung begann auch das Wachstum des ARC. Nachdem Engelbart bislang Mitarbeiter des Computer Techniques Laboratory ausgeliehen hatte, stellte er ab 1964 eigenes Personal ein. Zu den ersten Mitarbeitern Engelbarts gehörte auch William K. English (* 1929), der bereits

¹ Brief von Engelbart an J. C. R. Licklider, Advanced Research Projects Agency, 4. Mai 1964, SUL/EC, 15-2; Memorandum von Pat Conley an Torben Meisling über »Projects Relating to Man-Computer Program«, 21. Juli 1966, SUL/EC, 15-2.

² D. C. Engelbart, Report über »Meeting on 13 February with NASA at OART Office, 1512 N St., Washington«, 7. März 1963, SUL/EC, 6-15; Augarten 1995.

³ Engelbart 1986/87, Interview 3, Tape 1, Side A; Engelbart 1986.

⁴ Memorandum von Engelbart an Torben Meisling, Jerre Noe und William K. English über »Telephone Contact with Robert Taylor, Information Processing Techniques, Office of Secretary of Defense, ARPA, 8/4/65«, 5. August 1965, SUL/EC, 5-2. – Weitere Details über die Quellen und die Höhe der finanziellen Unterstützung sind im Anhang zu finden.

seit 1958 am SRI beschäftigt war. English war in den folgenden Jahren als Engelbarts Stellvertreter für die Hardware verantwortlich und übernahm die Leitung der durch die NASA geförderten Projekte. Bis 1969 stieg die Mitarbeiterzahl des ARC auf knapp 20 an, dabei rekrutierte sich der Großteil des Personals aus Absolventen der nahegelegenen Universitäten Stanford und Berkeley, die teilweise bereits vor ihrem Abschluß am ARC beschäftigt waren.¹

Schließlich ermöglichte der Vertrag mit der NASA den Umstieg auf einen CDC-3100-Computer im November 1965. Der CDC 160-A war zwar unter den Umständen von 1963 ein annehmbarer Kompromiß gewesen, hatte aber auch gravierende Schwächen. Trotz der Betriebsweise als Einbenutzercomputer war er wegen der vergleichsweise geringen Leistung und eines fehlenden Plattenspeichers letztlich nicht als Basis für ein interaktives Online-System geeignet. Obwohl auch mit dem CDC 3100² kein Time-Sharing möglich war, eignete er sich zumindest für die Durchführung erster Versuche zur Mensch-Computer-Interaktion.³ Das Problem der geeigneten Hardware wurde erst im Herbst 1967 gelöst, als das ARC mit finanzieller Unterstützung der NASA einen der ersten kommerziellen Time-Sharing-Computer, den SDS 940 von *Scientific Data Systems* anschaffte. Mit diesem Rechner konnte endlich gewährleistet werden, daß jeder Mitarbeiter des ARC täglich mehrere Stunden im unmittelbaren Dialog mit dem Computer arbeiten konnte, so wie es Engelbarts *Bootstrapping*-Methodik vorsah.

5.5 Die Gestaltung der Mensch-Computer-Schnittstelle

Bereits 1963 begann Engelbart mit der Untersuchung von Ein- und Ausgabegeräten für die interaktive Mensch-Computer-Kommunikation, der Realisierung eines ersten Systems zur computerunterstützten Textverarbeitung sowie der Analyse der Arbeit mit diesem System.⁴

Dabei interessierte sich Engelbart vor allem für die Wirkungszusammenhänge eines komplexen Mensch-Maschine-Systems. Engelbart ging im Sinne der Systemtechnik davon aus, daß jede Änderung eines Systemelements zu einem Wiederhall, d. h. zu Auswirkungen auf andere Systemelemente führt. Wegen dieser

¹ Engelbart 1988, S. 193; Engelbart 1986/87, Interview 3, Tape 2, Side B; Brief von Engelbart an Major Seymour Jeffrey, Electronic Systems Division, 25. Mai 1965, SUL/EC, 15-2.

² Der CDC 3100 hatte einen Hauptspeicher mit 16 384 24-bit-Worten und einer Zugriffszeit von 1,75 μ s, einen IBM 1311 Plattenspeicher mit einer Kapazität von 2 Mio. Zeichen, drei Magnetbandlaufwerke sowie Ein- und Ausgabe über Lochkarten und Lochstreifen. Vgl. English et al. 1965, S. 60.

³ Brief von Engelbart an Major Seymour Jeffrey, Electronic Systems Division, 25. Mai 1965, SUL/EC, 15-2.

⁴ Brief von Engelbart an Harold Wooster, Air Force Office of Scientific Research, 29. November 1963, SUL/EC, 6-12.

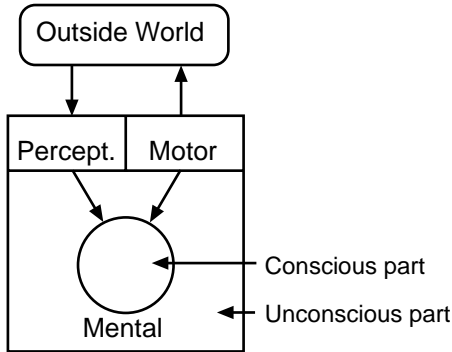


Abbildung 36. Engelbarts einfaches Modell der menschlichen Informationsverarbeitung. Originalbildunterschrift: »We build on basic human capabilities, absorbing the rest from out cultures. Human societies had to develop complex systems of languages, custom, tools and methods so that the human's tiny grasp could cope with large tasks and problems.«

Interdependenzen müsse die Arbeit auf *allen* Ebenen gleichzeitig begonnen werden. Dabei sei es sinnvoller, mit den vorhandenen technisch-wissenschaftlichen Möglichkeiten zu beginnen, ohne Innovationen von außerhalb abzuwarten, etwa im Bereich der kognitiven Psychologie, der Künstlichen Intelligenz oder bei der Entwicklung von Programmiersprachen.¹

Engelbart war ohnehin der Meinung, die Arbeit an einem Intelligenzverstärker sollte mit einer systematischen Untersuchung der sogenannten *grundlegenden menschlichen Fähigkeiten* beginnen (Abb. 36).² Für ihn war das sensomotorische System die Schnittstelle zwischen den bewußten und unbewußten kognitiven Vorgängen auf der einen und der Außenwelt auf der anderen Seite. In der Sichtweise der kognitiven Psychologie ist die Perzeption damit die Grundlage jeder menschlichen Tätigkeit.³

Engelbart und seine Mitarbeiter begannen deswegen ihre praktischen Arbeiten mit der Analyse des sensomotorischen Systems und dem Entwurf von Verfahren und technischen Geräten, mit denen die sensomotorischen Fähigkeiten des Menschen auch beim Umgang mit dem Computer möglichst optimal genutzt werden konnten. Sie entwickelten eine Reihe von technischen Ein- und Ausgabegeräten, die die große Bedeutung der Sensomotorik auf der Seite des Computers widerspiegeln. Dabei handelt es sich um die Bildschirmausgabe zur Unterstützung der visuellen Fähigkeiten sowie die Maus und die Einhandtastatur zur Unterstützung der manuellen Fähigkeiten.

Schon seit 1960 hatte Engelbart experimentelle empirische Untersuchungen in diesem Bereich durchführen wollen. Dabei dachte er zunächst an die unvoreinge-

¹ Engelbart 1965, S. 23.

² Engelbart 1962c, S. 11f., 21; Engelbart 1963, S. 5, 11.

³ Diese Anfang der sechziger Jahre relativ neue Sichtweise gehört heute zu den Grundüberzeugungen einer bestimmten Richtung der Kognitionswissenschaften. Vgl. etwa Varela 1992, S. 335.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

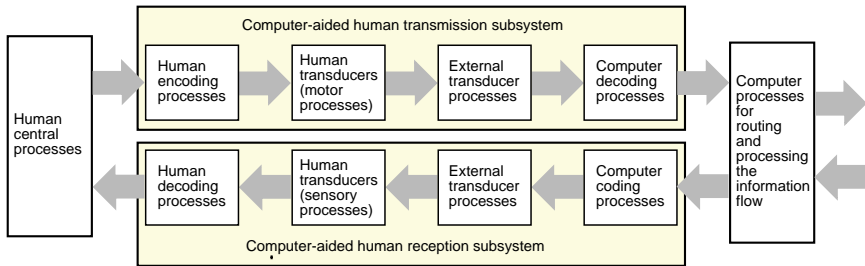


Abbildung 37. Engelbarts Modell der Mensch-Computer-Kommunikation (1964)

nommene Untersuchung aller Möglichkeiten zur Datenein- und -ausgabe.¹ Dieses Vorhaben entwickelte sich rasch zu einer Projektidee über das automatische Erlernen psychomotorischer Fertigkeiten mit Hilfe des Computers, für die Engelbart zunächst erfolglos Fördergelder beantragte. Erst 1963 konnte sich Engelbart der theoretischen und praktischen Untersuchung dieses Themenkomplexes widmen.²

Dabei entstand zunächst das in Abbildung 37 dargestellte Modell der Mensch-Computer Kommunikation, in dem die bislang abstrakte Schnittstelle zwischen Mensch und Maschine so weit differenziert wurde, daß sich die einzelnen Elemente nach den im konzeptionellen Rahmen definierten Dimensionen klassifizieren und analysieren lassen. Engelbart identifizierte Kodierungs- und Übertragungsprozesse, die er entweder ausschließlich dem Menschen oder dem Computer zuordnen konnte. Da die Mensch-Computer-Kommunikation demnach eine Kette von Prozessen darstellt, war ihre Effizienz stets von der Leistung des schwächsten Gliedes abhängig. Bei der Optimierung mußte also darauf geachtet werden, möglichst alle Elemente der Kette gleich leistungsfähig zu gestalten, wobei berücksichtigt werden muß, daß nicht alle Prozesse im gleichen Maße zu beeinflussen sind.³ Damit konkretisierte sich Engelbarts erstes Teilziel. Im April 1963 schrieb er in einem Brief an Robert Taylor:

¹ Memorandum von Engelbart an Reid Anderson, Jerre Noe, James Norton und Robert Wing über »Individual's Information Handling«, 26. Juli 1960, SUL/EC, 17-7.

² Engelbart, D. C., »Rough-Form Think Piece Regarding Research Possibilities at SRI in Areas of Man-Machine Communication Means, and Automated Psycho-Motor Skill Training«, 3. März 1961, SU/EC, 16-6; Engelbart, D. C., »Automated Psycho-Motor Skill Training«, 13. Dezember 1961, SUL/EC, 4-9; Engelbart 1962b.

³ Engelbart 1965, S. 4ff.; Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15.

»... the problem is to develop interface signal forms and codes, and real-time display and feedback aids that allow the quickest and easiest communication of... information (consistent with system needs and possibilities, of course).«¹

5.5.1 Dateneingabe: Möglichkeiten und Widerstände

Um die Aufgabe nicht weiter zu komplizieren, verzichtete man zunächst auf die Konstruktion eines optimalen Eingabegeräts. Man versuchte stattdessen, die Eigenschaften bereits bekannter Verfahren zu analysieren. Bei der Betrachtung dieser Verfahren ergaben sich für Engelbart schnell Hinweise, welche Formen der Dateneingabe erfolgsversprechend waren und welche weniger.²

Die natürlichste Form der Kommunikation ist die gesprochene Sprache, die nicht eigens erlernt werden muß und eine relativ hohe Datenübertragungsrate ermöglicht. Wegen ihrer komplexen Signalform und hochgradig redundanten Kodierung existierten allerdings keine brauchbaren Verfahren zur Erkennung gesprochener Sprache durch den Computer.³

Für die Aufzeichnung von geschriebener Sprache existierte eine Vielzahl von technischen Verfahren. Das verbreitetste dieser Verfahren ist die *Schreibmaschinentastatur*, die bereits seit Mitte der fünfziger Jahre auch als Eingabegerät für den Computer verwendet wurde. Selbst ein ungeübter Benutzer konnte mit einer solchen Tastatur nach kurzer Übung eine mäßig schnelle Dateneingabe erreichen, unter anderem, da die Kodierung (jede Taste steht für ein Zeichen) sehr einfach ist. Es gab aber auch Gründe, die gegen die Verwendung der Schreibmaschinentastatur sprachen. Dazu gehörten vor allem die beidhändige Bedienung und die große Bauform. Eine kleinere und weitaus schnellere Tastatur besaß die *Stenografiermaschine*, wie sie vor allem an amerikanischen Gerichten zur Protokollierung verwendet wird. Die Kodierung ist bei dieser Tastatur wesentlich komplexer als bei der Schreibmaschinentastatur, jede Kombination der 23 Tasten kann für ein bestimmtes Symbol, ein Zeichen oder eine Silbe stehen. Dies hat zur Folge, daß erheblich höhere Datenraten erreicht werden können, allerdings erst nach einer langen Trainingszeit. Da die Dekodierung von Text erhebliche Probleme bereitete, schien sich auch die unmodifizierte Stenotastatur nicht als Eingabegerät für den Computer zu eignen. Ein verbreitetes und sehr einfaches Eingabegerät war der *Morsetaster*, der allerdings schon wegen der sehr geringen Datenrate ausschied.

¹ Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15.

² Engelbart 1965, S. 7; Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15.

³ Engelbart 1965, S. 8; Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

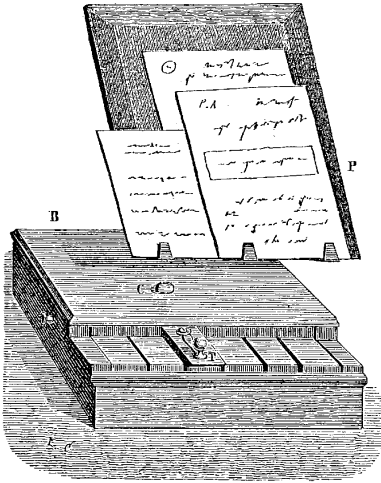


Abbildung 38. Emile Baudots Manipulator von 1874

Nur der Vollständigkeit halber führte Engelbart auch die Möglichkeit der *handschriftlichen Eingabe* an, deren Signal und Kodierung allerdings ebenso komplex ist wie die der gesprochenen Sprache und nur eine sehr langsame Dateneingabe ermöglicht. Reizvoll erschien ihm allerdings die sehr komfortable und intuitive Möglichkeit zum Einsetzen und Löschen bei handschriftlicher Eingabe.¹

Obwohl Engelbart mit keiner dieser Eingabetekniken wirklich zufrieden war, ergab seine Analyse Ansatzpunkte für die weitere Entwicklung. Zum einen war die Schreibmaschinentastatur so allgegenwärtig, daß auch er nicht auf ihre Verwendung verzichten konnte. Zum anderen kam Engelbart zu der Überzeugung, daß eine *einhandig* zu bedienende Tastatur für seine Zwecke vorteilhaft sein könnte. Sie sollte allerdings, im Gegensatz zur Stenografentastatur, nur fünf Tasten besitzen, über die sich auch nur die normalen alphanumerischen Zeichen eingeben ließen. So konnten gleichzeitig geringe Abmessungen und eine hohe Datenrate erreicht werden, ohne daß sich das Problem der Dekodierung ergab. Die einzige Voraussetzung war, daß die Benutzung einer solchen Einhandtastatur einer gewissen Übung bedurfte. Engelbart sah darin allerdings keine Einschränkung, da nach der Lernphase die Effektivität deutlich höher war.²

¹ Engelbart 1965, S. 8ff.; Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15.

² Ratz and Ritchie 1961; Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15 – Dies ist ein typisches Beispiel dafür, was Engelbart unter der Koevolution des H-LAM/T-Systems verstand. Nicht die intuitive Bedienbarkeit war das Ziel seiner Bemühungen, die Verbesserung der Systemeffektivität konnte auch durch das Training des Menschen mit einem neuen Artefakt erfolgen. Vgl. dazu Engelbart 1988, S. 199ff.

Im Rahmen des AFOSR-Projekts wollte Engelbart eine solche Einhandtastatur systematisch untersuchen. Dabei sollte z. B. die Häufigkeit von Zeichen und Zeichenkombinationen sowie die menschliche Merkfähigkeit für Zeichengruppen untersucht und die Tastaturbelegung, d. h. die Kodierung entsprechend optimiert werden.¹ Es ergab sich jedoch ein unerwartetes Hindernis, das das latente Spannungsverhältnis zwischen Engelbart und seinen militärischen Geldgebern offenbart. Harold Wooster vom AFOSR wand nämlich ein, daß Engelbarts Einhandtastatur mit fünf Tasten alles andere als neu sei. Schon 1874 hatte Emile Baudot für seinen Schnelltelegraphen einen sogenannten Manipulator zur Zeicheneingabe entwickelt, der auffallende Ähnlichkeit mit Engelbarts geplanter Einhandtastatur besaß (Abb. 38 und 39). Wooster gab zu bedenken, daß die Telegrafeningenieure des vergangenen Jahrhunderts ähnliche Versuche durchgeführt hätten, und der Baudotsche Manipulators sich wohl aus technischen Gründen nicht habe durchsetzen können. Er sei aber nicht gewillt, »die Neuerfindung des Hochrades mit staatlichen Geldern« zu fördern.² Engelbart konnte Wooster nur mit Mühe und mit Hinweis auf Lickliders persönliche Unterstützung davon überzeugen, daß es sich hier um eine *Systemfrage* handle und das Verschwinden des Baudotschen Manipulators aus der Telegrafentechnik nicht bedeutete, daß deshalb eine ähnliche Einhandtastatur als Komponente eines interaktiven Computer-Systems ungeeignet sei.³ Engelbart und seine Mitarbeiter kamen sogar zu der Überzeugung, daß man alle scheinbar vertrauten technischen Möglichkeiten zur Dateneingabe neu analysieren und gegebenenfalls modifizieren müsse.⁴ Weil es aber mit Sicherheit nicht *das* optimale Eingabegerät gebe, müßten für die Mensch-Computer-Kommunikation mehrere Eingabegeräte parallel zur Verfügung stehen.⁵

Neben der eher konventionellen Einhandtastatur schlug Engelbart in seinem Projektantrag an die NASA einige Eingabetechniken vor, die weitaus exotischer und weniger einfach zu realisieren waren. So regte er an, die Eingaberate durch den Einsatz einer Zehnfingertastatur zu erhöhen. Der Benutzer einer solchen Tastatur sollte die vollständige Kodierungsaufgabe übernehmen und seine Eingaben unmittelbar in einem computerlesbaren 10-bit-Code machen, mit dem sich nicht nur die alphanumerischen Zeichen, sondern insgesamt 1023 Symbole – alphanumerische Zeichen, viel verwendete Zeichenfolgen und ganze Befehlssequenzen – kodieren ließen. Obwohl sich Engelbart bewußt war, welchen Trainingsaufwand

¹ Engelbart 1962c, passim, Engelbart 1962b, S. 7.

² Brief von Harold Wooster, Air Force Office of Scientific Research an Engelbart, 18. Oktober 1962, SUL/EC, 6-12.

³ Brief von Engelbart an Harold Wooster, Air Force Office of Scientific Research, 15. November 1962, SUL/EC, 6-12; Licklider 1988b, S. 119.

⁴ Lincicome, Donald L., »Man-Display Communication«, 1. Februar 1963, SUL/EC, 5-17.

⁵ Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

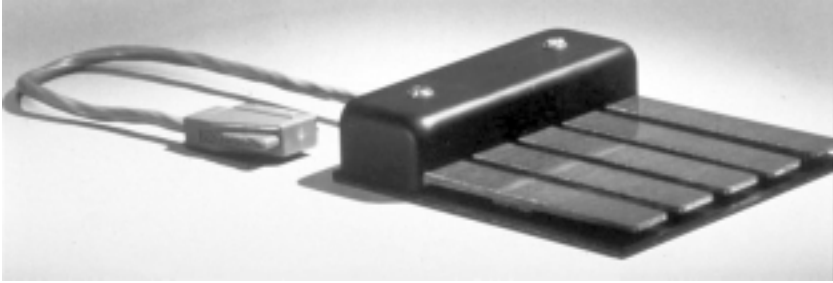


Abbildung 39. Einhandtastatur (Five-finger keyset)

dies für einen Benutzer bedeuten würde, hielt er die Untersuchung von Trainingsaufwand und erreichbarer Geschwindigkeit durchaus für lohnend. Ursache für diesen Vorschlag war die schnelle Gewöhnung an die Einhandtastatur, die er bei sich selbst und bei einer Schreibkraft des ARC festgestellt hatte.¹

Ein anderer Vorschlag hatte die weitere Verkleinerung des Eingabegeräts zum Ziel. Engelbart schlug dazu die Verwendung eines »Datenhandschuhs« vor, in dessen Fingerspitzen druckempfindliche Sensoren eingebaut sein sollten, die die Funktion der Tasten bei der Einhandtastatur übernehmen. Der Benutzer könnte dann seine Eingaben durch das Aufdrücken der Finger auf einer beliebigen glatten Oberfläche vornehmen. Zusätzlich – so Engelbart – könnte man den Zeigefinger mit einer Photozelle ausstatten und als Ersatz für den Lichtgriffel verwenden.² Engelbarts Ideen mögen auch die Zeitgenossen fasziniert haben. Ihre Realisierung war nach den Problemen, die Engelbart bereits bei der Verwendung der Einhandtastatur gehabt hatte, im Rahmen der militärischen Forschungsförderung nicht möglich.

Die Einhandtastatur stellte sich freilich in ausführlichen Experimenten sowie während der täglichen Arbeit als hilfreiches Eingabegerät heraus, das der herkömmlichen Tastatur in puncto Eingabegeschwindigkeit zwar unterlegen war, aber bei der Eingabe von kurzen Befehlssequenzen am Bildschirm von Vorteil war.³ Obwohl sie sich während der sechziger Jahre am ARC neben der Schreibmaschinentastatur als Haupteingabegerät etablierte, konnte sich die Einhandtastatur nicht gegen die Übermacht der herkömmlichen Schreibmaschinentastatur durchsetzen. Versuche des Xerox PARC und von IBM zur Einführung von er-

¹ Engelbart 1965, S. 11ff.; Brief von Engelbart an J. C. R. Licklider, Advanced Research Projects Agency, 4. Mai 1964, SUL/EC, 15-2.

² Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15.

³ Engelbart and Sorensen 1965; Engelbart 1969, S. 66f.

gonomisch optimierten Einhandtastaturen während der siebziger Jahre scheiterten ebenso wie Engelbarts Anstrengungen in den sechziger Jahren.¹ Grund für das Beharrungsvermögen dieses historisch gewachsenen Standards war schlicht der wirtschaftliche Wert, den die weitverbreitete Fähigkeit zur Bedienung einer Schreibmaschinentastatur darstellte.²

5.5.2 Revolution oder Verschwendung: Arbeiten am Bildschirm

Weit weniger kontrovers lief die Diskussion über das geeignete Ausgabemedium ab. Für Engelbart stand von Anfang an fest, daß als Ausgabegerät für sein Computersystem nur ein Bildschirm in Frage kam. Damit teilte er die Vorstellung der meisten anderen Verfechter des interaktiven Computerbetriebs, so wie es in Kapitel 3 und 4 dargestellt wurde. Engelbart hatte aber andere Ideen über die Einbettung des Bildschirms in das Computersystem. Bislang wurde der Bildschirm zur Dateneingabe während des Programmlaufs (z. B. bei Buchungssystemen), zur interaktiven Fehlersuche oder zur grafischen Ausgabe von Ergebnissen verwendet. Alle sonstigen Tätigkeiten, vor allem die Eingabe von Text, insbesondere von Programmtext, wurde bis dahin *nicht* am Bildschirm durchgeführt; dafür gab es Eingabegeräte wie den Flexowriter, mit dem man Daten auf einen Lochstreifen schreiben konnte, der dann in den Computer eingelesen wurde.

Diese Betriebsweise des Computers hatte rein ökonomische Gründe, schließlich war Rechenzeit eine knappe und deshalb teure Ressource.³ Im Zusammenhang mit der Nutzung des CDC 160-A im Einbenutzerbetrieb setzte sich Engelbart 1964 auch über dieses ökonomische Gebot hinweg und entschied sich für die »Man-Display Communication«, bei der *alle* Tätigkeiten am Bildschirm durchgeführt werden sollten.⁴

Hintergrund dieser Entscheidung war die Tatsache, daß die Manipulation von Symbolen mit unmittelbarem Feedback den Menschen von der Notwendigkeit befreite, die Struktur eines Programms oder Dokuments vollständig im Kopf zu haben. Die Online-Arbeit senkte somit die geistige Beanspruchung und ermöglichte

¹ Rochester et al. 1978; Thacker et al. 1982, S. 552.

² David 1985; Campbell-Kelly and Aspray 1996, S. 30ff. Eine überschlägige Rechnung mag dies veranschaulichen: Wenn man vorsichtig annimmt, daß es 100 Millionen Menschen gibt, die diese Fähigkeit besitzen, 100 Stunden für den Erwerb der Fähigkeit benötigt werden und jede Stunde 10 \$ kostet, dann hat die herkömmliche Schreibmaschinentastatur einen Wert von 100 Milliarden Dollar. Vgl. Email von Ivan E. Sutherland an den Autor, 18. September 1997.

³ Friedewald 1998. Tatsächlich war es an einigen Universitätsrechenzentren noch bis Anfang der achtziger Jahre explizit verboten, knappe und teure Rechenzeit für das Editieren von Texten am Bildschirm zu »verschwenden«.

⁴ Lincicome, Donald L., »Man-Display Communication«, 1. Februar 1963, SUL/EC, 5-1; Engelbart 1986.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

effektiveres Arbeiten. Engelbart dachte zwar zunächst an rein quantitative Verbesserungen, d. h. eine schnellere Ausführung grundlegender Tätigkeiten, hoffte aber, daß die so freiwerdenden Kapazitäten auch zu einer qualitativen Verbesserung der Arbeit führen würden.¹

Damit stellte sich Engelbart die Frage, wie er mit seinen begrenzten finanziellen Mitteln einen Bildschirm beschaffen konnte, der den Anforderungen des Online-Betriebs in bezug auf Bildwiederholrate und Auflösung gerecht wurde, zumal er bereits 1963 angeregt hatte, daß das Ausgabemedium neben Text auch Grafiken und Diagramme darstellen können sollte.² Dabei muß berücksichtigt werden, daß Bildschirm-Terminals zu Beginn der sechziger Jahre neben der Kathodenstrahlröhre aus einem Zeichengenerator und diverser Schnittstellenelektronik bestand. Da sie noch nicht zur Standardausrüstung von Computern gehörten, boten Computerhersteller, wenn überhaupt, nur wenige Modelle an. Engelbart mußte deshalb das benötigte Terminal aus Komponenten verschiedener Hersteller zusammenstellen und durch Eigenentwicklungen ergänzen. Das letztlich verwendete Terminal (Abb. 40) war dann mit etwa 80 000 \$ fast so teuer wie der Computer selbst. Aus diesem Grund wurden während der ersten Jahre im ARC auch weiterhin Fernschreiber verwendet.³

Schließlich erlaubte die Arbeit direkt am Terminal auch die Eingabe von Kommandos über Tastaturkürzel. Diese Form der Eingabe war zuvor als nicht praktikabel verworfen worden, weil dabei angeblich zu viele Fehler auftraten. Da der Benutzer nun über den Bildschirm eine unmittelbare Rückmeldung über seine Eingaben und die Folgen der von ihm initiierten Befehle erhielt, ging die Fehlerrate drastisch zurück, so daß auch die Bearbeitung der eingegebenen Texte am Computer deutlich beschleunigt wurde.⁴

5.5.3 Grafische Eingabegeräte: Von Mäusen und Käfern

Ein wichtiger Bestandteil der Texteingabe und -bearbeitung im Dialog mit dem Computer war für Engelbart die Möglichkeit, einzelne Buchstaben und Wörter

¹ Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15. Zum Konzept der begrenzten mentalen Ressourcen bei Mehrfachbelastungen vgl. Wessells 1990, S. 96ff.; Card et al. 1983, S. 387ff.

² Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15.

³ Memorandum von Engelbart an Tom L. Humphrey, John M. Yarborough, DCC, C. Bruce Clark, Jack Goldberg, Milton B. Adams, Jerre D. Noe, DAB, Donald C. Lincicome, Elmer B. Shapiro, John H. Wensley, Robert M. Madwick über » »Cheap Station« work at CTL«, 12. November 1963, SUL/EC, 17-6; Memorandum von Milton B. Adams an Douglas C. Engelbart über »CTL Help on 004531-612«, 13. November 1963, SUL/EC, 17-6; Engelbart 1986.

⁴ Engelbart 1986/87, Interview 2, Tape 2, Side A.



Abbildung 40. Bildschirmarbeitsplatz, an dem 1963/64 die Experimente mit unterschiedlichen Eingabegeräten durchgeführt wurden. Rechts im Bild ist das Grafacon zu sehen.

oder ganze Sätze und Absätze zu löschen, zu verschieben, einzurücken oder sonstwie zu bearbeiten. Zu diesem Zweck sollte eine Technik zur Auswahl von Bildschirmobjekten gefunden werden, die so einfach und effektiv sein sollte, daß sie den Arbeitsfluß des Menschen nicht wesentlich beeinträchtigte. Deshalb führten Engelbart und seine Mitarbeiter 1963/64 Versuche durch, um ein für diese Aufgabe geeignetes Eingabegerät zu identifizieren. Dabei betraten sie auch methodisch Neuland, da die Experimente nicht nur am Rechner durchgeführt, sondern auch vom Rechner überwacht und ausgewertet wurden.¹

Das System, das für die Experimente verwendet wurde (Abb. 40), spiegelte zunächst weniger Engelbarts große Vision als vielmehr die limitierten technischen Möglichkeiten wieder. Der CDC-160-A-Computer wurde mit einem experimentellen Zeichengenerator ausgerüstet, mit dem sich auf einem kreisrunden Bildschirm von 16 Zoll Durchmesser 16 Zeilen mit je 64 Zeichen darstellen ließen. Der Zeichensatz umfaßte dabei lediglich 36 alphanumerische Zeichen, darunter keine Kleinbuchstaben. Angeschlossen war außerdem eine Schreibmaschinenta-

¹ English et al. 1965, S. 1, 40f.

statur, die nach Spezifikationen des ARC speziell angefertigt wurde, ferner eine Einhandtastatur und eine Schalttafel.¹

An dieses Computersystem wurden unterschiedliche *grafische* Eingabegeräte angeschlossen, d. h. solche Geräte, die nicht zur Eingabe alphanumerischer Zeichen dienten. Dazu gehörten damals bereits etablierte und vielverwendete Geräte wie der Lichtgriffel, der Steuerknüppel (engl. *joystick*) oder die Rollkugel. Darüber hinaus interessierte sich Engelbart auch für die Untersuchung neuer Möglichkeiten zur Dateneingabe, z. B. durch Augen-, Nasen- oder Fußsteuerung.²

Tatsächlich für die Experimente verwendet wurden ein Steuerknüppel, ein Lichtgriffel und das *Grafacon* von der *Data Equipment Corp.* sowie zwei selbst-konstruierte Eingabegeräte. Ursprünglich hatten Engelbart und English geplant, auch das von der RAND Corporation entwickelte Grafiktablett³ zu testen. Nach Engelbarts Auffassung war das RAND-Grafiktablett 1963 das am besten für seine Erfordernisse geeignete Eingabegerät, da es sich nach seiner eigenen Erfahrung einfacher und genauer zur Auswahl von Objekten am Bildschirm eignete als jeder Lichtgriffel, den er bislang verwendet hatte.⁴ Da die Grafiktablets noch nicht kommerziell erhältlich waren⁵, versuchte Engelbart, einen der Prototypen für seine Versuche auszuleihen. Da die RAND Corporation kein Grafiktablett entbehren konnte, machte sich Engelbart an die Konstruktion eines ähnlichen Eingabegeräts.⁶ Engelbart vergleicht dies heute gern mit der Periodentafel der Elemente. So wie sich Elemente einer Hauptgruppe in wesentlichen Eigenschaften gleichen, habe er versucht, ein Eingabegerät zu finden, das der gleichen »Hauptgruppe« angehört wie das RAND-Grafiktablett.⁷

Das Resultat dieser Suche waren die Maus und ein mit dem Knie zu bedienendes Eingabeinstrument. Bei beiden Geräten handelte es sich um indirekte Eingabegeräte. Während der Benutzer mit dem Lichtgriffel die gewünschte Position direkt auf dem Bildschirm auswählte, mußte er mit einem indirekten Eingabegerät den auf dem Bildschirm angezeigten Positionsindikator auf die gewünschte Position bewegen. Dabei konzentrierte er sich im besten Fall allein auf den Positionsindikator, während die Bedienung des Eingabegeräts mit der Hand quasi

¹ English et al. 1965, S. 95ff.

² Brief von Engelbart an Robert W. Taylor, NASA Office of Advanced Research and Technology, 5. April 1963, SUL/EC, 6-15.

³ Davis and Ellis 1964.

⁴ Memorandum von Engelbart an J. H. Wensley, Tom L. Humphrey, Donald C. Lincicome, FKT, Charles P. Bourne, LJC, JJB, SDM über »Report on RAND Tablet«, 10. Oktober 1963, SUL/EC, 6-15.

⁵ Erst 1965 begann die *Data Equipment Corp.* mit der Vermarktung des RAND-Tablets unter der Bezeichnung *Grafacon*. Gleichzeitig wurde das bislang unter diesem Namen angebotene Gerät, das auch Engelbart verwendete, vom Markt genommen. Vgl. English et al. 1965, S. 90ff.

⁶ Engelbart 1986/87, Interview 3, Tape 1, Side B.

⁷ Maisel 1996.



Abbildung 41. Experimentelles Eingabegerät, das mit dem Knie zu bedienen war

automatisch ablief. Auf technischer Seite bedeutete dies, daß die Daten, die ein solches Gerät produzierte, ständig vom Computer überwacht werden und die Position des *Bug* (Käfer) genannten Positionsindikators nachgehalten werden mußte. Damit stellen indirekte Eingabegeräte hohe Anforderungen an die Reaktionszeit des Computersystems, die im Millisekundenbereich liegen muß.¹

Beim Entwurf beider Geräte orientierte sich Engelbart an Vorlagen aus anderen Technikbereichen. So stellte er fest, daß sich die Geschwindigkeit eines Autos erstaunlich feinfühlig über ein Pedal steuern ließ. Warum also kein Eingabegerät, das mit dem Fuß oder Bein zu bedienen wäre, zumal auf diese Weise beide Hände zur Bedienung der Tastatur freibleiben? Das Resultat dieser Überlegung war ein Eingabegerät, das aus einem Metallbügel bestand, der über dem Knie angebracht war (Abb. 41). Die Steuerung des *Bug* erfolgte durch die seitliche Bewegung des Knies für die horizontale und ein Heben und Senken des Knies (durch Abrollen des Fußes über den Ballen) für die vertikale Richtung.

Bei dem anderen Eingabegerät orientierten sich Engelbart und English am Planimeter (Abb. 42), einem mathematischen Gerät zum Messen des Flächeninhaltes ebener Figuren durch Umfahren ihrer Randkurve. Dies geschieht mit dem Ende des sogenannten Fahrstabes, an dessen Ende eine sogenannte Meßrolle befestigt ist, deren Drehung dem umfahrenen Flächeninhalt entspricht.² Engelbart und English variierten diese Idee, indem sie zwei Räder senkrecht zueinander

¹ Norberg et al. 1996, S. 125.

² Von seiner Funktionsweise her ist das Planimeter ein enger Verwandter der analogen Rechentechnik, die Vannevar Bush in den zwanziger und dreißiger Jahren entwickelt hatte. Vgl. Owens 1986.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

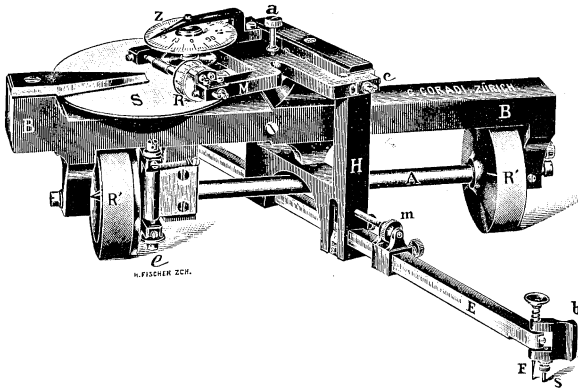


Abbildung 42. Planimeter der Firma G. Coradi, Zürich (1894)

anordneten, die über ein Potentiometer jeweils die Information über die horizontale bzw. vertikale Position lieferten. Diese Anordnung wurde zunächst in einem einfachen Holzgehäuse untergebracht, auf dessen Oberseite eine Taste angebracht war, die betätigt wurde, um die aktuelle Position des *Bug* zu bestätigen (Abb. 43).¹ Um die Namensgebung für dieses Eingabegerät ranken sich einige Legenden, Tatsache dürfte wohl sein, daß es mit seinem gedrungenen Gehäuse und dem langen Kabel tatsächlich an eine Maus erinnert. Der scherzhafte und inoffizielle Name war, ebenso wie das komplette Design, als Provisorium gedacht, und beide erwiesen sich als sehr dauerhaft.²

Das letzte der verwendeten Eingabegeräte war das sogenannte Grafacon, dessen Verwandtschaft mit dem Planimeter noch offensichtlicher ist. Es bestand aus einem Eingabearm, an dessen Spitze ursprünglich ein Zeichenstift befestigt werden konnte, welcher allerdings für die Versuche am ARC durch einen Taster ersetzt wurde. Dieser Arm konnte nach vorne und hinten verschoben und zur Seite gedreht werden, so daß man ganz intuitiv jeden Punkt einer Fläche ansteuern konnte.³

Bei mit diesen Geräten durchgeführten Experimenten sollten die Versuchspersonen bestimmte Teile eines Textes auf dem Bildschirm, d. h. einzelne Zeichen oder Worte, mit Hilfe des Eingabegeräts ansteuern und auswählen (engl. *point and click*). Über die Tastatur sollten sie dann die durchzuführenden Befehle eingeben.⁴ Die Versuche orientierten sich streng an Engelbarts Modell der Mensch-

¹ English et al. 1965, 92ff.; English et al. 1967, S. 7.

² Engelbart 1988, S. 195f.; Engelbart 1986/87, S. 38f.

³ English et al. 1965, S. 90ff. Anders als die Maus lieferte das Grafacon die Positionsdaten in Polarkoordinaten, die erst mit dem Computer in kartesische Koordinaten umgerechnet werden mußten.

⁴ English et al. 1965, S. 35.



Abbildung 43. Erste von William K. English konstruierte Maus aus dem Jahre 1964 (abgedruckt mit Genehmigung von SRI International)

Computer-Kommunikation. Im Sinne von Bewegungsstudien wurde die durchzuführende Tätigkeit bis in elementare Prozeduren zerlegt, die entweder vom Menschen oder von der Maschine ausgeführt wurden. Im Rahmen des experimentellen Systementwurfs konnten diese elementaren Prozesse dann unterschiedlich angeordnet oder in ihrer Abfolge variiert werden. Ein weiterer Vorteil dieser Vorgehensweise war, daß sich jede durchzuführende Aufgabe in Form eines Netzwerks bzw. Graphen darstellen und die mittlere Bearbeitungszeit berechnen ließ.¹ Diese Methodik wurde in den folgenden Jahren zu einem Werkzeug für die Implementierung von Benutzerkommandos beim On-Line System (NLS)² erweitert (Abb. 54, S. 204).³

Der grundsätzliche Ablauf der Auswahlexperimente wurde bereits vor Beginn der Testreihe in Form eines Computerprogramms festgelegt. Demnach mußte die Versuchsperson warten, bis ihm der Computer an einer beliebigen Stelle des Bildschirms einen Text präsentierte, der aus einer Matrix von 3×3 Buchstaben bzw. Worten bestand. Durch das Drücken einer Taste (auf der Tastatur) gab sie zu erkennen, daß sie mit der Auswahl des Textes beginnen wollte, bewegte dann den *Bug* bzw. den Lichtgriffel an die gewünschte Stelle und bestätigte schließlich durch Drücken eines Knopfes auf dem Eingabegerät (z. B. Maustaste) die Textauswahl. Der Computer registrierte die Zeiten, die die Versuchspersonen bis zur ersten Bewegung des Eingabegeräts bzw. bis zum Erreichen des Ziels benötigten sowie die Anzahl der Auswahlfehler (Abb. 44). Diese Versuche wurden sowohl

¹ English et al. 1965, S. 19ff.

² Da es sowohl ein Online- als auch ein Offline System gab, wurde das Akronym aus dem zweiten Buchstaben gebildet.

³ Engelbart and English 1968, S. 403f.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

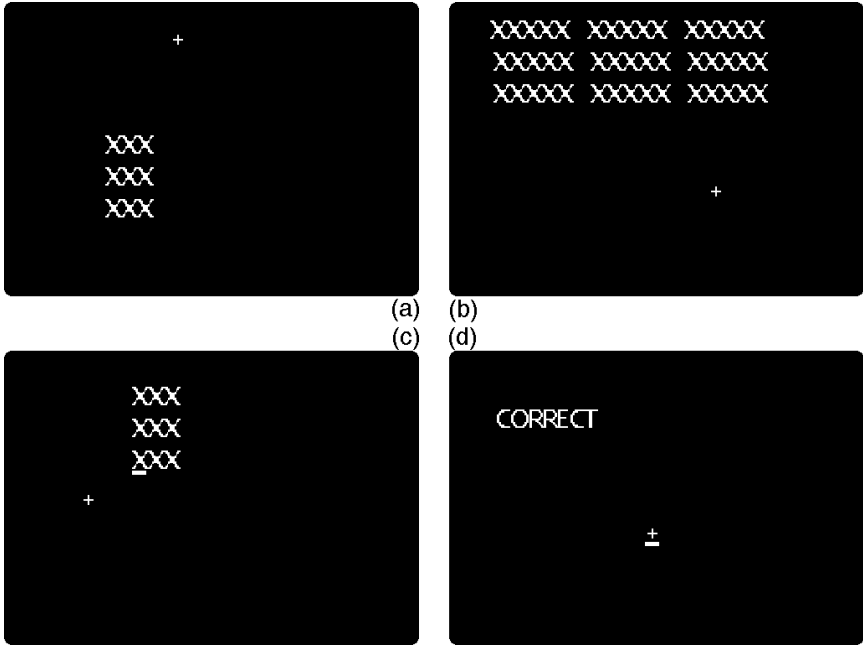


Abbildung 44. Unterschiedliche Ziele bei den Textauswahl-Versuchen. (a) Zeichenmodus: das mittlere X muß ausgewählt werden (b) Wortmodus: die mittleren fünf X müssen ausgewählt werden (c) Fehlerhafte Auswahl: die Zeichen bleiben sichtbar, das fehlerhaft ausgewählte Zeichen ist unterstrichen (d) Korrekte Auswahl: Die Position des Ziels wird durch den Cursor (Pluszeichen) angezeigt.

mit ungeübten als auch mit geübten Versuchspersonen durchgeführt, um neben der natürlichen auch die durch Training erreichbare Leistung zu ermitteln.¹

Die Experimente wurden nicht nur am Computer durchgeführt, die Meßwerte wurden auch in Echtzeit vom Computer aufgezeichnet und mit Hilfe von selbstentwickelten Statistikprogrammen ausgewertet. Schon bei diesen frühen Experimenten wurde also das *Bootstrapping* praktisch angewendet, obwohl die Entwicklung von computergestützten Werkzeugen zur Datenanalyse nicht zu Engelbarts

¹ English et al. 1965, S. 35ff.; English et al. 1967, S. 7ff. Im Nachhinein kritisierte English, daß die Messung von nur zwei Punkten der Lernkurve eigentlich keine sonderlich aussagekräftigen Ergebnisse geliefert habe. Vgl. Card et al. 1978, S. 601 .

Hauptzielen gehörte.¹ In den späten sechziger Jahren wurde diese Technik immer mehr zu einem Werkzeug für die Analyse und Validierung von Systementwürfen.²

Die Versuchsergebnisse waren gleichermaßen überraschend wie zukunftsweisend. Sowohl mit dem Steuerknüppel als auch mit dem Grafacon gab es Schwierigkeiten, einen Punkt auf dem Bildschirm präzise anzusteuern, da die Versuchspersonen dazu tendierten über das Ziel hinauszuschießen. Beim Lichtgriffel machte sich die ergonomisch unvorteilhafte Haltung des Arms bemerkbar, die zur schnellen Ermüdung führte. Gleichzeitig stellte sich der verwendete Lichtgriffel als wenig trennscharf heraus, so daß leicht Fehler bei der Auswahl einzelner Zeichen auftraten. Sowohl bei der Geschwindigkeit als auch bei der Genauigkeit waren die Maus und die Kniestuerung die besten Eingabemedien – wenn auch nur mit geringem Abstand.³

Dabei war die Maus, verglichen mit anderen Eingabegeräten, nicht einmal sonderlich genau. Umfuhr man mit ihr eine geschlossene Kurve, dann erreichte der *Bug* auf dem Bildschirm nicht wieder den Ausgangspunkt, weil die Räder auf der Unterseite der Maus bei diagonalen Bewegungen teilweise rutschten statt zu rollen. Weil der Nutzer aber über die Position des *Bug* auf dem Bildschirm eine ständige Rückmeldung erhielt, konnte er dennoch eine hohe Eingabegenauigkeit erreichen.⁴

Engelbart und English waren jedoch vorsichtig bei der Interpretation ihrer Versuchsergebnisse. Sie gaben zu bedenken, daß die Skalierung, d. h. die Empfindlichkeit der verschiedenen Geräte, unterschiedlich und die Treibersoftware des Joystick mangelhaft sei und daß das Grafacon und der Lichtgriffel konstruktiv noch verbessert werden könnten. Deshalb sei keine endgültige Bewertung möglich.⁵ Obwohl die Versuche methodisch nicht völlig überzeugend waren, wie Bill English im Nachhinein einräumte, wurde die Maus am ARC ab 1965 als Standardeingabegerät verwendet, zumal sie im Vergleich zum Lichtgriffel oder Grafiktablett sehr preiswert war.⁶

Wichtiger für die hohe Akzeptanz der Maus bei den Verwendern dürfte wohl die Tatsache gewesen sein, daß die Maus wesentlich benutzerfreundlicher war als

¹ English et al. 1965, S. 42ff.

² Engelbart 1969, S. 58f. Dieses Nebenprodukt von Engelbarts Arbeit wurde schon bald von anderen Abteilungen des SRI aufgegriffen und in deren Arbeit integriert. Vgl. Ball and Hall 1970.

³ English et al. 1965, S. 48ff.; Engelbart and English 1968, S. 397.

⁴ Engelbart and SRI-ARC Staff 1968.

⁵ English et al. 1965, S. 67ff.; Engelbart 1988, S. 196f.

⁶ Card et al. 1978, S. 601f. Bei der ersten Kleinserie, die 1967 bei einem Unternehmen im benachbarten San Carlos in Auftrag gegeben wurde, beliefen sich die Stückkosten einer Maus auf 162.50 \$. Vgl. »SRI Project 5890 ARNAS Requisitions, 1966–67«, SUL/EC, 2-3. Der Prototyp des RAND-Grafiktablets kostete dagegen 1963 knapp 3 000 \$. Vgl. Memorandum von Engelbart an J. H. Wensley, Tom L. Humphrey, Donald C. Lincicome, FKT, CPB, LJC, JJB, SDM über »Report on RAND Tablet«, 10. Oktober 1963, SUL/EC, 6-15.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

die anderen Eingabegeräte. Sie blieb da liegen, wo sie hingelegt wurde. Es gab kein Hantieren, um sie aus einer Halterung zu nehmen wie beim Lichtgriffel. Sie benötigte keine spezielle Unterlage wie ein Grafiktablett und war deswegen auch auf einem überfüllten Schreibtisch zwischen Papieren, Stiften und einer Kaffeetasse zu benutzen.¹

Die Maus, die eigentlich nur als Provisorium entworfen wurde, veränderte sich in ihrer Konstruktion im Laufe der nächsten Jahre nur sehr wenig und wurde 1970 für das Stanford Research Institute patentiert.² Da der Siegeszug der Maus als millionenfach produziertes Eingabemedium erst Mitte der achtziger Jahre einsetzte und diese Mäuse anders konstruiert waren, hatte das Patent für SRI bzw. Douglas Engelbart nur einen geringen finanziellen Wert. Engelbart selbst hat insgesamt nur etwa 40 000 \$ an Lizenzgebühren erhalten, die aus einer Lizenzvergabe an Apple Computers um 1980 stammten.³

Langfristig noch wichtiger als die Entwicklung der Maus selbst war die Erkenntnis, daß die Güte eines Eingabegeräts, sei es einer Tastatur oder eines grafischen Eingabegeräts, immer von der jeweiligen Aufgabe abhängig ist. English und Engelbart resümierten:

»... what is of importance is not the particular way, in which desk-top horizontal motion of the hand is actually converted into computer-entered coordinates for control of the bug ... What is important to fast efficient display selection is the particular *feel to the user* of the thing he grasps and moves ... «⁴

Sie mußten also erkennen, daß es beim Entwurf der Mensch-Computer-Schnittstelle keine Entweder-Oder Entscheidungen zu treffen gibt, sondern daß erst das Sowohl-als-auch eine gute Schnittstelle ausmacht.⁵

Speziell mit den Arbeiten im Bereich der grafischen Eingabegeräte wurde schließlich eine engere Kopplung des Benutzers an den Computer erreicht. Wie beim geistigen Vorläufer der Maus, dem Planimeter, werden die Bewegungen des Menschen in eine mathematische, maschinell zu verarbeitende Form übersetzt. Die Maus ist damit nicht nur ein Instrument zum Markieren von Objekten

¹ Engelbart 1973, S. 222. Lediglich die NASA, die die Entwicklung der Maus finanziert hatte, war nicht zufrieden, angeblich, weil sie in der Schwerelosigkeit nicht zu verwenden war. Vgl. Maisel 1996.

² Engelbart 1970b.

³ Maisel 1996; Engelbart 1986/87, Interview 3, Tape 1, Side B. Tatsächlich hat Engelbart 1978 das Patent an der Maus für 5 000 \$ von Tymshare, Inc. zurückerworben. Im April 1997 wurde Engelbart für die Erfindung der Maus mit dem Lemelson Award des MIT ausgezeichnet, der mit 500 000 \$ dotiert war. Vgl. Anonymous 1997; Louis 1997.

⁴ English et al. 1965, S. 67. Hervorhebung des Autors.

⁵ English et al. 1965, S. 69.

auf dem Bildschirm, sie ermöglicht dem Benutzer vielmehr die Bewegung innerhalb des künstlich geschaffenen »Informationsraums« der Mensch-Maschine-Schnittstelle. Auf diese Weise unterstützte kognitive Vorgänge waren nicht mehr rein symbolisch-abstrakter Natur, sondern waren konkrete körperliche Aktionen innerhalb des vom Computer geschaffenen Raums.

5.6 Servicesystem und Benutzersystem

Bevor nun im Detail auf die Software eingegangen werden kann, die in Engelbarts Labor seit 1963 entwickelt wurde, muß kurz erläutert werden, wie Engelbart sein eigenes Projekt im Vergleich zu anderen Softwareprojekten der Zeit sah.

Dabei kann zunächst festgestellt werden, daß Engelbart kaum Ambitionen bei der Entwicklung von Hardware und Betriebssystemsoftware im engeren Sinne hatte. Dies war die Aufgabe anderer großer Projekte, etwa des *Projects MAC* am MIT oder des *Projects Genie* in Berkeley, die eng mit Computerherstellern wie General Electric und Scientific Data Systems kooperierten. Obwohl die Forschung am ARC von der Existenz leistungsfähiger Time-Sharing-Systeme abhängig war, sah Engelbart keinerlei Notwendigkeit, selbst in diesem Bereich tätig zu werden. Schließlich waren die meisten der Projekte unter dem Dach der ARPA angesiedelt, so daß er einen schnellen und reibungslosen Technologietransfer erwarten durfte.

Nach Engelbarts eigener Einschätzung mußte man bei einem Computersystem das sogenannte *Servicesystem* und das darauf aufbauende *Benutzersystem* unterscheiden. Zum Servicesystem, dessen Gestaltung Engelbart nicht als sein eigentliches Ziel betrachtete, gehörten die Hardware und die darauf abgestimmte Systemsoftware. Zu den Parametern des Servicesystems gehören die Antwortzeit, die Speicherkapazität, die Zuverlässigkeit des Computers, die Art und Leistung der Peripheriegeräte, der Umfang der Kommandosprache usw.¹

Die Gestaltung des Servicesystems sollte nach Engelbarts Auffassung allein die Aufgabe haben, an das Benutzersystem angepaßte Funktionen zur Verfügung zu stellen. Dabei hatte Engelbart offenbar völlig andere Vorstellungen als die Entwickler der ersten Time-Sharing-Systeme, bei denen die Mensch-Computer-Kommunikation so aussah, daß ein Befehl über ein Fernschreibterminal eingetippt wurde, und der Computer innerhalb *mehrerer Sekunden* auf diesen Befehl reagierte. Engelbart hatte hingegen ein Arbeiten am Bildschirm im Sinn, das eine *sofortige* Reaktion des Computers auf alle Benutzereingaben erforderte, beispielsweise die unmittelbare Darstellung jedes einzelnen über die Tastatur einge-

¹ Engelbart 1965, S. 17.

gebenen Zeichens.¹ Aus dieser Sichtweise heraus erklärt sich die Überraschung der anderen ARPA-Vertragspartner, als ihnen Engelbart schon 1963 erklärte, die Antwortzeit eines Time-Sharing-Systems müsse für sinnvolle Anwendungen 250 Millisekunden oder weniger betragen.²

Schließlich forderte Engelbart, daß auch bei der Entwicklung des Benutzersystems auf allen Ebenen gleichzeitig begonnen werden müsse. In Anlehnung an seinen konzeptionellen Rahmen unterschied er eine Material-, Werkzeug- und Systemebene.

Auf der Materialebene seien Algorithmen anzusiedeln, die man für die Entwicklung von Anwendungsprogrammen benötige, also beispielsweise Sortierverfahren oder Algorithmen zur Projektion von dreidimensionalen Objekten auf den zweidimensionalen Bildschirm.³ Auf der Werkzeugebene war nach dieser Klassifikation der Großteil der damaligen Programme einzuordnen, die eine Effektivierung intellektueller Arbeiten zum Ziel hatten, weil sie Werkzeuge für eng begrenzte Arbeitsbereiche darstellten. Im Gegensatz dazu wollte Engelbart allgemeinere Werkzeuge und Methoden entwickeln, die für *alle* intellektuellen Tätigkeiten verwendet werden konnten. Demnach waren Ivan Sutherlands zukunftsweisendes Grafikprogramm Sketchpad und dessen Nachfolger lediglich Werkzeuge für die Erstellung von Konstruktionszeichnungen, das von Glen Culler entworfene »Culler-Fried System« ein Werkzeug für den Mathematiker oder das 1962 von Licklider und Welden Clark vorgestellte Programm ein Hilfsmittel für den Planer von Krankenhäusern.⁴

Obwohl auch am ARC Programme entstanden, die der Werkzeugebene zuzuordnen waren (z. B. zur Textverarbeitung), wollte Engelbart nicht auf dieser Ebene verharren, sondern alle Resultate der Material- und Werkzeugebene auf der Systemebene integrieren – zu seinem Intelligenzverstärker.

5.7 Das Offline-Text-System

Neben den Arbeiten zur Entwicklung und Bewertung von Ein- und Ausgabegegeräten wurde Ende 1963 auch mit der Entwicklung eines ersten Textbearbeitungsprogramms begonnen. Mit dieser Aufgabe betraute Engelbart Tom L. Humphrey, der neben seinem Studium an der Stanford University in Engelbarts Projekt mitarbeitete. Das Programm sollte allerdings im Gegensatz zu anderen frühen Editoren

¹ Lincicome, Donald L., »Man-Display Communication«, 1. Februar 1963, SUL/EC, 5-17.

² Engelbart 1986.

³ Engelbart 1965, S. 17f.

⁴ Engelbart 1965, S. 17f. Zu den einzelnen Programmen vergleiche Sutherland 1963, Culler and Huff 1962; Licklider and Clark 1962.

weit über die reine Textein- und -ausgabe hinausgehen. Engelbart hatte bereits sehr früh formuliert, wie für ihn die zentralen Funktionen des Textverarbeitungsprogramms aussahen:

»If it is just about as easy to accomplish as to state, ›Throw out this paragraph, and this and this, move this section here, insert that section from this other memo, re-number the sections according to their new order, change all occurrences of ›Smith‹ to ›Jones,‹ and send the result to Brown as a new memo,‹ then there will surely be . . . a notable difference in mode of work on both personal and group level.«¹

Innerhalb weniger Monate entwickelte Humphrey ein rudimentäres Programm, das der beschränkten Hardwareausstattung des ARC Rechnung trug. Damit war es grundsätzlich möglich, den kompletten Vorgang der Textbearbeitung am Computerbildschirm durchzuführen. Da aber bis 1967 lediglich ein einziges Terminal zur Verfügung stand, wurde das *Online-Text-System* (NLTS) durch ein *Offline-Text-System* (FLTS) ergänzt.²

Wie der Name bereits andeutet, wurde der Text beim Offline-Text-System weiterhin im Offline-Betrieb eingegeben. So hatte Engelbart in seinem Büro seinen eigenen Lochstreifenschreiber (engl. *tapewriter*), mit dem er alle anfallenden Schriftstücke, vor allem Notizen, interne Memos und Forschungsberichte, unmittelbar auf Lochstreifen eingeben konnte.³

Bei solchen Geräten wurde der eingegebene Text zur Kontrolle außerdem auf Papier ausgedruckt. Merkte der Benutzer schon während der Texteingabe, daß er Änderungen vornehmen mußte, etwa weil er sich vertippt oder ein Wort vergessen hatte, konnte er spezielle Befehlssequenzen eingeben, die bei der späteren Bearbeitung durch den Computer ausgeführt wurden. So konnten Buchstaben, ganze Worte, Zeilen oder Absätze gelöscht und in den Text eingesetzt werden.⁴

Außer diesen Editierkommandos gab es auch Kommandos, mit denen der Text für den späteren Ausdruck formatiert werden konnte. Diese Befehle betrafen vor allem die Groß- und Kleinschreibung sowie die Hervorhebung von Textteilen durch Unterstreichung. Eine spezielle Markierung von großzuschreibenden Buchstaben war bei den ersten Versionen des Offline-Text-Systems notwendig, da man bei den anfangs verwendeten Lochstreifenschreiber (Teletype Model 33 ASR von

¹ Memorandum von Engelbart an John H. Wensley, Tom L. Humphrey, Donald C. Lincicome, Elmer B. Shapiro und Roy C. Amara über »Introductory thoughts on a Transcription Central – a computer-aided, multiple-access, time-shared transcription system«, 29. Oktober 1963, SUL/EC, 17-6.

² Engelbart 1965, S. 24ff.

³ Engelbart 1965, S. 27f.

⁴ Engelbart 1965, S. 28f. Ein ähnliches Konzept verfolgte etwa zur gleichen Zeit auch Calvin Mooers bei seinem »Reactive Typewriter«. Vgl. Mooers 1963.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

<pre>+EXAMPLE: /THIS PASSAGE, SHOWN IN BOTH ORIGINATION FORM (/TELETYPE) AND IN POST- /TRANSLATOR FROMZ1WFORM (/FLEXOWRITER), ILLUSTRATES THE USE OF <UPPER-CASE> AND <UNDERLINING> +CONTROL +CODES. /MISCELLANEOUS EXAMPLESSSZ2C: +CDC +160A; 160/A; +ANFS//Q32; +ANFS/+Q-32. /IN GENERAL, MIXINGZ1L Z1I9L(+AN)Z2I</pre>	<pre>AN EXAMPLE: This passage, shown in both origination form (Teletype) and in post- Translator form (Flexowriter), illustrates the use of <u>upper-case</u> and <u>underlining</u> CONTROL CODES. Miscellaneous examples: CDC 160A; 160A; ANFS/Q32; ANFS/Q-32.</pre>
--	--

Abbildung 45. Beispiel für eine Ein- und Ausgabe beim Offline-Text-System

Bell) nur Großbuchstaben zur Verfügung hatte.¹ Abbildung 45 zeigt einen Text, wie er über den Lochstreifenschreiber eingegeben wurde (links) und das Formatierungsergebnis, wie es nach der Bearbeitung durch den Computer und dem Ausdruck auf einem Typenraddrucker (rechts) aussah.

Zunächst muß man zwei Typen von Zeichenketten unterscheiden: normalen Text und Befehlssequenzen. Weil man mit dem Lochstreifenschreiber nur eine kontinuierliche Folge von Zeichen eingeben konnte, mußte man ein Zeichen festlegen, das den Beginn eines Befehles kennzeichnete. Um kein weiteres Zeichen für die Kennzeichnung des Befehlsendes verschwenden zu müssen, wurde ein festes Format für jeden Befehl definiert. Editierbefehle wurden beispielsweise stets mit einem großen Z eingeleitet. Um das im Beispiel versehentlich eingegebene Wort FROM in der dritten Zeile zu löschen, mußte der Benutzer die Befehlssequenz Z1W (= Lösche rückwärts ein Wort) und das richtigen Wort FORM eingeben. In ähnlicher Weise wurde mit dem Befehl Z1L (= Lösche rückwärts eine Zeile) die komplette vorletzte Zeile gelöscht. Da das System alle Eingaben als Kleinbuchstaben interpretierte, mußte durch den Schrägstrich (/) und das Pluszeichen (+) markiert werden, daß der folgende Buchstabe bzw. das folgende Wort in Großbuchstaben geschrieben werden sollte. Zeichenketten, die durch < und > begrenzt waren, wurden unterstrichen.²

Zu einem wirklich mächtigen Werkzeug wurde das Offline-Text-System durch die Befehle zum Einsetzen von Zeichenketten an beliebigen Stellen des (vorhergehenden) Textes. Ein solcher Einsetz-Befehl hatte eine komplexere Form als die einfachen Löschbefehle und wurde wegen seiner flexiblen Länge durch die Zeichenketten Z1I und Z2I begrenzt. Zwischen diesen Begrenzungszeichen wird

¹ Engelbart 1965, S. 27f.

² Engelbart 1965, S. 27f. Da durch dieses Vorgehen die entsprechenden Zeichen für den eigentlichen Text fehlten, mußten diese über weitere Befehlssequenzen eingegeben werden.

die (relative) Position und der einzufügende Text angegeben. In Abbildung 45 findet sich ein solcher Befehl in der letzten Zeile. Der Ort, an dem Text eingesetzt werden soll, ist mit 9L angegeben, also neun Zeilen (lines) zurück. Dort wird die Zeichenkette +AN eingesetzt. Die Positionsangabe kann aber auch komplizierter ausfallen. Die Angabe 2L3W1C bezeichnet beispielsweise eine Position, die 2 Zeilen (lines), 3 Worte und 1 Zeichen (character) zurückliegt.

Mit Hilfe von Einsetz-Befehlen konnten beliebige Zeichenfolgen, also auch weitere Z-Befehle eingegeben werden. Durch diese Möglichkeit ließen sich Befehle eingeben, deren Komplexität (technisch) nur durch die Hauptspeichergröße des Computers begrenzt wurden. Bei der Verwendung von verschachtelten Einsetz-Befehlen konnte der Benutzer allerdings leicht die Übersicht verlieren, so daß die Befehle bei der späteren Bearbeitung durch den Computer nicht die gewünschten Resultate lieferten oder sogar die vorzeitige Beendigung der Bearbeitung erforderlich machten.¹

Für die Strukturierung eines Textes standen bei diesem System zunächst kaum Werkzeuge zur Verfügung. Es sah lediglich die Einteilung des Textes in Absätze vor, die – möglicherweise in Anlehnung an die Terminologie der Programmiersprache ALGOL – als *statements* bezeichnet wurden. Diese Wortwahl deutet an, daß Engelbart den Absatz als kleinste Sinneinheit innerhalb des Textes verstand. Für die Formatierung von Absätzen existierte der Befehl *ZnT*, mit dem ein Absatz um *n* Tabulatorsprünge eingerückt werden konnte.²

Der Lochstreifen, den man auf diese Weise mit dem Offline-Text-System produziert hatte, konnte dann in den Computer eingelesen und verarbeitet werden. Der bearbeitete Text wurde wieder auf einem Lochstreifen gespeichert und konnte dann ausgedruckt oder weiterverarbeitet werden. Nachdem beim CDC-3100-Computer auch Magnetbänder und -platten als Sekundärspeicher zur Verfügung standen, wurde die Funktionalität des Offline-Text-Systems deutlich erweitert. Mit der Inbetriebnahme des laboreigenen Time-Sharing-Computers wurde das Offline-Text-System 1967 vollständig durch das Online-Text-System abgelöst.³

In einem ersten Verarbeitungsschritt wurde der Text in den Hauptspeicher des Computers eingelesen und rückwärts nach Lösch- und Einsetzbefehlen durchsucht, die unmittelbar vom Computer ausgeführt wurden. Im zweiten Schritt wurde der Text dann für die Ausgabe vorbereitet. Dazu wurde er vorwärts nach den Steuerzeichen für Großschreibung, Unterstreichung und Sonderzeichen durchsucht, unnötige Leerzeichen und Tabulatoren wurden entfernt und Einrückungen

¹ Engelbart 1965, S. 31ff.

² Engelbart 1965, S. 37; Wilkes 1998, S. 23.

³ Engelbart 1986/87, Interview 2, Tape 2, Side B.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

durchgeführt. Der so formatierte Text wurde wieder auf einen Lochstreifen ausgegeben, der dann (wiederum Offline) auf einem Flexowriter ausgedruckt werden konnte.¹

Um die Benutzung des Offline-Text-System für eine möglichst repräsentative Aufgabe zu testen, verwendete es Engelbart Anfang 1965, um einen 72seitigen Forschungsbericht an das AFOSR zu verfassen. Die dabei gesammelten Erfahrungen stimmten Engelbart sehr zuversichtlich für die weitere Arbeit des ARC.

Zunächst stellte er befriedigt fest, daß es selbst mit den bescheidenen Mitteln des Offline-Text-Systems möglich sei, einen längeren wissenschaftlichen Text komplett am Computer zu schreiben und dabei nicht mehr Zeit zu benötigen als mit konventionellen Mitteln – es war schließlich nicht vorhersehbar, ob bereits der erste Systementwurf eine Effektivitätssteigerung gegenüber der herkömmlichen Arbeitsweise darstellen würde. Darüber hinaus meinte er, daß das System die Effektivität des Schreibens verbesserte, »solange die Wörter fließen.«² Erst, wenn er das System dazu verwende, um bislang unstrukturierte Gedanken in eine stringente Form zu bringen, zeigten sich die Schwächen des Offline-Text-Systems. Engelbart war aber zuversichtlich, daß dies beim Übergang zu einem Online-Textbearbeitungssystem behoben werden könnte.³

Nachdem der CDC 160-A-Computer 1965 durch den CDC 3100 ersetzt worden war, wurde das Offline-Text-System so erweitert, daß bereits auf Magnetband oder -platte gespeicherte Texte bzw. Textteile in neue Dokumente eingefügt werden konnten. Dadurch stellte sich ein erheblicher Rationalisierungseffekt ein, der mit der Effektivitätssteigerung des Programmierers bei der Einführung von Unterprogrammbibliotheken vergleichbar war.⁴

Vor allem aber stellte Engelbart mit einiger Verwunderung fest, daß es nicht das gleiche war, einen Text mit den herkömmlichen Mitteln oder am Computer zu verfassen:

»However the feeling of being tied into a system made a surprising difference in the work itself, revealing new kinds of inadequacies, and new kinds of possibilities.«⁵

Obwohl er es nicht explizit formulierte, konnte sich Engelbart also 1965 um einiges sicherer sein, daß die Wahl des Werkzeugs tatsächlich Einfluß auf Form und Inhalt intellektueller Arbeit hatte, und damit seine neo-whorfsche Hypothese zutreffend war.

¹ Engelbart 1965, S. 51ff.

² Engelbart 1965, S. 46.

³ Engelbart 1965, S. 46f.

⁴ Engelbart 1965, S. 45; Engelbart and Huddart 1965; Campbell-Kelly and Aspray 1996, S. 186.

⁵ Engelbart 1965, S. 3.

5.8 Das Online-System: Prototyp des Intelligenzverstärkers

5.8.1 Ein Time-Sharing-Computer für das Online-System

Parallel zum Offline-Text-System wurde seit 1964 auch die erste Version des Online-Text-Systems (NLTS) entwickelt, mit dem die bereits geschilderten Online-Eingabeexperimente durchgeführt wurden.¹ Wegen der geringen Leistung des CDC 160-A kam die Entwicklung eines Online-Systems, mit dem sich auch die täglichen Arbeiten bewältigen liessen, erst richtig in Gang, als Ende 1965 ein leistungsfähigerer CDC-3100-Computer angeschafft wurde.

In Anlehnung an die damals populäre Idee der »computer utility«² schlugen Engelbart und seine Mitarbeiter schon 1965 die Implementierung von Dienstleistungen vor, die nicht nur dem eigenen Labor, sondern dem ganzen SRI zugute kommen sollten. An etwa 20 Terminals sollte nicht nur das Textverarbeitungssystem, sondern auch ein Programmentwicklungssystem und ein »calculating service« angeboten werden.³ Es stellte sich aber bald heraus, daß auch der CDC 3100 nicht ohne weiteres für den Time-Sharing-Betrieb geeignet war. Da keine Ambitionen bestanden, die notwendige Betriebssystemsoftware selbst zu entwickeln und die ersten kommerziellen Time-Sharing-Computer bereits angekündigt waren, entschloß man sich schon nach wenigen Monaten, den CDC 3100 möglichst bald durch einen Time-Sharing-Computer zu ersetzen.

In der Zwischenzeit wurde die Software für das Online-System auf dem CDC 3100 weiterentwickelt, obwohl dessen Betriebsweise kein *Bootstrapping* im engeren Sinne ermöglichte. Da stets nur ein einziger Benutzer am Computer arbeiten konnte, mußten die Nutzungszeiten penibel geplant werden, wobei Programmieraktivitäten eine höhere Priorität hatten als die Erstellung von Textdokumenten. Es kam erschwerend hinzu, daß die Institutsleitung eine exklusive Nutzung des CDC 3100 durch das ARC unter Hinweis auf die jährliche Miete von etwa 80 000 \$ nicht zuließ. Engelbart mußte daher anderen Forschungsgruppen einen erheblichen Anteil der Rechenzeit überlassen, während der die Mitarbeiter des ARC keinen Zugriff auf ihren Computer hatten.⁴

Im Frühjahr 1966 beschäftigten sich English und Engelbart deshalb mit der Frage, welches der in Entwicklung befindlichen Time-Sharing-Systeme ihren Anforderungen am ehesten gerecht wurden. Anfangs wurde dabei die Anschaffung

¹ Engelbart 1965, S. 4.

² Greenberger 1964; Corbató und Vyssotsky 1965; Irwin 1966; Hellige 1996a, S. 224ff.

³ Memorandum von William K. English an das On-Line Computer Committee über »Multi-User System«, 27. Mai 1965, SUL/EC, 15-1.

⁴ Memorandum von Torben Meisling über »Meeting with Ivan Sutherland and Bob Taylor in Washington, October 7, 1965«, 15. Oktober 1965, SUL/EC, 1-29.

eines CDC-3300-Computers in Betracht gezogen, auf dem die bislang geschriebenen Programme ohne Änderung lauffähig gewesen wären. Es stellte sich allerdings schnell heraus, daß das Time-Sharing-Betriebssystem für diesen Computer nicht den hohen Anforderungen des ARC gerecht wurde.¹ So konzentrierte sich ihr Interesse bald auf den SDS-930-Computer von *Scientific Data Systems*, der im Rahmen des von der ARPA geförderten *Project Genie* an der Universität Berkeley zu einem leistungsfähigen Time-Sharing-System weiterentwickelt wurde, das ab 1966 als SDS 940 vermarktet wurde.² Neben der Förderung durch die ARPA und die räumliche Nähe schien es auch vorteilhaft, daß zwei wichtige Mitarbeiter des *Project Genie*, Butler Lampson und L. Peter Deutsch, schon 1965 am ARC gearbeitet hatten und mit den dortigen Arbeiten vertraut waren.³

Obwohl er sich noch im Teststadium befand, wurde im Oktober 1966 ein SDS 940 für das ARC bestellt, dessen Lieferung für das Frühjahr 1967 zugesichert wurde, sich dann aber doch bis Juni verzögerte.⁴ Im Gegensatz zu den bisher verwendeten Maschinen von Control Data war der SDS 940 ein Mainframe-Computer der mittleren Leistungs- und Preiskategorie. Der beim ARC installierte Computer kostete 430 000 \$ und hatte einen Hauptspeicher mit 65 536 24-bit-Worten und einer Zugriffszeit von 1,75 μ s, einen Trommelspeicher mit 4,5 MByte und einen Plattenspeicher mit 96 MByte Speicherkapazität. Als weitere Speichermedien standen drei Magnetbandlaufwerke und ein Bryant 4061 Wechselpplattenlaufwerk zur Verfügung.⁵

5.8.2 Konvergenz von Fernsehen und Computer

Ein leistungsfähiger Time-Sharing-Computer war freilich allein noch nicht genug. Jedes der zunächst sechs, später zwölf Terminals mußte mit einer modifizierten Schreibmaschinentastatur, einer Einhandtastatur, einer Maus sowie mit einem hochauflösenden 17-Zoll-Bildschirm ausgestattet werden (Abb. 46). Da Engelbart

¹ Engelbart et al. 1968, S. 13; Engelbart, D. C., »Pre-proposal (to ARPA) thinkpiece about providing expanded facilities for the SRI Augmented Human Intellect program«, 30. Juni 1966, SUL/EC, 1-18.

² Lichtenberger and Pirtle 1965; Lampson et al. 1966; Bell and Newell 1971, S. 542-560.

³ Brief von Engelbart an Seymour Jeffrey, Electronic Systems Division, 25. Mai 1965, SUL/EC, 15-2.

⁴ English, W. K., »Computer Augmented Management System«, Monthly Letter Report No. 8 to Rome Air Development Center, 11. November 1966, SUL/EC, 1-3; Engelbart, D. C., »Computer Augmented Management System«, Monthly Letter Report No. 16 to Rome Air Development Center, 3. Juli 1967, SUL/EC, 1-11.

⁵ Weisberg 1966; Anonymous 1965, S. 722; Engelbart, D. C., »Computer Augmented Management System«, Monthly Letter Report No. 15 to Rome Air Development Center, 3. Juli 1967, SUL/EC, 1-11.



Abbildung 46. Einfaches Terminal für das Online-System, bestehend aus Bildschirm, Einhand- und Schreibmaschinentastatur sowie der ersten in Kleinserie gefertigten Maus, um 1968.

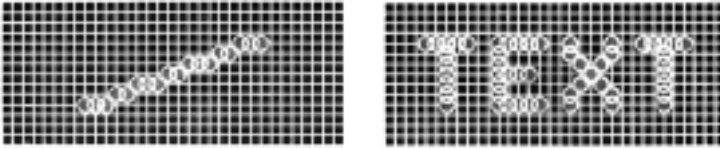
ein System vorschwebte, an dem jeder Benutzer mehrere Stunden täglich arbeitete, waren gerade die Eigenschaften des Bildschirms, wie Auflösung und Flimmerfreiheit, von entscheidender Bedeutung. Außerdem sollte die Ansteuerung der Bildschirme die Leistung des Computers nicht wesentlich beeinträchtigen.

Hier galt es, einen Kompromiß zwischen dem Gewünschten und dem Realisierbaren zu erzielen. Aus diesem Grund baten Engelbart und English im Sommer 1966 eine Vielzahl von Unternehmen um Angebote für das Bildschirmsystem. Das Resultat ihrer Recherche war ernüchternd: Die für ihre Zwecke geeigneten Monitore kosteten zwischen 15 000 und 20 000 \$. Die Anschaffung solcher Monitore hätte bereits die Hälfte des Budgets aufgezehrt, das dem ARC für die Anschaffung von Hardware zur Verfügung stand.¹

Leisten konnte man sich allerdings Bildschirme, die ursprünglich gar nicht als Computerperipherie entwickelt worden waren: Schwarzweiß-Videomonitore, d. h. Fernsehgeräte ohne Empfangsteil, waren preiswerte, sehr zuverlässige und

¹ »Correspondence and Memoranda concerning a Display System for the 940«, SUL/EC, 16-6. Für Informationen über die Preise von Computerbildschirmen und ihre Leistungen vergleiche Frank 1965; Lewin 1967; Machover 1967; Machover 1972.

5. Douglas Engelbart und der Computer als Intelligenzverstärker



DOT-BY-DOT method of generating lines and characters was originally used in computer graphics. Newer analogue techniques are much faster and create cleaner images (below).

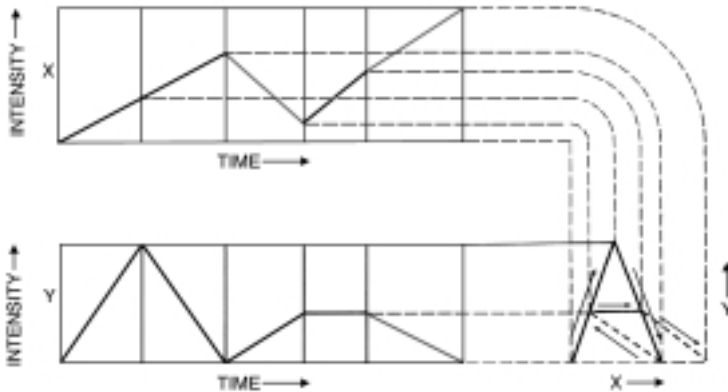


Abbildung 47. Rastergrafik (oben) und Vektorgrafik (unten). Bei der Vektorgrafik werden wie beim Oszilloskop getrennte Signale für die x - und y -Ablenkung des Elektronenstrahls erzeugt. Die Originalbildunterschrift deutet an, daß die Rastergrafik um 1970 nicht unbedingt als zukunftsweisende Technik galt.

relativ einfache Geräte. Unglücklicherweise konnte man sie aber nicht ohne weiteres an einen Computer anschließen, weil sie nach einem anderen technischen Verfahren betrieben wurden als damals gängige Computermonitore.

Bei einem Fernsehbildschirm ist die Bildschirmfläche in eine große Anzahl von Zeilen eingeteilt, die vom Elektronenstrahl mit Hilfe einer Ablenkeinheit in schneller Folge überstrichen werden. Dieses Zeilenmuster wird Raster genannt, weshalb man auch von Rasterbildschirmen spricht. Die Intensität des Elektronenstrahls wird entsprechend der Helligkeit der Bildpunkte (*picture elements, pixels*) variiert. Die für die Ansteuerung eines solchen Monitors benötigte Elektronik ist nicht sonderlich kompliziert, so daß die Gerätekosten gering bleiben.

Um einen Rasterbildschirm als Ausgabemedium nutzen zu können, mußte jeder Punkt des darzustellenden Bildes im Hauptspeicher des Computers abgelegt werden. Schon bei einem schwarzweißen Fernsehbild mit 625 Zeilen zu 833 Punkten ergab sich ein Speicherbedarf von mehr als 500 kbit.

Die meisten der damaligen Computerbildschirme funktionierten anders (vgl. Abb. 47). Bei den sogenannten Vektorbildschirmen wurden die Zeichen und Grafiken nicht aus einzelnen Punkten zusammengesetzt, sondern wie beim Oszilloskop vom Elektronenstrahl auf den Bildschirm »gemalt«. Solche Bildschirme benötigten nicht übermäßig viel Speicherplatz, da lediglich eine Liste mit den Koordinaten der darzustellenden Linien bzw. Zeichen verwaltet werden mußte. Vektorbildschirme waren zwar wegen der aufwendigeren Steuerelektronik deutlich teurer als Videomonitore, aber wegen des geringeren Speicherbedarfs in den sechziger Jahren weit verbreitet.

Vektorbildschirme waren schnell und erzeugten sehr scharfe Zeichen und Liniengrafiken. Es konnten allerdings nur Grafiken mittlerer Komplexität angezeigt werden, da sonst der Bildschirm zu flackern begann. Aus diesem Grund lagen die größten Schwächen der Vektorgrafik bei der Darstellung von Flächen und Graustufenbildern, die zeitaufwendig durch eine Vielzahl paralleler Linien erzeugt wurden. Rasterbildschirme hatten zwar Schwächen bei der Darstellung von Linien, konnten aber flächige Objekte durch einen Schwarm von Pixeln darstellen, die bei jedem Überstreichen vom Elektronenstrahl automatisch zum Leuchten angeregt wurden. Ihre vielleicht größte Stärke war, daß die Ausgabegeschwindigkeit nicht von der Komplexität des Bildschirminhalts abhing.

Es gab also gute Gründe für die Verwendung von Rasterbildschirmen. Die Videotechnik – in Form des Fernsehens – war während der sechziger Jahre schon so allgegenwärtig, daß es enorme technische und wirtschaftliche Vorteile mit sich gebracht hätte, wenn man sie mit der Computertechnik hätte verbinden können. Deshalb hatten Wissenschaftler an verschiedenen Laboratorien schon seit den frühen sechziger Jahren mit Experimenten begonnen, um die beiden Technologien zu verschmelzen. Aber der Fortschritt war langsam und begrenzt.¹

Allein aus Kostengründen waren um 1967 nur die wenigsten Computer mit so viel Hauptspeicher ausgestattet. Auch der SDS-940-Computer des ARC besaß nicht genügend Speicher, um nur einen einzigen Rasterbildschirm zu betreiben.² Deshalb entwarf Bill English im Frühjahr 1967 eine ebenso preiswerte wie leistungsfähige Alternative zur Rastergrafik, die am ARC einige Jahre in Gebrauch war.³

Bei diesem System, dessen Grundgedanke bereits 1964 von John Yarborough formuliert worden war⁴, wurden zwölf hochauflösende (Vektor-)Grafikmonitore mit einer Bildhöhe von nur 5 Zoll an den Computer angeschlossen. Das Bild

¹ Sutherland 1970; Bell 1988, S. 30; Newman 1976; Sutherland et al. 1969.

² Engelbart and SRI-ARC Staff 1970, S. 63.

³ Engelbart et al. 1968, S. 23; English, W. K., »Proposed storage-tube display system for the 940«, 7. Dezember 1966, SUL/EC, 16-6.

⁴ Memorandum von John M. Yarborough an Douglas C. Engelbart, DGW und Bonnie J. Huddart über »Televised Display for Project MAST Training Stations«, 30. Januar 1964, SUL/EC, 17-6.

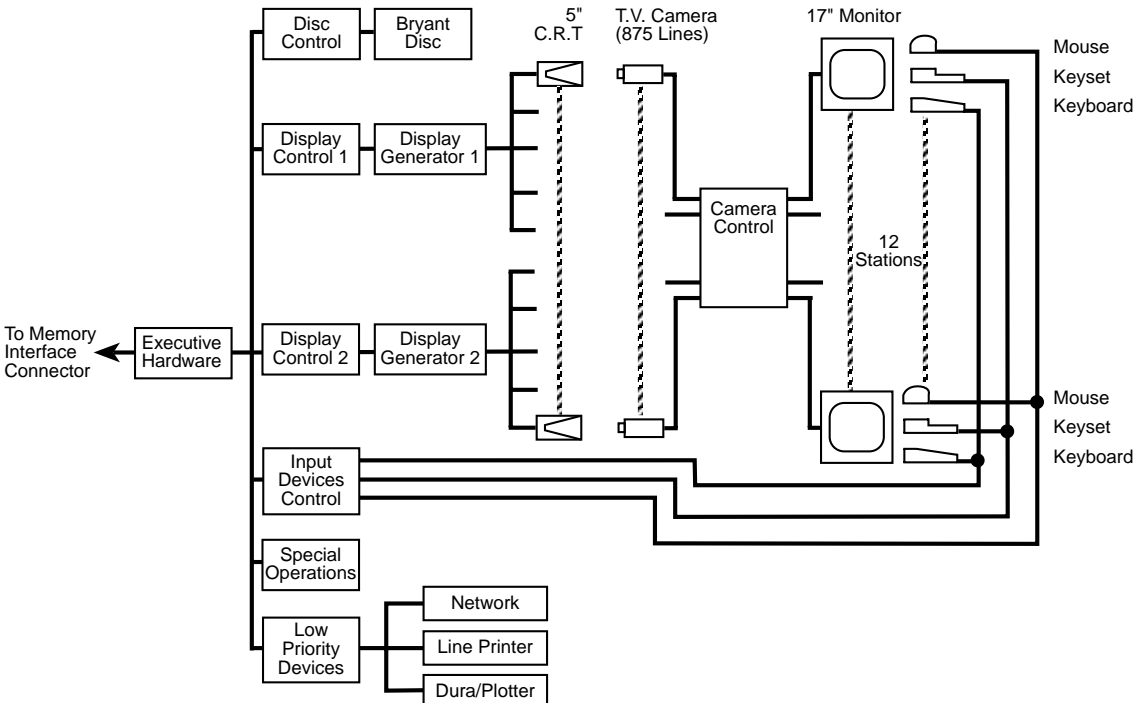


Abbildung 48. Kanal für spezielle Systemkomponenten mit dem Bildschirmausgabesystem und einer Möglichkeit zum Anschluß an das geplante ARPANET

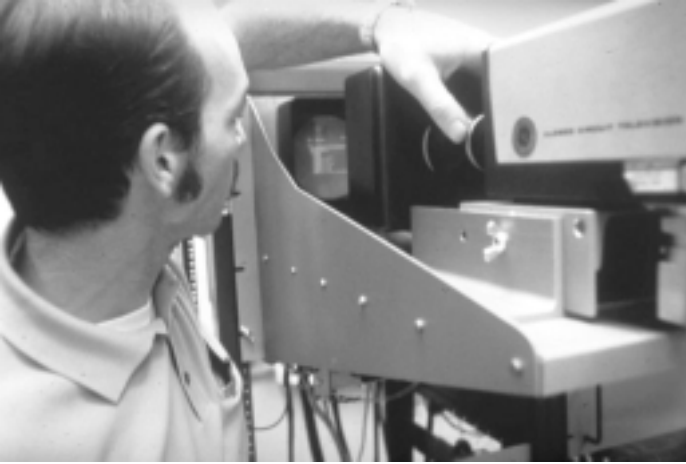


Abbildung 49. Teil des Bildschirmsystems. Die Anzeige des Computermonitors wird von einer Videokamera aufgenommen und zum Fernsehbildschirm des Terminals übertragen. Der Techniker justiert die Schärfe der Videokamera.

auf jedem Monitor wurde von einer Fernsehkamera aufgenommen, an die wiederum ein 17-Zoll-Fernsehschirm angeschlossen war (Abb. 48, 49). Da es sich sowohl bei den Kameras als auch bei den Bildschirmen um handelsübliche Geräte von General Electric handelte, konnten die Kosten pro Terminal auf etwa 5 500 \$ reduziert werden. Da der Fernsehbildschirm nur 600 \$ kostete, eröffnete sich die Möglichkeit, selbst sporadischen Nutzern des Systems ein eigenes Terminal zur Verfügung zu stellen. Vor allem aber erlaubte die Trägheit der verwendeten Aufnahmeöhre eine flimmerfreie Darstellung, obwohl die kleinen Monitore nur mit einer Wiederholrate von 15 Bildern/s arbeiteten.¹ Schließlich gelang es auf diese Weise auch, daß weniger als 1% der Prozessorleistung für die Bildschirmausgabe benötigt wurde.²

Die Verwendung der Fernsehtechnik für das Ausgabesystem eröffnete auch weitere, ursprünglich nicht erwartete Möglichkeiten. Man konnte das Videosignal beispielsweise invertieren und so eine für das menschliche Auge angenehme Darstellung von schwarzen Zeichen auf weißem Untergrund realisieren. Eine weitere Möglichkeit bestand darin, die Computerausgaben mit beliebigen ande-

¹ Die einzigen Probleme traten bei den schnell bewegten *Bugs* auf, die aus diesem Grund einen Schweif hinter sich herzogen. Vgl. Engelbart and SRI-ARC Staff 1968.

² Engelbart et al. 1968, S. 17; Engelbart 1969, S. 11f. So exotisch, wie es auf den ersten Blick scheinen mag, war diese Lösung freilich nicht. Wegen ihrer unbestreitbaren Vorteile wurde eine ähnliche Anordnung z. B. auch bei der RAND Corporation verwendet. Vgl. Uncapher 1971.

5. Douglas Engelbart und der Computer als Intelligenzverstärker



Abbildung 50. Kooperatives Arbeiten am Bildschirm, 1967. Von links nach rechts: Don Andrews, Dave Hopper, William K. English, Douglas C. Engelbart und Barry Wessler (Assistent von Robert Taylor)

ren Videosignalen, etwa dem Videobild eines entfernten Gesprächspartners zu mischen (Abb. 59, S. 216). Schließlich konnte das Videosignal an mehrere Bildschirme gleichzeitig übertragen werden, so daß mehrere Benutzer gleichzeitig an einem Dokument arbeiten konnten. Das Recht zur Eingabe und Modifikation von Daten besaß dabei allerdings immer nur *einer* der Benutzer. Diese Möglichkeit wurde schon bald zur Unterstützung von Arbeitsgruppen ausgenutzt. Dazu hatte Engelbart den Tisch in einem Konferenzraum mit einer Reihe von Monitoren, Tastaturen und Mäusen ausstatten lassen (Abb. 50).¹

Die computergestützte Zusammenarbeit paßte genau in Engelbarts Konzept, bot sie doch die Möglichkeit, eine Gemeinschaft der Informationsarbeiter aufzubauen, deren Fähigkeit zur Problemlösung größer wäre als die des Einzelnen. Diese Entwicklung stellt ein treffendes Beispiel für die Koevolution der technischen und menschlichen Komponenten eines Mensch-Computer-Systems dar, wie sie Engelbart anstrebte.

Aber nicht nur technisch betrat man beim Entwurf der Terminals Neuland. Bis zu diesem Zeitpunkt sahen die meisten Bildschirmterminals wie Gefechtsstände in einem U-Boot aus, bei denen der Bildschirm und die Tastatur fest in einem Metallschrank eingebaut waren (vgl. Abb. 12 und 29). Engelbart entschied, mit dieser Tradition zu brechen und stattdessen Einzelgeräte für die Terminals zu ver-

¹ Engelbart 1969, S. 12, 60ff.; Licklider et al. 1968, S. 24.



Abbildung 51. Systemprogrammierer Bill Paxton an der beliebten Yoga Workstation

wenden, die jeder Benutzer nach seinen persönlichen Vorlieben anordnen konnte. So entstanden Terminaltypen für unterschiedliche Anforderungen und Geschmäcker. Bei der Yoga Workstation (Abb. 51) war der Bildschirm auf einem kurzen Dreibein montiert, während die Tastatur und die anderen Komponenten auf einem flachen Tisch standen, vor dem man im Lotussitz Platz nahm.¹ In Zusammenarbeit mit dem Büromöbelhersteller Herman Miller, Inc. wurde ein Arbeitsplatz entworfen, bei dem alle Komponenten in eine modische zweiteilige Plastikkonsole integriert waren (Abb. 52). Dabei wurde ein schwenkbares Tablett am Schreibtischstuhl montiert, auf dem sich die Tastaturen und die Maus befanden, während das andere Teil für den Monitor gedacht war.²

Mit dem SDS-940-Computer und seinen ausgefeilten Ein- und Ausgabetechniken sowie der leistungsfähigen Time-Sharing-Software des *Project Genie* standen Engelbart und seinen Mitarbeitern nun alle Möglichkeiten offen, durch *Bootstrapping* die Programme zu entwickeln, die aus dem Computer ein nützliches Werkzeug und Medium für den Informationsarbeiter machen sollten. Diesem Ziel kam Engelbarts Labor mit der Entwicklung der als NLS (für Online-System) bezeichneten Software in den Jahren 1967 und 1968 einen großen Schritt näher.

¹ Die Beliebtheit der Yoga Workstation gibt einen Hinweis, daß die kalifornische Gegenkultur und Hippiebewegung auch vor dem sonst eher konservativen SRI nicht Halt machte. Die jüngeren Labormitglieder ließen ihre Haare lang wachsen, und selbst Engelbart, der zu diesem Zeitpunkt bereits über Vierzig war, ließ sich kurzzeitig einen Vollbart stehen.

² Engelbart et al. 1968, S. 24; Engelbart 1988, S. 200ff.

5. Douglas Engelbart und der Computer als Intelligenzverstärker



Abbildung 52. Herman Miller Workstation

5.8.3 Bootstrapping in der Praxis: Das NLS-Softwaresystem

Die Erweiterung der Routinen für die Online-Experimente auf dem CDC 160-A zu einem eigenständigen Online-Text-System (NLTS), das später einfach nur Online-System genannt (NLS) wurde, begann im Herbst 1965 mit der Anschaffung des CDC 3100 Computers. Zu diesem Zeitpunkt hatte Engelbart die Gewißheit, daß die Verstärkung der menschlichen Intelligenz mit Hilfe des Computers grundsätzlich möglich sei, gleichzeitig hatten seine Mitarbeiter eine Reihe von Programmen entwickelt, die nun zu einem Prototypen des Intelligenzverstärkers zusammengeführt werden sollten. Nachdem im Oktober 1966 auch die gemeinsame Förderung durch die ARPA und das RADC einsetzte, konnte Engelbart mit einem guten Dutzend Mitarbeiter diese Aufgabe forciert angehen.¹

Die Entwicklung des NLS umfaßte allerdings weit mehr als eine Portierung des Offline-Text-Systems auf den neuen Computer und eine Erweiterung für die Arbeit am Bildschirm. Obwohl Engelbarts Hauptinteresse in der Entwicklung des Benutzersystems lag, mußte er sich notgedrungen auch auf die Weiterentwicklung des Servicesystems einlassen, da die benötigten Hilfsmittel (Betriebssystemsoftware, Programmiersprachen, etc.) entweder nicht existierten oder sich noch in der Entwicklung befanden. Vorrangiges Ziel war jedoch die evolutionäre Entwicklung des komplexen Hardware-Software-Systems, der Entwurf neuer Verfahren zur Bearbeitung bestimmter Aufgaben, zur Konzeptionalisierung, Organisation

¹ Engelbart 1969, S. 6; Stanford Research Institute Weekly Time Reports, SUL/EC, 7 und 8.

und Visualisierung von schriftlichem und grafischem Material sowie neuer Ansätze für individuelles und kooperatives Arbeiten mit Hilfe des Computers.¹

Die erste Version des Online-Systems war in der Assemblersprache für den CDC 3100 programmiert und lief unter der Kontrolle des selbstentwickelten »Controlled On-Line Program Executer« (COPE), einem einfachen Systemprogramm, das speziell für den Betrieb eines Computers mit Bildschirmterminal entwickelt worden war. Es übernahm gleichzeitig die Übersetzung der für den 160-A in FORTRAN entwickelten Routinen und machte so erst die Übersetzung der in verschiedenen Sprachen geschriebenen Programme mit dem von CDC gelieferten Compiler möglich.²

Da sich Engelbart der Ineffizienz dieses Vorgehens bewußt war, wurde das NLS für den SDS 940 in eine Reihe von unabhängigen Modulen aufgeteilt, für die jeweils die geeigneten Entwurfsmethoden und -werkzeuge hergestellt werden konnten (Abb. 53). Da Scientific Data Systems als Standardsoftware neben ihrem Makroassembler ARPAS nur Compiler für FORTRAN IV, ALGOL 60 und Basic liefern konnte³, und sich diese Programmiersprachen nur schlecht für die Systemprogrammierung eigneten, entschloß man sich, zwei spezialisierte Programmiersprachen zu entwickeln. Dies waren die *Command Meta Language* (CML) für die Definition des Mensch-Computer-Dialogs und die *Machine Oriented Language* (MOL) für alle weiteren Programmieraufgaben.

Command Meta Language

Die *Command Meta Language* basierte auf Engelbarts Versuch, den Mensch-Computer-Dialog zu formalisieren und damit der Analyse und systematischen Gestaltung zugänglich zu machen. Neu bei der CML war dabei die Trennung der Benutzeraktionen von der verwendeten Hardware. Darum wurde die CML so entworfen, daß sie das Wissen über die Struktur des Computers beinhaltet und z. B. die benötigten Daten stets im richtigen Format anbot. Die *Command Meta Language* selbst wurde in einer weiteren Beschreibung niedergelegt (A), aus der dann mit dem Tree-Meta Compiler, einem Compiler-Compiler für kontextfreie Sprachen (B), der Übersetzer für die CML hergestellt wurde (C). Aus der Beschreibung der Benutzerbefehle (D) konnte dann unmittelbar der Kontrollprozessor (E) erzeugt werden, der ein Teil des NLS-Laufzeitsystems war. Die Verwendung einer solchen Methodik hatte den Vorteil, daß die CML so nicht nur unabhängig von der verwendeten Hardware wurde, sondern auch um neue Sprachkonstrukte erweitert werden konnte, ohne das gesamte System neu übersetzen zu müssen.

¹ Engelbart 1969, S. 5; Engelbart 1968, S. 3.

² Engelbart 1969, S. 7f.

³ Brief von Gerard Guyod, Scientific Data Systems an John H. Wensley, SRI, 6. September 1966, SUL/EC, 6-3.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

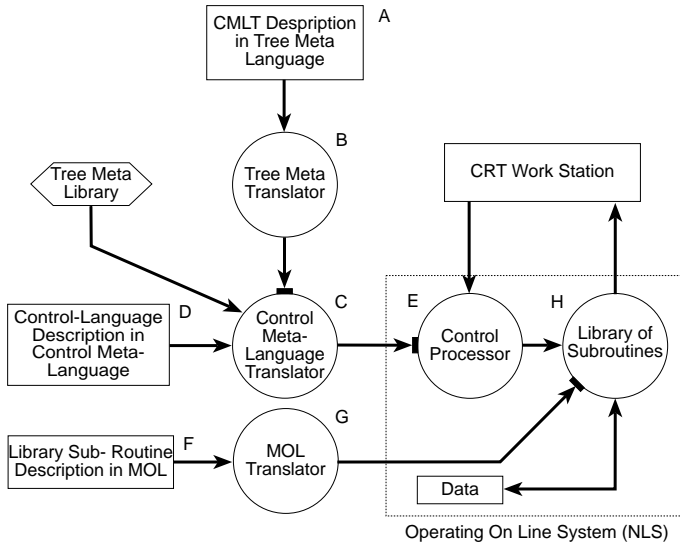


Abbildung 53. Beziehungen zwischen den Systemkomponenten des Online-Systems

Sie war damit ein erstes Ergebnis des *Bootstrapping* zur Unterstützung von Programmierfähigkeiten.¹

Engelbart konnte dabei auf Vorarbeiten zurückgreifen, die seit 1962 an der University of California in Los Angeles, bei SDC, BBN und am SRI durchgeführt worden waren. 1964 hatte Val Schorre erstmals einen Compiler-Compiler vorgestellt², den Engelbarts spätere Mitarbeiter Charles R. Kirkley und Johns F. (Jeff) Rulifson (* 1941) seit 1966 zu dem am ARC verwendeten Tree Meta Compiler weiterentwickelten.³

Für die Definition der eigentlichen Benutzerkommandos wurden die bereits beim Offline-System verwendeten Zustandsdiagramme benutzt, bei denen alle möglichen Systemzustände und Benutzeraktionen in Form eines Netzwerks dargestellt werden können (Abb. 54). Die *Control Meta Language* war ihrerseits so strukturiert, daß eine problemlose Überführung der Definitionen von der einen in die andere Darstellungsform möglich war. Das folgende Programmfragment, das u. a. die Definition des Befehls `Delete Word` enthält, soll dies erläutern. In

¹ Engelbart et al. 1968, S. 4ff.; Engelbart 1969, S. 37ff. – Der Begriff des *Bootstrapping* wird in der Informatik heute vor allem für eine ähnliche Vorgehensweise beim Compilerbau verwendet. Vgl. Wirth 1986a, S. 102ff.

² Schorre 1964.

³ Engelbart et al. 1968, S. D-3ff.; Andrews and Rulifson 1967, S. 104ff.

Abbildung 54 ist der entsprechende Pfad mit den entsprechenden Zahlen (3, 3C, 3C2) beschriftet:

```

3  (wc:) zap case

    3A (b) [edit] dsp(backward ^es*) . case
    ...
    3B (c) [edit] dsp(copy ^es*) :s true => <am> adjl; . case
        3B1 (c) ...
        3B2 (w) ...
        3B3 (l) ...
        3B4 (v) ...
        ...
        3B10 endcase +caqm ;
    3C (d) [edit] dsp(delete ^es*) . case
        3C1 (c) ...
        3C2 (w) s*=dw dsp(^delete word) e*=w,word +buglspec +wdr
            (b1,p1,p2,p3,p4) +del ;
        3C3 ...
    ...

```

Auf der obersten Ebene (3) wird zunächst der gewünschte Befehl durch Eingabe eines ersten Zeichens spezifiziert.¹ Bei Eingabe eines *d* wird beispielsweise zu Statement 3C verzweigt, in dem die Löschbefehle definiert sind. Durch Eingabe eines zweiten Zeichens wird die Art des Operanden näher spezifiziert und im Falle eines *w* (= Delete Word) zu Statement 3C2 weiterverzweigt. Dieses Statement enthält nun die eigentliche Definition des Befehls. Das dort aufgerufene Unterprogramm *+buglspec* wartet auf die Auswahl des zu löschenden Wortes mit Hilfe der Maus, und das Unterprogramm *+wdr* bestimmt den Anfangs- und Endpunkt des zu löschenden Wortes. Nach Betätigung der Taste *CA* (*Command Accept*) wird das Unterprogramm *+del* gestartet, das die eigentliche Löschung vornimmt. Danach kehrt das Programm in den Wartezustand zurück.²

Trotz der eindeutigen, wenn auch nicht unbedingt offensichtlichen Semantik, war die CML nur ein Werkzeug für den geübten Systementwickler. Engelbart betonte dazu immer wieder, man dürfe in der CML kein Mittel zur Dokumentation oder zum Training sehen. Ganz im Sinne seiner Koevolutionsforderung machte erst die Erfahrung des Programmierers aus ihr ein effizientes Werkzeug. Dennoch war Engelbarts Hoffnung, mit Hilfe von Zustandsdiagrammen das komplette Benutzersystem mit allen seinen Eigenschaften gestalten zu können, voreilig. John McCarthy zitierend, mußte er feststellen, die Methode sei zwar »metaphysisch adäquat, nicht aber epistemologisch«, da sich herausstellte, daß das Modell eines

¹ Der Befehl *wc* in Statement 3 steht für »What Case«. Er wird in einer endlosen Schleife durchlaufen und hat keine andere Aufgabe, als auf mögliche Eingaben über die Maus oder Tastatur zu warten. In der heutigen Terminologie würde man vom *Main Event Loop* des NLS sprechen.

² Engelbart 1969, S. 38ff.; Engelbart and English 1968, S. 406.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

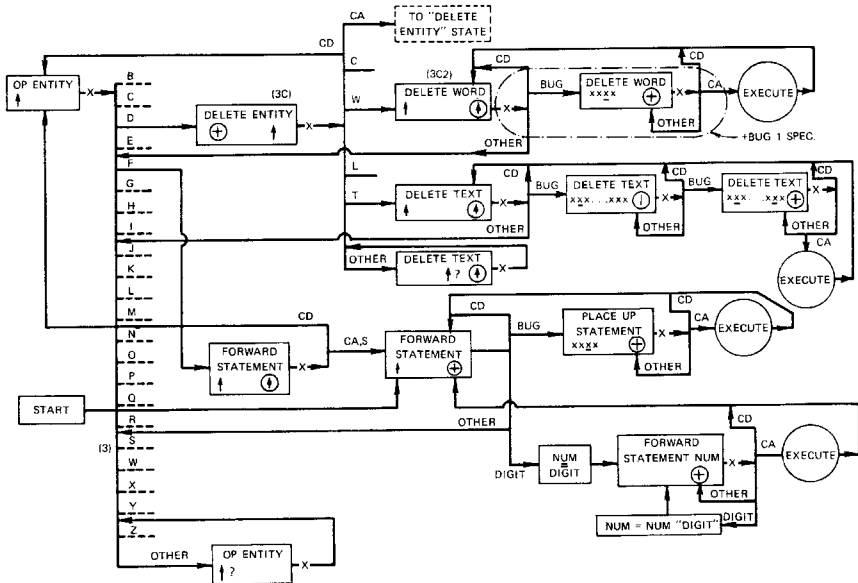


Abbildung 54. Beschreibung eines Teils der Textbearbeitungsbefehle in Form eines Zustandsdiagramms. Es enthält in abgekürzter Form die Beschreibung der relevanten Systemzustände (Rahmen) und möglichen Benutzeraktionen (Pfeile) für die Befehle »Delete Word«, »Delete Text«, »Place Up Statement« und »Forward Statement«.

realen Benutzersystems zu viele mögliche Systemzustände und Zustandsänderungen besaß, um noch analytisch handhabbar zu sein.¹

MOL 940

Da beim Betrieb des NLS kaum Berechnungen durchgeführt werden mußten, bestand das Programm des Kontrollprozessors (Abb. 53, E) fast ausschließlich aus Unterprogrammaufrufen. Die Bibliothek der vom NLS verwendeten Unterprogramme (H) war somit die zweite große Komponente des NLS-Programmsystems. Während der Kontrollprozessor die Schnittstelle zwischen dem Benutzer und dem Online-System definierte, enthielt die Unterprogramm-bibliothek die Anweisungen für die Durchführung bestimmter Aktionen auf der aktuellen Hardware. Ähnlich wie Bush bei seinem Memex wollte Engelbart auf diese Weise die durch die Aufgabe definierte Funktionalität von der tatsächlichen Realisierung auf einem bestimmten Computer trennen.

¹ Engelbart et al. 1968, S. 5.

Während die Trennung in ein sogenanntes Front- und Backend auf dem SDS 940 nur ansatzweise zu erkennen ist, wurde sie in den siebziger Jahren zu einem wichtigen Forschungsschwerpunkt des ARC, als man dazu überging, die beiden Komponenten des NLS auf verschiedenen Rechnern laufen zu lassen, die über ein Computernetzwerk miteinander verbunden waren.¹

Die Routinen der Unterprogrammibibliothek wurden in der *Machine Oriented Language for the 940* (MOL 940) geschrieben, einer von Ray Hay und Jeff Rulifson am ARC speziell für diesen Zweck entwickelten höheren Programmiersprache. Dies war nötig geworden, da sich keine der von SDS mitgelieferten Programmiersprachen für die Systemprogrammierung eignete. MOL 940 vereinte die Syntax und Strukturierbarkeit von ALGOL mit der Möglichkeit, besonders kritische Passagen in Form von Assemblerbefehlen realisieren zu können.² Damit war MOL 940 zwar eine ausgesprochen hardwareabhängige, gleichwohl besonders effiziente Programmiersprache, die auch von anderen Nutzern des SDS-940-Computers außerhalb von Engelbarts Labor sehr geschätzt wurde.³

Diese Werkzeuge für den Systementwickler wurden von Don Andrews, Ray Hay und Jeff Rulifson in der Zeit zwischen der Bestellung und Lieferung des SDS 940, also zwischen Oktober 1966 und Juni 1967 am ARC entwickelt. Dabei nutzten sie einen SDS-930-Computer, der bereits im November 1966 im Laboratorium für angewandte Physik des SRI installiert worden war. Auch die Portierung des existierenden Online-Systems von der Assemblersprache in die *Control Meta Language* und die *Machine Oriented Language* durch William H. Paxton, Margaret Ann Watson und Stephen R. Levine war bis zum Frühsommer 1967 weitgehend abgeschlossen.⁴ In den darauffolgenden zwölf Monaten wurde das NLS von den Mitarbeitern des ARC intensiv genutzt, um dessen Funktionalität im Bootstrapping-Verfahren weiter zu verbessern und erste konkrete Anwendungen zu entwickeln. Dieser Prototyp, dessen Struktur in Abbildung 55 dargestellt ist, wurde dann im Dezember 1968 auf der Herbsttagung der *American Federation of Information Processing Societies* (AFIPS) und im Oktober 1969 auf der Jahrestagung der *American Society for Information Science* einer breiteren Öffentlichkeit präsentiert.

¹ Engelbart et al. 1968, S. 8; Watson 1974, S. 20f.; Engelbart 1988, S. 221ff. – Das Backend lief bei dieser Konstellation gewöhnlich auf einem Großcomputer wie dem PDP-10, während für das Frontend ein kleiner Minicomputer verwendet wurde.

² Hay and Rulifson 1968.

³ Engelbart et al. 1968, S. 9. Das Konzept einer maschinenorientierten Sprache geht nicht allein auf Engelbarts ARC zurück. Auch für den AN/FSQ-32 und das IBM System/360 wurden zur gleichen Zeit MOLs implementiert. Vgl. Engelbart and SRI-ARC Staff 1970, S. 89.

⁴ Engelbart, D. C. and English, W. K., »Computer Augmented Management System«, Monthly Letter Reports No. 8–18, November 1966–September 1967, SUL/EC, 1; »SDS 940 – Internal Memos«, 1967, SUL/EC, 6-4.

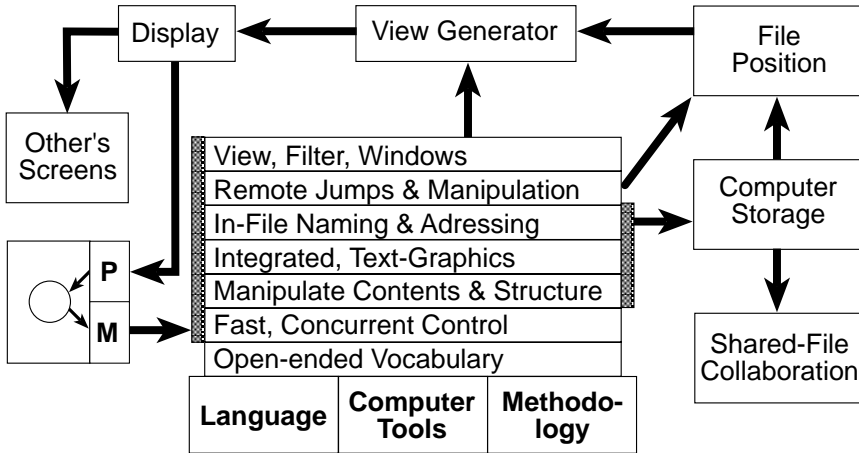


Abbildung 55. Struktur des On-Line Systems (NLS). Der Kasten auf der linken Seite steht für den Benutzer mit seinen perzeptiven (P) und motorischen (M) Fähigkeiten. Der Kreis deutet die menschliche Informationsverarbeitung an.

5.8.4 Mehr als Textverarbeitung

Zur Entstehungszeit des Online-Systems gab es bereits Texteditoren und Formatierprogramme, ohne die man natürlich keine Programme schreiben konnte. Das *Project MAC* am MIT hatte vermutlich die fortschrittlichsten Editier- und Formatierprogramme, die von den Mitgliedern des Labors dazu verwendet wurden, Dokumente aller Art zu verfassen, einschließlich der viele Hundert Seiten umfassenden Jahres- und Forschungsberichte.¹ Aber diese Programme waren schwierig zu bedienen, unter anderem, weil man eine Vielzahl von kryptischen Befehlen und Prozeduren auswendig kennen mußte. Um etwa ein Wort zu löschen, mußte man das entsprechende Kommando (das anders war als das zum Löschen von einzelnen Buchstaben oder ganzen Sätzen) eingeben sowie die Nummer der Zeile und dem Wort selbst. Um Text aus einem anderen Dokument zu kopieren, mußte man das aktuelle Dokument schließen, das neue Dokument öffnen, den gewünschten Text identifizieren, ihn in einem neuen Dokument abspeichern, das erste Dokument wieder öffnen, sich an den Namen des gerade erzeugten Dokuments erinnern und dessen Inhalt mit einem bestimmten Kommando in das aktuelle Dokument übernehmen.

Die umständliche Bedienung solcher Programme läßt sich verstehen, wenn man bedenkt, daß sie in der Regel für die Benutzung an den damals weitverbreite-

¹ Stallman 1981.

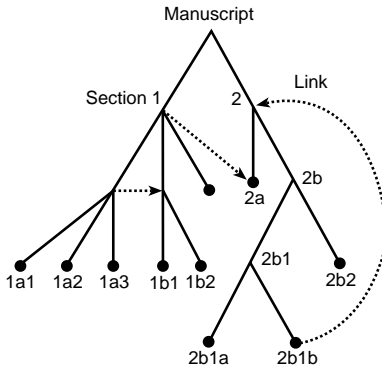


Abbildung 56. Hierarchische Struktur der Statements in einem NLS-Dokument. Jeder Knoten repräsentiert einen Textabschnitt bzw. Statement. Die gepunkteten Verbindungen stehen für optionale Querverweise zwischen einzelnen Statements.

ten Fernschreibterminals entwickelt wurden. Aber auch wenn bereits Bildschirme verwendet wurden, so orientierte sich ihre Nutzung als »gläserner Fernschreiber« am bestehenden Standard.¹

Dokumentenorganisation

Für Engelbart war dies eine immense Unterausnutzung der Fähigkeiten des Computers zur Darstellung und Manipulation von Information. Das NLS sollte viel mehr sein: ein hochentwickeltes »Vehikel zum Bereisen des ›Informationsraumes‹², das das Potential des Videobildschirms ausnutzte und über leicht zu merkende Kommandos zu bedienen war. Grundlage des Systems war die durchgehend hierarchische Organisation aller Dokumente. Dazu verwendete man auf der Seite der Systemsoftware das vom *Project Genie* entwickelte hierarchische Dateiverzeichnis mit Baumstruktur.³ Aber auch jedes Dokument war hierarchisch gegliedert und konnte deshalb genau wie das Dateiverzeichnis durch eine Baumstruktur wiedergegeben werden (Abb. 56).⁴

Statt der Formlosigkeit, die die meisten papiernen Dokumente charakterisierte, bestand ein Text in NLS aus automatisch durchnummerierten »Informationsatomen«, den sogenannten Statements. Die hierarchische Gliederung von schriftlichem Material und die Aufteilung in kleinste Einheiten hatte Engelbart bereits in seinem konzeptionellen Rahmen als wichtige Grundlage eines Intelligenzverstärkers herausgestellt, sie geht aber weiter zurück auf Engelbarts Mikrodokumentationssystem und letztlich auf Bushs Schilderung in »As we may think«. Obwohl die Benutzer in der Regel zunächst reserviert reagierten, gewöhnten sie sich schnell

¹ Deutsch and Lampson 1967; van Dam and Rice 1971; Fischer 1980; Card et al. 1983, S. 107ff.

² Evans 1969, S. 213; Engelbart and SRI-ARC Staff 1969.

³ Engelbart 1969, S. 11.

⁴ English et al. 1965, S. 11; Engelbart 1968, S. 14ff; Rice and van Dam 1972, S. 138ff.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

```
HN↑↑ HOP TO NAME ↑+WAIT
4C (D) [EDIT] DSP(DELETE *ES*) . CASE
4C2 (W) S*=DW DSP(+DELETE WORD) E*= W,WORD+BUG1SPEC
      +WDR (B1,P1,P2,P3,P4) + DEL ;

3B (+BUG1SPEC) BA . CASE
3B1 (CD) GOTO [S]
3B2 (BUG) SPEC(BUG) BP +WAIT RETURN
3B3 ENDCASE GOTO WC ↑

3J (+WAIT) BP . CASE
3J1 (CD) GOTO [S]
3J2 (CA) RETURN
3J3 ENDCASE REPEAT 0(.)

3K (+DEL)
3K1 :C ST B1 <- SF(B1) P3,P4 SE(B1): DELPTR(P1 P2) GOTO [S]
3L (+WDR) (A1, A2, A3, A4, A5) <P> WORD DELIMIT<P>
3L1 :P C(A1) >CH SLD *A3<-A3 (SP *A5 C(A1) <CH SLD *A2<-A2 *A4/
3L2 <P> SAMPLE WORDS
3M (+INSRT)
3N (+INTEGER) NUM*= D CASE
```

Abbildung 57. Teil eines CML Programms, das den Gebrauch von symbolischen Referenzen zeigt, deren Namen in runden Klammern angegeben werden. Die oberen sechs Statements sind »eingefroren«.

an das Arbeiten mit dem sogenannten strukturiertem Text¹, der die Grundlage des Online-Systems darstellte, weil er den Computer in die Lage versetzte, ein Dokument in nahezu beliebiger Weise zu manipulieren.

Eingabe und Modifikation von Text

Was man auf dem Bildschirm sah – der sogenannte *user's view*, heute würde man von der Benutzungsoberfläche sprechen – war übersichtlich und leicht bedienbar (Abb. 57 und 58). An der Oberseite des Bildschirms gab es ein Feld, in dem das aktuelle Kommando im Klartext angezeigt wurde. Erschien zusätzlich ein kleiner Pfeil unter dem Kommando, war das System bereit, den Befehl auszuführen. Links davon gab es ein Feld, in dem die sechs zuletzt eingegebenen Zeichen dargestellt wurden, egal ob es sich dabei um Text oder Befehle handelte. Rechts konnten die sogenannten *view specifications* angezeigt werden, Parameter, über die sich die Bildschirmdarstellung verändern ließ. Darunter befand sich der große Arbeitsbereich, in dem der Inhalt des Dokuments erschien.² Dort konnte man sich auch eine Liste der Dateien ausgeben lassen, die auf der Magnetplatte gespeichert waren. Da diese Dateilisten Dokumente wie alle anderen waren, konnten sie in gleicher Weise betrachtet und manipuliert werden.

Die Eingabe und Modifikation des Textes erfolgte durch die Eingabe von Kommandos, die aus jeweils zwei Buchstaben bestanden.³ Dabei gab der erste Buchstabe die durchzuführende Operation und der zweite Buchstabe den Typ des Operanden an. Um ein neues Dokument zu erzeugen, mußte man eine neue Datei erzeugen und konnte dann nach Eingabe von IS für »Insert Statement« den Text

¹ Engelbart 1969, S. 48ff.; Engelbart and English 1968, S. 398f.

² English et al. 1965, S. 14f.

³ Eine Übersicht der wichtigsten NLS-Befehle und ihrer Funktionalität ist in Anhang 8 zu finden.

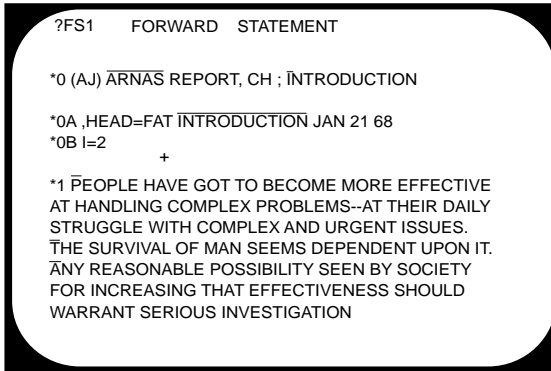


Abbildung 58. Teil eines Textdokuments. Großbuchstaben sind überstrichen. Die Statements 0A und 0B enthalten zusätzliche Informationen über die Datei und den darin gespeicherten Text.

eingeben. Obwohl das System hauptsächlich für den Umgang mit Textdokumenten ausgelegt war, konnte ein Statement auch Grafiken aus Linien und einfachen geometrischen Formen enthalten, die man über spezielle Grafikroutinen erzeugen konnte.¹

Für die Modifikation von Text war der für das Offline-Text-System entwickelte Befehlssatz übernommen und erweitert worden. Um etwa ein Wort zu löschen, gab man – vorzugsweise über die Einhandtastatur – die Buchstabenfolge DW ein, in der Kommandozeile erschien dann im Klartext »Delete Word«. Dann bewegte man den *Bug* mit der Maus zum gewünschten Operanden, wählte diesen durch Betätigen der Maustaste aus, und unter dem gewünschten Wort erschien als Bestätigung ein Pfeil. Durch nochmaliges Drücken der Maustaste wurde das gewünschte Wort dann gelöscht.

Wählte man einen Operanden aus, auf den die Operation nicht angewendet werden konnte, erschien statt des Pfeils ein Pluszeichen, und man konnte einen neuen Operanden auswählen oder den Befehl abbrechen.² Keine der so durchgeführten Operationen war endgültig, sie ließen sich alle wieder rückgängig machen. Falls bei der Befehlseingabe Probleme auftraten, konnte man schließlich eine umfangreiche Online-Hilfe konsultieren.³

Dokumentenstudium

Die wahrscheinlich eindrucksvollsten Aspekte von Engelbarts System waren die Möglichkeiten zum Dokumentenstudium, die Verfahren zur Veränderung der Bildschirmdarstellung, zur Navigation innerhalb des Dokuments und zur automatischen Modifikation umfaßten.

¹ Evans 1969, S. 219.

² English et al. 1965, S. 14f.; Engelbart 1968, S. 37.

³ Engelbart and SRI-ARC Staff 1968.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

Der Benutzer konnte beispielsweise *Viewing Specifications*, kurz ViewSpecs, angeben und damit festlegen, wie ihm ein Dokument am Bildschirm präsentiert wurde. Er hatte etwa die Möglichkeit, die Numerierung der Statements zu unterdrücken oder die Zeichengröße und damit die Anzahl der darstellbaren Zeilen zu variieren. Er konnte auch präzise angeben, was man von einem Dokument sehen wollte und konnte z. B. festlegen, daß nur alle Statements der obersten Hierarchieebene angezeigt werden sollten, um einen schnellen Überblick über die Struktur des Dokuments zu erhalten. Um mehr zu sehen, konnte man die unteren Ebenen einblenden lassen oder auch nur die ersten Zeilen eines jeden Statements.

Für die Navigation standen eine Vielzahl von Befehlen zur Verfügung, mit denen man sich innerhalb der Hierarchie bewegen konnte. Zum letzten Statement der aktuellen Ebene etwa oder zum Beginn der nächsthöheren bzw. nächstniedrigeren Stufe. Man konnte aber auch beliebige Punkte des Dokuments ansteuern. Dazu mußte bei dem entsprechenden Befehl nur die Nummer des gewünschten Statements angegeben werden (Abb. 56). Es bestand aber auch die Möglichkeit, Statements einen Namen zu geben und dann diesen Namen anzusteuern.¹

Damit konnte man nun realisieren, wovon Bush in »As we may think« nur träumen konnte, nämlich assoziative Verbindungen zwischen verwandten Informationen angeben und Argumentationspfade aufbauen. Dazu wurden die Referenzen so in andere Statements eingebettet, daß man sie mit der Maus anklicken und dann ausführen konnte. In einem typischen Text konnte z. B. der Literaturverweis »see (Engelbart and Huddard, 1965)« erscheinen, über den man direkt zum Literaturverzeichnis des Dokuments gelangen konnte. Die Funktion beschränkte sich allerdings bald nicht nur auf Verweise innerhalb eines Dokuments, sondern ließ auch Verweise zu anderen Dokumenten, auch solchen von anderen Autoren zu. Diese Möglichkeit von NLS entwickelte sich bald zur Grundlage für das kooperative Arbeiten und zur Schaffung einer *Bootstrapping Community*, einer virtuellen Gemeinschaft von Computerbenutzern.² Diese Form der nichtlinearen Strukturierung von Dokumenten, die Engelbart in seinen damaligen Veröffentlichungen nicht übermäßig stark herausstellte, war bereits 1965 von Ted Nelson als »Hypertext« bezeichnet worden³, und wird heute häufig als wichtigste Entwicklung Engelbarts neben der Maus dargestellt.⁴

Außerdem besaß das NLS einen sogenannten *pattern matcher*, mit dem man alle Dokumente nach Begriffen durchsuchen und aus den gefundenen Statements

¹ Engelbart 1968, S. 16ff.; Engelbart 1969, S. 18ff. – In Abbildung 57 hat das Unterprogramm in Statement 3B den Namen +BUG1SPEC, die Überschrift des Dokuments in Abbildung 58 hat den Namen AJ.

² Engelbart 1969, S. 56ff.; Memorandum von Stephen R. Levine und Margaret Ann Watson über »New Commands in the 3100 NLTS: >HOP TO FILE<«, 21. Juni 1967, SUL/EC, 17-2.

³ Nelson 1967.

⁴ Conklin 1987; Nielsen 1993.

ein neues Dokument erstellen lassen konnte. Schließlich konnten einzelne Statements auf dem Bildschirm »eingefroren« werden, während auf dem Rest des Bildschirms ein anderer Teil des Dokuments angezeigt wurde (Abb. 57). Die eingefrorenen Statements wurden seit Ende der sechziger Jahre durch die Verwendung von Fenstern ergänzt, d. h. der Bildschirm konnte in bis zu acht unterschiedlich große, sich allerdings nicht überlappende Rechtecke aufgeteilt werden, in denen jeweils ein anderer Dokumentenausschnitt oder ein anderes Dokument dargestellt werden konnte.¹

Es gab weiterhin Programmfunktionen, die die Dokumente speziell für die Ausgabe auf dem Zeilen- oder Typenraddrucker, später auch auf einem Mikrofilmbelichter, formatierten. Die zunächst nur rudimentären Fähigkeiten zur Formatierung wurden im Laufe der siebziger Jahre, parallel zur Weiterentwicklung der Druckertechnik, zu einem eigenständigen *Output Processor* ausgebaut, der die Einbindung von Grafiken in (gedruckte) Dokumente unterstützte und eine ganze Bibliothek von Formatvorlagen enthielt.²

Aus heutiger Sicht betrachtet umfaßte das NLS damit fast alles, was auch ein modernes Textverarbeitungssystem beinhaltet.³ Es darf aber nicht übersehen werden, daß NLS durch seinen experimentellen Charakter und die weit über die reine Textverarbeitung hinausgehenden Ziele Engelbarts kein unmittelbarer Vorläufer heutiger Programme ist. Es ist vielmehr so, daß sich eine spezifische Arbeitspraxis erst im täglichen Umgang mit dem Online-System herausbildete, also durch *Bootstrapping*.⁴

5.8.5 Arbeiten mit dem NLS

Konkret bedeutete dies, daß man so schnell wie möglich allen Mitarbeitern des ARC, von der Sekretärin bis zum Programmierer, die Möglichkeit geben mußte, einen nennenswerten Anteil ihrer Arbeiten direkt am Computer durchführen zu können. Nach der Installation des SDS-940-Computers und der Portierung der NLS-Software war dieser Zustand im Herbst 1967 erreicht, wenngleich sich die etwa zehn Mitarbeiter des ARC die zunächst sechs Terminals während der noch eingeschränkten Betriebszeit (20 Stunden pro Woche) teilen mußten.⁵

¹ Engelbart 1969, S. 20ff.; Memorandum von Margaret Ann Watson über »The PATTERN MATCHER – Summary of the 22 May 67 Bootstrap meeting explanation«, 1967, SUL/EC, 16-2; Augmentation Research Center 1975, S. 42, 169; Irby 1974; Irby, C. H., »Window user interface documentation«, 28. Januar 1976. Stanford University Libraries, Charles H. Irby Papers, Box 6, Folder 7 (künftig SUL/CHIP, Box-Folder).

² »Output Processor Users' Guide«, 29. Juli 1975, Journal # 32 812, SUL/CHIP, 6-6.

³ Barnes 1997.

⁴ Hofmann 1996.

⁵ Engelbart 1969, S. 45f.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

Obwohl keiner der Pilotanwender Erfahrungen mit der Arbeit am Computer besaß und sich das Online-System noch im Entwicklungsstadium befand, stellten sich keine gravierenderen Probleme ein. Insbesondere kam es nicht zu den besonders gefürchteten Datenverlusten durch Systemabstürze. Es stellte sich vielmehr heraus, daß die tatsächlich vorkommenden sporadischen Verluste eher durch eine unsachgemäße Bedienung auftraten. Grund hierfür war die hohe Geschwindigkeit, mit der sich Dateien erzeugen, ändern oder löschen ließen, und so wurden gelegentlich Dateien verändert, bevor sich der Benutzer über die Tragweite seiner Eingaben vollends bewußt war. Nach einiger Zeit legten sich die meisten Nutzer allerdings Schutzmechanismen zu und erstellten Sicherungskopien der zu bearbeitenden Dokumente.¹

Auch der sehr umfangreiche und teilweise redundante Befehlssatz war im praktischen Gebrauch vorteilhaft, weil er die Herausbildung unterschiedlicher Nutzungsweisen ermöglichte. Bei den Beobachtungen stellte sich nämlich heraus, daß es individuelle Unterschiede bei der Beherrschung der Systembefehle gab. Jeder Nutzer beherrschte einen unterschiedlich umfangreichen Teil der möglichen Befehle und führte sie mit seiner individuellen Geschwindigkeit aus, ohne daß dies zu signifikanten Unterschieden beim Arbeitsergebnis führte. Engelbart fühlte sich durch diese empirischen Ergebnisse in der Hoffnung bestärkt, mit Hilfe des NLS ein besseres Verständnis für die Gesetze der Mensch-Computer-Interaktion zu erhalten, auf deren Basis zukünftige Systeme entworfen werden könnten.²

Vor allem aber ergab sich die methodische Frage, welche Arbeiten man während der beschränkten Online-Arbeitszeit durchführen sollte, und anhand welcher Kriterien diese dann bewertet werden konnten. Es stellte sich heraus, daß trotz einer anfänglichen Belastung der Mitarbeiter eine erhebliche Akzeptanz der Online-Arbeit eintrat und daß schon bald versucht wurde, *alle* anfallenden Arbeiten mit dem NLS zu bewältigen. Für die Bewertung wurde in Ermangelung einer theoretisch fundierten Methodik eine pragmatische Mischung aus subjektiver Bewertung und automatisierter Messung, z. B. von Antwort- oder Bedenkzeiten, angewendet. Die dabei erhaltenen Ergebnisse wurden dann wieder dazu verwendet, um die Möglichkeiten zur Unterstützung bestimmter Arbeitsabläufe weiter zu optimieren.³

Deshalb berichtete Engelbart in einem Abschlußbericht an die NASA über seine *persönlichen* Erfahrungen bei der Arbeit mit dem Online-System. Dabei

¹ Engelbart 1969, S. 52.

² Engelbart 1969, S. 57ff.

³ Engelbart 1969, S. 46ff.; Engelbart and SRI-ARC Staff 1970, S. 94ff. Noch heute wird bei der Bewertung von Mensch-Maschine-Systemen in der Regel ein Mix von subjektiven und objektiven Verfahren angewendet. Vgl. Johannsen 1993, S. 189ff.

stellte er fest, daß es mit NLS sehr viel einfacher wurde, sich während der Arbeit Notizen zu machen und diese später zu einem größeren Ganzen, etwa einem Forschungsbericht, zusammenzufügen. In ähnlicher Weise wurde das NLS während interner Arbeitstreffen als Kommunikationsmedium verwendet, mit dem alle aufkommenden Ideen protokolliert und strukturiert werden konnten, so daß die Ideen am Ende nicht nur diffus »im Kopf hängen«, sondern in einer für jedermann abrufbaren Form als NLS-Dokument existierten. Damit wurde NLS zu einem adäquaten Werkzeug zur Unterstützung von Arbeitsgruppen und von Brainstorming-Methoden zur Problemlösung.¹

In bezug auf seine individuelle Arbeit beim Erstellen von Texten beobachtete Engelbart, daß er mit dem NLS wesentlich freier und schneller schrieb als mit den überkommenen Verfahren. Weil auf dem Bildschirm nichts endgültig sei und alles wieder geändert werden könne, habe er weniger Hemmungen, Worte zu produzieren und Formulierungen auszuprobieren, Fehler zu verbessern und Gedankengänge nur zum Test neu zu strukturieren. Das alles, so schien es ihm, verbesserte das Bewußtsein des NLS-Benutzers für sein Material und dessen innere Struktur und regt die Experimentierfreudigkeit an. Dabei darf natürlich nicht unerwähnt bleiben, daß auf diese Weise auch völlig überflüssige Informationen produziert und gespeichert wurden, die bei herkömmlicher Arbeitsweise bereits in vorgelagerten Arbeitsschritten ausgefiltert worden wären. Folge des kreativeren und produktiveren Arbeitens war eine insgesamt höhere Zufriedenheit mit den Resultaten der Arbeit.² Dies bedeute aber nichts anderes, als daß NLS tatsächlich Engelbarts Intelligenz verstärken konnte:

»Important to me is that this increased ›practice‹ with formulating thoughts and relationships, different in nature from and greater in every-hour quantity than anything I have experienced before, gives me a feeling that it really will increase the growth rate of my intellectual capability.«³

Auch für externe Beobachter, die keinen Einblick in die theoretischen Grundlagen von Engelbarts Arbeit besaßen, bekam der Computer nun medialen Charakter. Der Journalist Nilo Lindgren prägte für den im Computer gespeicherten Datenbestand mit seinen expliziten und impliziten Querverweisen den Begriff der »Informationslandschaft«, die mit Hilfe eines Fahrzeuges, eben des NLS, bereist werden kann. Dabei sei völlig unerheblich, wie und wo die Daten physisch lokalisiert sind oder wie das Fahrzeug intern funktioniert. Wichtig sei nur, daß es

¹ Engelbart 1969, S. 48ff.; Engelbart and SRI-ARC Staff 1970, S. 36f.

² Engelbart 1969, S. 50ff.

³ Engelbart 1969, S. 52. Im Original ist das Zitat in drei Unterstatements aufgeteilt, die hier nicht übernommen wurden.

ein effektives und zuverlässiges Werkzeug zur Erschließung von Information darstellt. Damit prägte er für Engelbarts System – vermutlich unbewußt – das gleiche Bild, das Vannevar Bush bereits Jahre zuvor für den Memex und seinen Benutzer, den Mathematiker, herangezogen hatte (vgl. S. 55).¹ Die technische Realisierung des Artefakts muß vollständig hinter der Anwendung verschwinden, während eine sorgfältig gestaltete Schnittstelle zwischen Artefakt und Anwendung vermittelte. David Evans' Bemerkung, »NLS sei ein hochentwickeltes System zur Unterstützung von Monologen« deutet aber auch an, daß es Ende der sechziger Jahre noch weit davon entfernt war, *alle* Hoffnungen zu erfüllen, die Engelbart und seine Mitarbeiter in den Intelligenzverstärker setzten.²

Speziell bliebe zu diskutieren, inwieweit die Verwendung von explizit hierarchisch organisierten Dokumenten tatsächlich intellektuellen Tätigkeiten angemessen ist. Selbst wenn man annimmt, daß die menschlichen Fähigkeiten hierarchisch aufgebaut sind und ein technisches System analog gestaltet sein sollte, so bleibt es zweifelhaft, ob ein Mensch-Computer-System ebenso organisiert sein muß. Es ist durchaus möglich, daß das Arbeiten mit dem NLS eine ganz neue Art zu denken und argumentieren hervorgebracht hat, die sich mehr an den Möglichkeiten der Technik orientierte als an den Bedürfnissen des Menschen. Es darf bezweifelt werden, ob Engelbart dies mit der Koevolution von Mensch und Technik innerhalb eines symbiotischen Mensch-Computer-Systems gemeint hat.³

5.8.6 San Francisco, 9. Dezember 1968: Douglas Engelbarts große Show

Zu Beginn des Jahres 1968 hatte Engelbart jedenfalls das Gefühl, ein System zu besitzen, das man einer größeren Öffentlichkeit vorführen könnte, um ihr die Mission des ARC nahezubringen. Deshalb entschloß er sich, sein Online-System auf der Herbsttagung der AFIPS vorzustellen, die im Dezember 1968 in San Francisco stattfinden sollte. Wochenlang legten Engelbart und seine Belegschaft ihre normale Arbeit zu Seite und bereiteten ihre große Show vor.

Da es nicht möglich war, das komplette Computersystem nach San Francisco zu transportieren, stellte Bill English mit einem Team über gemietete Telefonleitungen und Richtfunkstrecken die Verbindung zum 50 km entfernten Computer in Menlo Park her. Auf der Bühne des Auditoriums in der Brooks Hall (heute Civic Center) bauten sie die extra für diesen Zweck entworfene Herman-Miller-Workstation auf, an der Engelbart während seiner Präsentation saß. Mit

¹ Lindgren 1971, S. 53; Licklider et al. 1968. Engelbart hat dieses Bild später selbst gerne verwendet. Vgl. Engelbart et al. 1973, S. 13.

² Evans 1969, S. 152, 213.

³ Engelbart 1969, S. 53ff.

Hilfe eines Videoprojektors wurde das Geschehen auf der Bühne und dem Bildschirm auf eine sechs Meter hohe Leinwand projiziert. Auch in Menlo Park wurde eine Workstation mit Videokamera eingerichtet, die von den Wissenschaftlern des Labors verwendet werden sollte. Alle Videosignale, inklusive derjenigen von den Computerbildschirmen, liefen bei einem Steuerpult hinter der Bühne zusammen, wo Bill English Regie führte. Er konnte zwischen den unterschiedlichen Computer- und Videosignalen hin- und herschalten, sie überlagern, und den Bildschirm in mehrere Fenster unterteilen. Allein dies war bereits eine außergewöhnliche technische Leistung, die allerdings mehr als 10 000 \$ kostete.¹ Engelbart und English gingen für ihre effektvolle Präsentation ein gewisses Risiko ein, handelte es sich doch um die zweckwidrige Verwendung staatlicher Fördermittel. In einem Interview erinnerte sich Engelbart 1987:

»I was pretty sure that we were getting money from NASA and ARPA. . . . How much should I tell them? I got far enough so that they got the idea of what I was trying to do and they were essentially telling me, »Maybe it's better that you don't tell us.« . . . We had a lot of research money going into it and I knew that if it really crashed or if somebody really complained, there could be enough trouble that it could blow the whole program; they would have to cut me off and black ball us because we had misused government research money.«²

Die Vorführung, die dann am 9. Dezember 1968 vor einem Publikum von etwa 800 Personen stattfand³, war eine technische und inszenatorische Glanzleistung. In neunzig Minuten präsentierte Engelbart alle wichtigen Funktionen des NLS. Er zeigte dem Publikum die Maus und die Einhandtastatur und erklärte, wie sie funktionieren und wie man sie benutzt, wie man Dateien anlegt, Text editiert, wie man Querverweise erzeugt und wie man in einem Dokument sucht. Schließlich führte er vor, wie man das NLS dazu verwenden konnte, mit einer entfernten Person zusammenzuarbeiten. Teile der Präsentation wurden nämlich nicht von Engelbart selbst, sondern von Bill Paxton und Jeff Rulifson in Menlo Park durchgeführt. Über einen Befehl konnte Engelbart die Kontrolle über den Computer an seine Mitarbeiter übergeben, und das staunende Publikum erlebte, wie das Dokument, das auf dem Monitor (und auf der Leinwand) in San Francisco angezeigt wurde, von einem 50 km entfernten Terminal in Echtzeit bearbeitet werden konnte.⁴

¹ Engelbart 1988, S. 202ff.; Engelbart 1986/87, Interview 3, Tape 2, Side A; Engelbart et al. 1969, S. 53ff.

² Engelbart 1986/87, Interview 3, Tape 2, Side A.

³ Spätere Darstellungen sprechen von 2 000 oder gar 3 000 Anwesenden. Es scheint auch, als würden sich viele der nachweislich nicht anwesenden Personen heute gern an Engelbarts Präsentation erinnern. Vgl. Augarten 1995 und Engelbart 1986/87, Interview 3, Tape 2, Side A.

⁴ Engelbart and English 1968; Engelbart and SRI-ARC Staff 1968.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

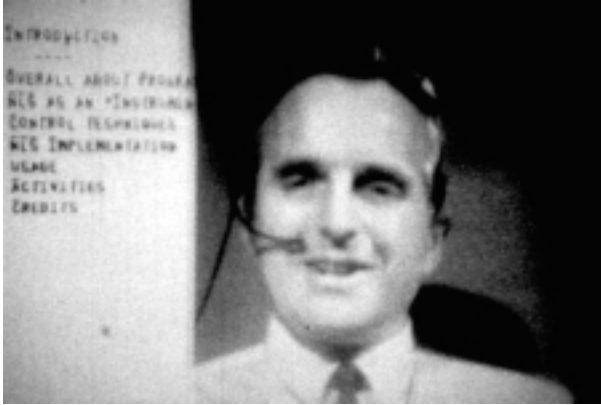


Abbildung 59. Ein Bildschirmfoto von Engelbarts Präsentation auf der Fall Joint Computer Conference am 9. Dezember 1968. Auf der linken Seite sind die Ausgaben des Computers, eine Gliederung von Engelbarts Vortrag zu sehen.

Computervorfürungen sind normalerweise wenig spannend. Jemand sitzt vor einem Bildschirm, tippt etwas über die Tastatur ein, spielt mit der Maus herum und versucht gleichzeitig zu erklären, was passiert. Normalerweise geht dabei etwas schief, im besten Fall nichts schlimmes und die Demonstration kann weitergehen. Engelbarts Vorführung war anders. Es gelang ihm in bemerkenswerter Weise, seine Zuhörer zu fesseln, die zwar selbst Computerfachleute waren, aber zum größten Teil in der Welt der konventionellen Datenverarbeitung lebten und noch nie interaktiv mit dem Computer gearbeitet hatten. Er gab ihnen einen Eindruck davon, wie der Computer zu einem Werkzeug sowohl für alltägliche als auch berufliche Aufgaben werden konnte. So verdeutlichte er die Editierfunktionen – Einfügen, Kopieren, Sortieren und Gliedern – nicht an einem wissenschaftlichen Beispiel, sondern an einer Einkaufsliste für Lebensmittel. Andererseits präsentierten Engelbart und seine Mitarbeiter auch das enorme Potential, das NLS bei der gemeinschaftlichen Entwicklung von Software in einer strukturierten Programmiersprache bot.¹ Es war für die meisten der Anwesenden ein kurzer und faszinierender Einblick in eine völlig fremde Welt.

Es war »eines der eindrucksvollsten Dinge, die ich je in meinem Leben gesehen habe«, sagte Charles H. Irby, ein damals 23jähriger Programmierer. »Die Leute waren verzaubert. ... Niemand hatte so etwas jemals zuvor gesehen ... Es liefen einem Schauer den Rücken hinunter. Das Publikum war total gefesselt.«² Und nach dem Ende der Demonstration bekam Engelbart, ein Wissenschaftler, von dem bislang kaum einer der Anwesenden gehört hatte, stehende Ovationen. Eine ähnlich elektrisierende Wirkung wie auf Irby, der kurz darauf eine Stelle

¹ Engelbart and English 1968; Engelbart and SRI-ARC Staff 1968.

² Charles H. Irby, zitiert in Augarten 1995.

am ARC annahm, hatte die Präsentation auch auf andere junge Wissenschaftler wie Butler Lampson oder Alan Kay, die sich davon für ihre eigenen Forschungen inspirieren ließen (vgl. Kapitel 6).¹

Wie wenig Verständnis aber der durchschnittliche Computerwissenschaftler für Engelbarts Ziele aufbringen konnte, zeigt die überlieferte Reaktion eines Konferenzteilnehmers. Nachdem sich dieser von Engelbart die Funktionsweise des Systems hatte erklären lassen, fragte er ein wenig enttäuscht: »You mean it doesn't run through and answers questions and things like that?«² Hier wird offensichtlich, welche Diskrepanz zwischen der Spitze der Wissenschaft und der alltäglichen Computerpraxis existierten. Computerbetrieb hieß eben immer noch in der überwältigenden Zahl der Fälle Stapelbetrieb, Lohnabrechnungen und numerische Mathematik. Gleichzeitig hatten die KI-Wissenschaftler enorme Hoffnungen geschürt, die Engelbart mit seinem System nicht erfüllen konnte und nicht erfüllen wollte. Auf der anderen Seite wurden Ende der sechziger Jahre auch andere Programme eingeführt, mit denen sich in eng begrenzten Bereichen die Arbeit des Wissenschaftlers und Ingenieurs effektiver gestalten ließ. Diese Programme, darunter vor allem solche für *Computer Aided Design* (CAD) und *Computer Aided Manufacturing* (CAM), sollten zwar nicht die ambitionierten Ziele Engelbarts erfüllen, sie wurden aber sehr schnell zu selbstverständlichen Werkzeugen im Alltag von Ingenieuren und Informatikern. Damit begannen auch Engelbarts Ideen in den Augen mancher Wissenschaftler zu einem alten Hut zu werden.³

Es ist deshalb auch nicht verwunderlich, daß sich nach Engelbarts Präsentation die großen Computerhersteller wie IBM oder DEC nicht danach drängten, mit dem ARC zusammenzuarbeiten. Die Computerindustrie fuhr fort, die Produkte herzustellen, die sie schon immer hergestellt hatte, hauptsächlich Systeme für die Stapelverarbeitung und Time-Sharing-Computer mit Fernschreibterminals. Niemand trat an SRI oder Engelbart heran und bot die kommerzielle Auswertung des Online-Systems an oder wollte Engelbart als Berater engagieren. Lediglich einige Forschungslaboratorien (z. B. die Bell Laboratorien) interessierten sich für die Verwendung der Maus.⁴ Dieser Umstand ist wiederum auch auf Engelbarts Unfähigkeit zurückzuführen, sich und sein Forschungsprogramm effektiv zu verkaufen und dabei Kompromisse mit den technischen und wirtschaftlichen Erfordernissen potentieller Interessenten einzugehen.

Auch auf wissenschaftlicher Seite sah die Rezeption von Engelbarts Arbeit nach dem Dezember 1968 nicht viel anders aus als zuvor. Obwohl über Engelbarts Arbeit 1967/68 in einigen leserstarken wissenschaftlichen Zeitschriften sehr

¹ Lampson 1988, S. 293f.; Kay 1996, S. 518f.

² Zitiert in Lindgren 1971, S. 55.

³ Lindgren 1971, S. 56; Coons 1966b; Coons 1966a; Rheingold 1985, S. 202f.

⁴ Miller 1971; Ball and Hall 1970; Licklider and Vezza 1978.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

positiv berichtet¹ und unmittelbar nach der Präsentation in San Francisco auch in der populärwissenschaftlichen Presse ausführlich und geradezu euphorisch über Engelbart geschrieben worden war², setzte kein formeller Wissens- oder Technologietransfer ein. Das Interesse an NLS und Engelbarts Ideen über die zukünftige Nutzung des Computers blieben – so zeigt der Blick in den *Science Citation Index* – an das organisatorische und kulturelle Umfeld der ARPA und seiner Vertragspartner beschränkt.³ Erst sehr viel später, Ende der siebziger Jahre, begann man sich wieder für seine Arbeiten zu interessieren, diesmal mit Blickrichtung auf die sogenannte *Office Automation*.⁴ Engelbart mußte sich also weiterhin damit begnügen, seine Arbeit in der – finanziell gesehen sehr komfortablen – Nische der militärischen Forschungsförderung fortzuführen.

5.8.7 Erste Anwendungen

Spätestens seit 1966 entwickelten Engelbart und seine Mitarbeiter auch Anwendungsprogramme, die – auf höherem Niveau als das Basissystem – zur Unterstützung von Managementaufgaben dienen sollten. Dabei standen vor allem die Bedürfnisse des eigenen Labors, zunehmend aber auch diejenigen von Engelbarts Geldgebern im Vordergrund. Es handelte sich dabei meist um große militärische Forschungslaboratorien (z. B. das RADC) oder logistische Einheiten wie das *Army Materiel Command*, die mit großen Informationsmengen umgehen mußten, wie sie bei der Planung und Koordinierung von militärischen Entwicklungsprojekten anfallen. Vor diesem Hintergrund waren seit dem Zweiten Weltkrieg bereits eine Vielzahl von Planungsmethoden wie die lineare Programmierung oder die Netzplantechnik entstanden.⁵ Die am ARC entwickelten Programme kamen bei diesen Institutionen allerdings erst zu Beginn der siebziger Jahre zum Einsatz, als es über das ARPANET möglich war, sich auch von entfernten Orten in den Computer des ARC einzuloggen.⁶ Entwickelt wurden u. a. Werkzeuge zur Kosten- und Personalplanung, zur Überwachung von Projekten und zur Unterstützung von Arbeitsgruppen, wobei keines der Werkzeuge als eigenständiges Programm, sondern

¹ Coons 1966b; Brodey and Lindgren 1967; Brodey and Lindgren 1968; Anonymous 1968.

² Haavind 1969; Field 1969; Lindgren 1971.

³ Selbst innerhalb der ARPA wurden Engelbarts Arbeiten nur sehr beschränkt rezipiert. So ist in den umfangreichen Interviews, die Aspray, Norberg und O'Neill für ihre Studie über die Geschichte der ARPA mit den Beteiligten führten, von Engelbart allenfalls am Rande die Rede. In ihrem Buch (Norberg et al. 1996) wird auf seine Arbeiten nicht eingegangen.

⁴ Beispielsweise Judge 1977; Bamford 1979; Meyer 1983; Cole 1983; Rice 1987.

⁵ Lindner et al. 1984, S. 153ff.

⁶ Duane L. Stone, »Augment's Support of Organizations: A Brief History«, Augment Document #132 615, Bootstrap Institute, May 1991.

Period	Salary	Burden	Ovhd	NonLabr	TotCost	Fee	TotChge
1	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-
3	676	128	773	23062	24639	739	25378
4	1080	205	1234	18061	20580	617	21198
5	1016	193	1160	23876	26245	787	27032
6	1167	221	1334	29655	32377	971	33348
7	2499	474	2854	24434	30261	907	31169
8	2522	479	2881	25003	30885	926	31811
9	3092	587	3532	29663	36874	1106	37980
10	4787	909	5468	25978	37142	1114	38257
11	2736	520	3126	40168	46550	1396	47946
12	2309	438	2638	8782	14167	425	14592
13	4466	848	5102	67147	77563	2326	79890

Abbildung 60. Ausschnitt aus der Datei HISCO

stets als Komponente des Online-Systems realisiert wurde, das auf dem *gemeinsamen* Datenbestand operierte.¹

Eines der ersten Programme dieser Art diente zur Überwachung der finanziellen Ausgaben des ARC. Das Programm HISCO (für *Cost History*) enthielt für jedes SRI Projekt ein eigenes Statement, in das die Daten aus den internen Zeiträumen der einzelnen Mitarbeiter und die unmittelbar dem Projekt zugeordneten Sachkosten und Gebühren eingetragen wurden. Aus diesen Rohdaten konnten dann mit einem Rechenprogramm, das Bestandteil des Betriebssystems war, verschiedene Kostenarten wie Gehaltskosten, Verwaltungskosten, Gebühren und schließlich die Gesamtkosten ermittelt werden. Das Ergebnis der Analyse wurde dann in Form einer Tabelle dargestellt, die dem Manager einen bequemen Überblick über die für seine Entscheidungen wichtigen Informationen lieferte (Abb. 60). Mit Hilfe der Darstellungsoptionen, speziell der Funktionen zum Ein- bzw. Ausblenden von ganzen Hierarchieebenen, war es schließlich möglich, den Detailliertheitsgrad der angezeigten Information an die aktuelle Aufgabe anzupassen.²

Ein weiteres Managementwerkzeug, das sogenannte *Baseline Record System*, wurde für die Personaleinsatzplanung entwickelt. Durch den Einsatz des mächtigen *Pattern Matcher* zum Durchsuchen von Datenbeständen konnte auf der gleichen Datenbasis wie bei der Kostenkontrolle ermittelt werden, mit welchen Aufgaben ein Mitarbeiter augenblicklich beschäftigt war oder welchen Bearbeitungsstand bestimmte Projekte augenblicklich hatten. Neben der reinen Auflistung wurde auch die Wichtigkeit der einzelnen Aufgaben, die Beteiligung anderer Mitarbeiter und die zur Fertigstellung noch benötigten Ressourcen angezeigt

¹ Engelbart and SRI-ARC Staff 1970, S. 5ff.; Lindgren 1971, S. 57.

² Engelbart and SRI-ARC Staff 1970, S. 6ff. Anders als bei den heutigen Tabellenkalkulationsprogrammen wurden die Kosten bei Änderungen der Eingangsdaten nicht *automatisch* neu berechnet.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

(Abb. 61). Mit diesem Werkzeug konnte man umfangreiche Projekte koordinieren, knappe Personal- oder Sachressourcen effizient verwalten und die Einhaltung von Terminen verbessern.¹

5.9 Vom Individuum zur Gruppe: Die *Online Community* und das ARPANET

All diese frühen Managementwerkzeuge waren, ebenso wie das gesamte NLS, auf die Unterstützung eines einzelnen, isoliert arbeitenden Informationsarbeiters ausgelegt. So hilfreich sie auch für die praktische Arbeit in einem großen Labor wie dem ARC waren, entsprachen sie doch nicht ganz Engelbarts Vorstellungen. Bereits in seinem »Conceptual Framework« hatte Engelbart auf die große Rolle von Arbeitsgruppen in Wissenschaft, Technik und Wirtschaft hingewiesen. Eine zu schaffende *Online Community* eröffnete auch in Hinblick auf das *Bootstrapping* neue Perspektiven, denn »if we wanted to pursue significant augmentation results, it was absolutely necessary to have a community of real-world users.«² Ganz in diesem Sinne forcierte Engelbart zwischen 1970 und 1972 eine Neuorientierung der Forschungen des ARC, die nicht mehr das Individuum, sondern das Team in den Vordergrund stellte. Das neue erklärte Ziel war nun die Schaffung von Werkzeugen und Organisationsformen für die effektivere Gestaltung kooperativer Arbeit.³ Dies geschah zunächst mit Blick auf kleine, räumlich konzentrierte Teams wie das ARC selbst. Nachdem das Computernetz der ARPA, das ARPANET (vgl. Abs. 5.9.2), zu Beginn der siebziger Jahre langsam Gestalt annahm, weitete sich dieses Ziel auch auf größere und räumlich weit verstreute Arbeitsgruppen aus.

5.9.1 Das Journal

Das erste Werkzeug, das für diesen Zweck entworfen wurde, war das sogenannte Journal, ein permanent auf dem zentralen Computer gespeichertes und für jeden zugängliches Archiv von NLS-Dokumenten. Engelbart hatte die Entwicklung eines solchen Programmbestandteils schon 1966 angeregt, als er erste Überlegungen anstellte, wie man sein Computersystem zur Unterstützung von Managementaufgaben einsetzen könnte.⁴ Aber erst nachdem das Online-System auf

¹ Engelbart and SRI-ARC Staff 1972b, S. 42ff.

² Engelbart 1979, S. 64 (Zitat); Engelbart et al. 1973, S. 17.

³ Engelbart 1970a; Engelbart and SRI-ARC Staff 1970, S. 127f.; Engelbart 1972a; Engelbart et al. 1973, S. 11ff.

⁴ Engelbart et al. 1968, S. 26ff., speziell S. 47f.; Engelbart 1975.

5.9 Vom Individuum zur Gruppe: Die *Online Community* und das ARPANET

```

Baseline Task Estimates as of 26 APR 72          for: JCN
Months: > Apr-May-Jun-Jul- <
ONR Annual Report 8622 > !xxx          !4:1  5:2 <DCE JCN JBN??
Stacking Chairs> !xxx          !4/28  5/8 <DVN? JCN BER
Develop ARC APC Baseline> !xxxxx      !--   5:4 <PXR JCN DVN
Projection TV> !xxxxxx         !4:1  6:1 <EKV PBH JCN?
Demonstration Training> !xxxxxxx      !4:1  6:1 <JCN DVN MFA ???
RADC Final Report 8457> !xxxxxxxx     !4:1  6:2 <DVN JCN MFA CHI WHP
Resource Accounting Design> !xxxxxxxxx  !--   6:4 <JCN PKV RWW DCW
Operations Development> !xxxxxxxxxxxxx !--   ??? <JCN
NIC Operations Coordinator> !         !???  ??? <RWW JCN DCE
BRS Design> !-----          !--   -- <PXR JCN JDH
Recruiting> !-----          !--   -- <JCN CHI DCE RWW EKV
Journal Catalog> !-----          !--   -- <JCN HAH WSD
User Documentation Maintain>!-----  !--   -- <MFA JCN CP BER PML
Management And Coordination>!-----  !--   -- <DCE JCN RWW WHP
Accounting> !-----          !--   -- <JCN DVN
Visitors> !-----          !--   -- <JCN ALL
Vacations> !-----          !--   -- <ALL

```

Abbildung 61. Die Aufgaben einer Person (hier James C. Norton = JCN), erstellt vom *Baseline Record System*

dem SDS-940-Computer lief und eine nennenswerte Benutzung durch die Arbeitsgruppe einsetzte, wurde dieser Plan wiederaufgenommen und von David A. Evans 1968/69 implementiert.¹

Bei ihrem Entwurf orientierten sich Engelbart und Evans, wie der Name bereits andeutet, am traditionellen System der Fachzeitschriften. Wie schon Bush und Licklider vor ihnen, wollten auch sie auf die in Jahrzehnten gewachsenen und bewährten Verfahren zur Verteilung und Speicherung von Fachinformationen zurückgreifen, wie sie für die traditionellen Medien entwickelt worden waren.² Dazu wurden vier zentrale Merkmale ausgewählt und für die Arbeit mit dem Online-System modifiziert:³

- So wie Fachzeitschriften gezielt ein Fachpublikum ansprechen, gab es auch beim Journal die Möglichkeit zur gezielten Distribution von Dokumenten an

¹ Evans 1969. Evans war der einzige Wissenschaftler, der seinen Dokortitel für eine Arbeit in Engelbarts Projekt erhielt. Dafür gab es wohl zwei wichtige Gründe. Zum einen waren die Forschungen am ARC in so starkem Maße durch Engelbarts Ideen geprägt, daß es schwerfiel, innerhalb des Labors eigenständiges Profil zu entwickeln. Zum anderen wurde Engelbarts Arbeit außerhalb der ARPA Gemeinschaft immer noch sehr reserviert aufgenommen. Vgl. Email von David A. Evans an den Autor, 18. Mai 1997.

² Schon 1958, also unmittelbar nach dem Beginn seiner Tätigkeit am SRI, hatte sich Engelbart Gedanken über Verfahren zur Erfassung, Speicherung und Verteilung von Fachinformationen gemacht, die modernen Erfordernissen entsprachen. Vgl. Bourne and Engelbart 1958.

³ Evans 1969, S. 292ff.; Engelbart 1975, S. 174f. – Evans stellt in seiner Dissertation allerdings eher eine Analogie zu Henry David Thoreaus (1817–1862) Tagebüchern her, die dieser Journale nannte. Thoreau zeichnete darin, in geordneter Form und bereits mit Blick auf spätere Arbeiten, seine täglichen Erfahrungen und Erlebnisse auf. Vgl. Evans 1969, S. 245ff.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

Title	Date	Number	Author
* The TENEX Scheduler	21 Jul 71	7419	Andrews
* response memo	21 Jul 71	7415	Andrews
* response memo	12 Jul 71	7394	Andrews
* No title	14 May 71	6982	Andrews
* known performance problems	28 Feb 72	9313	ARC System
* SCHED1 problem	18 Feb 72	9232	ARC System
* Pod Reshuffle Proposal	13 Jul 72	11041	Auerbach
* Reply to 10954 (reply to 10953)	10 Jul 72	11002	Auerbach
* Reply on your comments on NIC documentation	5 Jul 72	10953	Auerbach
* re: update	5 Jun 72	10633	Auerbach
* let me tell you about the Handbook	5 Jun 72	10632	Auerbach
* re: attach/detach and at vs. after	23 May 72	10566	Auerbach
* re: output processor directives (again)	11 May 72	19417	Auerbach

Abbildung 62. ARC Journalverzeichnis, nach Autoren sortiert

bestimmte Teilgruppen mit unterschiedlichen Interessen. Wenn ein Dokument an das Journal übergeben wurde, legte der Autor fest, für welchen Adressatenkreis der neue Journaleintrag von Bedeutung war.

Jeder Benutzer besaß seinen eigenen elektronischen Briefkasten, in dem dann die für ihn bestimmten Dokumente (oder Verweise auf die Dokumente) abgelegt wurden. Traf ein neues Dokument ein, wurde er automatisch darauf aufmerksam gemacht, und er konnte seine elektronische Post dann am Bildschirm lesen oder ausdrucken.

- Damit besaß jeder Nutzer des NLS, ebenso wie der Leser einer Fachzeitschrift, sein persönliches Exemplar der ihn interessierenden Beiträge, die er bearbeiten, kopieren oder auch wegwerfen (löschen) konnte.
- Bibliotheken und spezielle Zeitschriften speichern und systematisieren die Beiträge der Fachzeitschriften und machen sie ihren Nutzern zugänglich. Diese Funktion übernahm beim NLS das Verzeichnis der Journaleinträge (Abb. 62) und der bereits erwähnte *pattern matcher*, mit dem das Journal nach Schlüsselworten durchsucht werden konnte. Die Effektivität dieses Verfahrens war allerdings durch die Kapazität des Plattenspeichers begrenzt, die es notwendig machte, Journaleinträge bereits nach kurzer Zeit auf Magnetband zu sichern.
- Schließlich gab es mit den Möglichkeiten zur Herstellung von Querverweisen zwischen NLS-Dokumenten eine formalisierte Methode für bibliographische Verweise. Dabei war es vorteilhaft, daß der Leser eines Dokuments mit einem einzigen Befehl auch auf ein zitiertes Dokument zugreifen konnte.

Zweck des *Journals* war die Unterstützung des (informellen) Dialogs zwischen den Mitarbeitern des Labors. Sie sollten so die Möglichkeit erhalten, neue Ideen schon in einem sehr frühen Stadium zur Diskussion stellen und die Ergebnisse der Diskussion bei ihrer weiteren Arbeit berücksichtigen zu können. Das Journal sollte hierdurch zu einem Mittel des *Bootstrapping* für lokale Arbeitsgruppen

werden. Zusätzlich versprachen sich Engelbart und Evans eine positive Wirkung auf die Arbeitsmotivation des Einzelnen und den Zusammenhalt innerhalb der Arbeitsgruppe.¹

Zunächst wurde das Journal allerdings mit gemischten Gefühlen von den Nutzern aufgenommen. Es gab beispielsweise einen Mitarbeiter des ARC, der nie eines seiner Dokumente in das Journal übergab, weil ihn der Gedanke abschreckte, dieser Vorgang sei unwiderruflich und das Dokument »für immer« abgespeichert.² Nachdem die anfänglichen Hemmungen einmal überwunden waren, entwickelte sich das Journal aber zu einem der am intensivsten benutzten Komponenten des NLS. Innerhalb der nächsten fünf Jahre wurden über 30 000 Einträge registriert, vom Einzeiler bis zum hundertseitigen Forschungsbericht, und das *Journal* wurde zu einem Vorbild für die wenig später entstehenden Systeme der Computerkommunikation, der elektronischen Post (email).³

5.9.2 ARPANET und Network Information Center

Das Journal war allerdings längst nicht das einzige Programm dieser Art. Bei praktisch jedem Time-Sharing-System seit Mitte der sechziger Jahre gab es die Möglichkeit, mit anderen Nutzern des gleichen Systems zu kommunizieren.⁴ Im Gegensatz zu diesen Kommunikationsprogrammen wurde das Journal auch in Hinblick auf die Rolle entworfen, die das ARC bei der Entwicklung eines von der ARPA initiierten überregionalen Computernetzwerks spielen sollte.

Das sogenannte ARPANET war das zweite große Projekt des IPTO in den sechziger Jahre und die konsequente Fortsetzung von Lickliders Kampagne zur Entwicklung von Time-Sharing-Systemen. Geistiger Vater des ARPANET war Lawrence G. Roberts (* 1937), der zum Vizedirektor des IPTO berufen wurde, als Robert Taylor 1966 Nachfolger von Ivan Sutherland als Direktor wurde. Zwischen 1969 und 1973 war Roberts schließlich selbst Direktor des IPTO.⁵ Er nahm Lickliders Idee eines »Intergalactic Computer Network« auf, das dieser bereits Anfang der sechziger Jahre formuliert hatte⁶ und entwickelte sie zum Konzept eines *Resource Sharing Networks* weiter. Ziel dieses Konzepts war es zunächst,

¹ Engelbart et al. 1973, S. 15; Engelbart 1975, S. 175; Evans 1969, passim.

² Engelbart 1988, S. 213.

³ Engelbart 1975, S. 175. Die Programme zum Verschicken und Empfangen von elektronischer Post, die sich ab etwa 1973 durchsetzten, waren allerdings sehr viel einfacher als Engelbarts Journal und konnten deshalb auf allen im ARPANET verwendeten Computern implementiert werden. Vgl. Hardy 1996 sowie Norberg et al. 1996, S. 178f.

⁴ Salus 1995, S. 95.

⁵ Norberg et al. 1996, S. 33ff.

⁶ Licklider 1960, S. 7. Den Begriff »Intergalactic Computer Network« prägte Licklider allerdings erst 1963 in einem internen Memorandum. Vgl. Norberg et al. 1996, S. 153f.

5. Douglas Engelbart und der Computer als Intelligenzverstärker



Abbildung 63. Lawrence G. Roberts, um 1970

die ungemein kapitalintensiven Time-Sharing-Computer möglichst ökonomisch zu betreiben. Mit Hilfe eines Computernetzes – so die Vorstellung – konnte man stets auf einer Maschine arbeiten, deren Auslastung momentan gering war, z. B. durch die Zeitverschiebung zwischen der amerikanischen Ost- und Westküste. Nach intensiver Weiterentwicklung des Konzepts durch Roberts, an der auch die von der ARPA geförderten Wissenschaftler mitgewirkt hatten, wurde das Entwicklungsprojekt im Juni 1968 öffentlich ausgeschrieben. Das ARPANET sollte ein dezentral organisiertes, paketvermitteltes Netz sein, das die wichtigsten Zentren der militärischen Forschung im ganzen Land miteinander verband.¹

Der offizielle Auftrag zur Entwicklung der notwendigen Hardware und zum Aufbau der Netzinfrastruktur wurde 1969 an Bolt, Beranek and Newman, Inc. vergeben. Für den Betrieb des Netzes waren neben dem ebenfalls von BBN betriebenen *Network Control Centers* noch zwei weitere Zentren vorgesehen. An der University of California in Los Angeles wurde das *Network Measurement Center* unter der Leitung von Leonard Kleinrock eingerichtet, das sich mit der Leistungsmessung und -bewertung beschäftigte. Das SRI wurde schließlich dazu ausersehen, das *Network Information Center* (NIC) zu betreiben, das als zentraler Umschlagplatz für alle das ARPANET betreffenden Informationen fungieren

¹ Licklider et al. 1968, S. 30ff.; Roberts and Wessler 1970; Kahn 1972; Roberts 1988, S. 143ff. Zur Entwicklungsgeschichte des ARPANET gibt es mittlerweile eine Vielzahl von wissenschaftlichen und populären Darstellungen. Vergleiche beispielsweise Salus 1995; Hauben and Hauben 1997; Hafner and Lyon 1996 und insbesondere Abbate 1999.

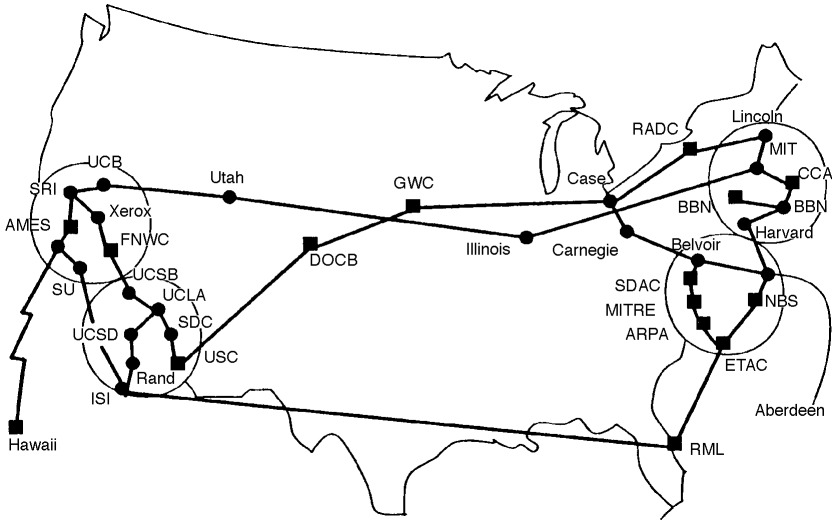


Abbildung 64. Struktur des ARPANET im Mai 1973

sollte. Das ARC schien für diese Aufgabe der ideale Partner zu sein, hatte Engelbart mit seinem Online-System doch ein Werkzeug, das dafür besonders geeignet erschien, speziell nach der Erweiterung um das Journal.¹

Im Oktober 1969 wurde die erste Verbindung des ARPANET zwischen der University of California in Los Angeles und dem SRI hergestellt. Bis Dezember waren auch die University of California in Santa Barbara und die University of Utah in Salt Lake City hinzugekommen. Schon im Dezember 1970 hatte das ARPANET 10 Knoten mit 19 angeschlossenen Computern und im Januar 1973 umfaßte es mit 35 Knoten und 38 Computern bereits die wichtigsten Zentren der durch die ARPA geförderten Zentren der Computerforschung (Abb. 64).² Obwohl der reguläre Netzbetrieb bereits im April 1970 eröffnet wurde, dauerte es aus technischen Gründen immerhin noch bis zum Oktober 1971, ehe das ARC den Nutzern des ARPANET alle Dienstleistungen des *Network Information Center* bzw. des NLS zur Verfügung stellen konnte.³

Durch die Einbindung des ARC in das ARPANET wurden einschneidende Veränderungen der verwendeten Hardware und Betriebssystemsoftware notwen-

¹ Roberts and Wessler 1970; Engelbart, D. C., »Study for the Development of Human Intellect Augmentation Techniques«, Quarterly Technical Letter Report No. 6 to NASA Langley Research Center, 28. November 1967, SUL/EC, 2-10.

² Shapiro 1969; Roberts 1988, S. 151f.

³ Engelbart and SRI-ARC Staff 1970, S. 120; Melvin and Watson 1971b.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

dig. Schon 1969 hatte sich gezeigt, daß der SDS-940-Computer nicht mehr als zwölf Bildschirmterminals in annehmbarer Weise bedienen konnte. Es war also absehbar, daß durch den Betrieb des *Network Information Centers* ein Ausbau der Rechnerleistung dringend notwendig wurde, zumal bereits für die Entwicklung des NIC zusätzliche Computerleistung benötigt wurde. Nachdem 1970 zunächst zusätzliche Minicomputer eingesetzt worden waren, wurde schon bald klar, daß nur der Ersatz des SDS-940-Computers durch eine leistungsfähigere Maschine das Problem würde lösen können. Dazu wurde zunächst eine vergleichende Studie der auf dem Markt verfügbaren Time-Sharing-Computer durchgeführt. Dabei stellte sich der PDP-10 von Digital Equipment mit dem von Bolt, Beranek and Newman entwickelten TENEX-Betriebssystem¹ als sinnvollste Lösung für die Anforderungen der ARC heraus.²

Der erneute Wechsel der Hardware hatte zur Folge, daß das Online-System, das zu diesem Zeitpunkt bereits einen Umfang von knapp 100 000 Zeilen Quellcode hatte, vollständig neu implementiert werden mußte. Zu diesem Zweck wurde – wie schon bei der SDS 940 – eine speziell für diesen Zweck entworfene maschinenorientierte Programmiersprache (L10) verwendet. Da die Portierung des NLS nicht erst mit der Lieferung des PDP-10 begonnen werden konnte, mußten Engelbart und seine Mitarbeiter zu einer neuartigen Vorgehensweise greifen. Mit Hilfe des Tree Meta Compilers entwickelten sie einen L10-Cross-Compiler für den SDS 940, mit dessen Hilfe die auf der 940 geschriebenen Programme in die PDP-Maschinensprache übersetzt werden konnten. Über das ARPANET wurden diese Programme zur University of Utah übertragen, die bereits einen PDP-10 am Netz hatten.³ Auf dieser Maschine konnten dann die noch vorhandenen Fehler gesucht und beseitigt werden, so daß die neue Version des NLS schon kurz nach der Lieferung des PDP-10-Computers im Februar 1971 lauffähig war.⁴

¹ Bobrow et al. 1972.

² Engelbart 1971, S. 1, 15. Der PDP-10 war der Nachfolger des PDP-6 und der größte von DEC produzierte Computer. Er kostete zwischen 110 000 \$ und 300 000 \$ und wurde im September 1967 auf den Markt gebracht. Es handelte sich um eine 36-bit-Maschine mit einer Leistung von 250 000 Instruktionen pro Sekunde, vergleichbar etwa mit einem mittleren Modell des IBM System/360. Der speziell für den Time-Sharing-Betrieb entworfene Computer wurde während der späten sechziger und frühen siebziger Jahre besonders an amerikanischen Universitäten häufig verwendet. Vgl. Pearson 1992, S. 54ff.; Bell and Bell 1983.

³ Bei dieser ersten großen Nutzung des ARPANET stellt sich auch heraus, wie sinnvoll die Computervernetzung war. In einem Interview erinnerte sich Engelbart: »So that was the first real use of the network; we used it to interlink facilities. (...) This saved us a great deal of time. It worked out beautifully. Even though there was a [PDP-10] that the AI group had just gotten in and was only 30 feet away from our 940, it was much better for us because we had this network connection to use, the one in Utah.« Vgl. Engelbart 1986/87, Interview 4, Tape 1, Side A.

⁴ Engelbart 1988, S. 208; Engelbart 1971, passim. Richard Watson schätzte den Entwicklungsaufwand des NLS für den SDS 940 auf 6–10 Mannjahre, während die PDP-10-Version mindestens den zehnfachen Aufwand erfordert habe. Vgl. Watson, R. W., »Experiences from the NIC Show-

Diese Version des NLS umfaßte nicht nur das bereits erläuterte Mail- und Journal-System, sondern auch spezielle Benutzerschnittstellen, die an die unterschiedlichen Zugriffsarten über das ARPANET angepaßt waren. Neben der bildschirmorientierten Version des NLS wurde nun auch eine Möglichkeit zur Nutzung des Online-Systems über ein Fernschreibterminal angeboten; sogar ein stapelorientierter Zugriff wurde geschaffen. Gleichzeitig wurde das sogenannte Telnet-Protokoll entwickelt, das die Form des Datenaustauschs zwischen einem beliebigen Fernschreibterminal und dem interaktiven Online-System standardisieren sollte.¹

Obwohl das für die SDS 940 entwickelte Videosystem zunächst auch zusammen mit dem PDP-10 weiter verwendet wurde, machten sich Engelbart und seine Mitarbeiter 1972 erneut Gedanken über ein preiswertes, leistungsfähiges, vor allem aber kommerziell erhältliches Terminal, mit dem die wachsende Zahl der Benutzer des *Network Information Center* in den Genuß der komfortablen Editiermöglichkeiten des NLS kommen konnten. Doch die Situation hatte sich seit 1967 nicht wesentlich verbessert. Immerhin gab es mittlerweile einen Hersteller, der ein Terminal anbieten konnte, das die technischen Spezifikationen weitgehend erfüllte. Die Grafikterminals vom Typ IMLAC PDS-1 kosteten allerdings immer noch 18 000 \$ statt der gewünschten 5 000 \$, so daß es für die Mehrzahl der sporadischen Nutzer nicht in Frage kam. Billigere marktgängige Geräte beschränkten sich häufig auf die Emulation eines Fernschreibterminals oder ließen die Nachrüstung mit einer Maus und einer Einhandtastatur nicht zu.²

Aus diesem Grund wurde für einfache, handelsübliche Textterminals eine Zusatzhardware, der sogenannte *Line Processor*, konstruiert, der eine zweidimensionale Ansteuerung des Bildschirms sowie den Anschluß von Maus und Einhandtastatur zuließ. Das Gerät, für das der 1971 auf den Markt gekommene Intel 4004 Mikroprozessor verwendet wurde und das etwa 1 200 \$ kostete, ließ sich so programmieren, daß man alle Funktionen des NLS, mit Ausnahme des Grafiksystems, mit einem beliebigen Textbildschirm über das ARPANET in annehmbarer Geschwindigkeit nutzen konnte.³

Gleichzeitig ergab sich das Problem, daß sich die Vorstellung eines trainierten Benutzers, der seine täglichen Aufgaben mit Hilfe des NLS bearbeitete, nicht länger aufrecht erhalten ließ. Die Mitarbeiter des ARC hatten es zunehmend mit »naiven« Nutzern⁴ zu tun, die nicht gewillt waren, mehr als ein Minimum an

ing Factors Creating Instability in Application Operations«, Journal #19 870, 26. Oktober 1973, SUL/EC, 27.

¹ Watson 1971a; Melvin and Watson 1971a; White 1971.

² Andrews 1974, S. 258f.

³ Engelbart 1973; Andrews 1974; Irby 1974, S. 247ff.

⁴ Jonathan Grudin hat mit Recht darauf hingewiesen, wie unglücklich die Begriffe des »naiven« oder »gelegentlichen« Nutzers gewählt ist, der verschleiert, daß es sich dabei häufig sehr wohl um

Lernaufwand in die Bedienung des NLS zu investieren. Die sich real formierende *Online Community* unterschied sich deshalb im Laufe der siebziger Jahre immer stärker von Engelbarts Idealvorstellung.¹

Es stellte sich aber bald heraus, daß angesichts der neuen Aufgaben des ARC eine grundlegende Neuorganisation des Labors notwendig war. Aus diesem Grund wurde das ARC 1973 mit seinen damals über 40 Wissenschaftlern in drei Untergruppen aufgeteilt. Die eine Gruppe unter der Leitung von Richard W. Watson war für die Softwareentwicklung zuständig, während James C. Nortons Gruppe für die Unterstützung der Anwender verantwortlich war.² Der Betrieb des *Network Information Centers* wurde von einer weiteren Gruppe unter der Leitung von Elizabeth (Jake) Feinler wahrgenommen, deren Aufgaben u. a. die Beratung der NIC-Benutzer, die Beschaffung und Analyse von Informationen über Netzwerkaktivitäten, die Verwaltung und Katalogisierung des wachsenden Bestands an NIC-Dokumenten, den Vertrieb von Dokumenten und Handbüchern sowie die Schulung von Benutzern umfaßten.³

Die Tatsache, daß die Dienstleistungen des *Network Information Center* auf der Basis des immer noch experimentellen Online-Systems angeboten wurde, wurde bald zum Kern eines laborinternen Konflikts. Für Engelbart war das NIC die Basis für die nächste Stufe des *Bootstrapping* und den Aufbau einer *Online Community*. Deshalb hatten die Forschungsaktivitäten des ARC meist Priorität vor den Belangen des *Network Information Centers*. Die Mitarbeiter des NIC bestanden jedoch darauf, daß sie einen eigenständigen Forschungsauftrag hätten. Elizabeth Feinler stellte in einem Memo beispielsweise fest, »that the NIC can create a whole new research area of resource sharing and information retrieval.« Im gleichen Memo konstatierte sie, daß das NIC innerhalb des ARC bedauerlicherweise als Stiefkind behandelt werde, statt angemessene Anerkennung und Unterstützung durch das ganze Labor zu erfahren.⁴ Wie berechtigt diese Kritik war, wurde durch die zunehmenden Klagen der NIC-Benutzer deutlich, die sich

Experten handelt. Deren Expertentum liegt freilich in anderen Bereichen als der Computerwissenschaft. So werden das Wissen und die Fähigkeiten des Computerexperten zum Bewertungsmaßstab für Computernutzer. Vgl. Grudin 1993, S. 112.

¹ Engelbart and SRI-ARC Staff 1972b, S. 125; Watson 1976, S. 358. – Diese Entwicklung war ursprünglich nicht geplant. Engelbart gab in einem Interview 1987 zu bedenken, er habe angenommen, das ARPANET sei zur Unterstützung von 15–17 Forschungseinrichtungen gedacht gewesen. Aus Rechtfertigungsgründen gegenüber dem Kongreß habe das IPTO zunehmend auch nichtwissenschaftlichen militärischen Einheiten mit ihren »naiven« Benutzern Zugang zum ARPANET gewährt. Vgl. Engelbart 1986/87, Interview 4, Tape 1, Side A.

² Engelbart 1988, S. 193.

³ Crocker 1971; Watson 1971b; North 1971; Engelbart and SRI-ARC Staff 1972b, S. 121ff.; Feinler 1977.

⁴ Feinler, E., »Adjunct to RWW's memo 19870«, Journal #19 874, 26. Oktober 1973, SUL/EC, 27.

über die ständigen Änderung in der Funktionalität des NLS und die häufigen Systemabstürze beschwerten.¹

Der zunehmende Anteil der Dienstleistungen an der Arbeit des ARC löste seit etwa 1973 auch einen schwerwiegenden Konflikt zwischen dem ARC und seinen militärischen Geldgebern aus, der innerhalb von etwa drei Jahren zum Zusammenbruch des Labors und zum Verkauf der verbliebenen Aktivitäten inklusive der Software Ende 1977 führten. Obwohl Engelbart nie sonderlich erfreut war, für eine wachsende Zahl von »naiven«, also für seine eigentlichen Forschungsziele völlig untypischen, Nutzern Dienstleistungen erbringen zu müssen, wurde er vom IPTO-Direktor Larry Roberts zu einer Fortführung der Aktivitäten angehalten.² Nach Roberts Ausscheiden im September 1974 bekam Engelbart das seit dem Mansfield Amendment von 1969 rauher gewordene Förderklima zu spüren, in dem jedes Projekt seine unmittelbare Bedeutung für die militärische Nutzung nachweisen mußte.³ Joseph Licklider, der im Januar 1974 zum zweiten Mal Direktor des IPTO geworden war⁴, sah zumindest keine Möglichkeit für die weitere Förderung der NIC-Aktivitäten und bekundete auch sonst starke Zweifel an der Förderungswürdigkeit des ARC. Nach Engelbarts Auffassung war Licklider davon überzeugt, »that: a) [ARC was] ferociously overcharging everybody; b) he just didn't see that as a technology transfer; and c) [ARC] had too many people out there supporting and training. This was an admission of the weakness of our system, that we couldn't teach the people how to use it.«⁵

Dies hatte zur Folge, daß die ARPA und das RADC ab 1974 ihre finanzielle Unterstützung schrittweise reduzierten und 1977 ganz einstellten, nachdem die Förderung durch die NASA bereits 1972 ausgelaufen war.⁶ Als eine eher hilflose Reaktion auf diese Entwicklung wurde in Kooperation mit Tymshare, Inc., einem kommerziellen Computerdienstleister und Betreiber eines eigenen Computernetzes im kalifornischen Cupertino, die Benutzung des NLS Anfang 1974 als kommerzieller Service angeboten, der allerdings wegen der geringen Anzahl an das ARPANET bzw. TYMNET angeschlossener Institutionen, der restriktiven Verga-

¹ Watson, R. W., »Experiences from the NIC Showing Factors Creating Instability in Application Operations«, Journal #19 870, 26. Oktober 1973, SUL/EC, 27.

² Watson, R. W., »Some Known ARC Problems«, Journal #8 432, 13. Januar 1972, SUL/EC, 18.

³ Hamilton 1969; Boffey 1970.

⁴ Der Übergang von Roberts zu Licklider als Direktor des IPTO markierte gleichzeitig das Ende der scheinbar grenzenlosen Forschungsfreiheiten. Unter dem Druck des Mansfield Amendment fand sich zunächst kein Nachfolger für Roberts, bis sich Licklider – ohne großen Enthusiasmus – zu einer zweiten Amtszeit überreden ließ. Sein Nachfolger David C. Russell (ab 1975) war schließlich auch kein Zivilist mehr, sondern aktiver Offizier. Vgl. Norberg 1996, S. 46f.

⁵ Engelbart 1986/87, Interview 4, Tape 2, Side A.

⁶ Engelbart 1988, S. 192f.; Bertram Raphael, »Trip & Contact Report: ARPA, NASA, RADC, DDR&E, AFOSR«, 3. Mai 1977, Journal #40 135, SUL/EC, 61.

be von Netzzugängen durch das amerikanische Verteidigungsministerium und des hohen Preises von 40 000 \$ pro Jahr kaum Interessenten fand.¹

5.9.3 Der Exodus

Neben der externen entwickelte sich nach 1970 auch eine laborinterne Krise, die sich auf technische, psychologische und soziale Ursachen zurückführen läßt und nicht weniger als die Probleme mit der ARPA zum Zusammenbruch des ARC beigetragen hat.

Zunächst hatten sich die technologischen Randbedingungen während der sechziger Jahre erheblich verändert. Durch die Fortschritte in der Halbleitertechnik, die Entwicklung von integrierten Schaltkreisen und die beginnenden Miniaturisierung begannen die Preise für Computerrechenleistung erheblich zu sinken und den Sinn von Time-Sharing-Systemen grundsätzlich in Frage zu stellen. Schließlich war Time-Sharing ursprünglich nur eine Möglichkeit gewesen, jedem Benutzer einen direkten Zugang zu einem Rechner zu verschaffen, dessen exklusive Nutzung durch eine einzelne Person unwirtschaftlich gewesen wäre. Dementsprechend entzündete sich bereits um 1965 die Diskussion, ob es nicht sinnvoller sei, kleine Einbenutzermaschinen statt der großen, teuren und komplexen Time-Sharing-Systeme zu verwenden. Dabei konnten die Gegner des Time-Sharings auch darauf verweisen, daß die Softwareentwicklung für Großcomputer vielfach problematischer, langwieriger und teurer war als man im Vorfeld erwartet hatte.²

Außerdem hatten die Betreiber von Time-Sharing-Computern mit ganz neuen Schwierigkeiten zu kämpfen. Die komplexe und deshalb anfangs stark fehlerbehaftete Software machte Time-Sharing-Computer tendenziell weniger zuverlässig als Einbenutzermaschinen. Die freie Verfügbarkeit von Rechenleistung führte bei Time-Sharing-Computern normalerweise zu einem starken Anstieg der Nutzung. Der dadurch erforderliche weitere Ausbau der Rechenleistung hinkte in der Regel stets der Nachfrage hinterher.³ Dies hatte zur Folge, daß Time-Sharing-Computer während des Tages meist überlastet waren, während in der Nacht teure Kapazitäten ungenutzt blieben.⁴

Die problematischen Aspekte von Time-Sharing-Computern und die sinkenden Preise für kleinere Computer führten dazu, daß selbst namhafte Verfechter des Time-Sharings wie Joseph Licklider die Überzeugung aussprachen, daß sich die kleinen Einbenutzermaschinen für den Einsatz in Wissenschaft und Technik mittelfristig gegen die großen Time-Sharing-Computer durchsetzen würden:

¹ Nouhuys, Dirk H. van, »Other Sources of Money, Some Thoughts and Leads«, Journal #7 388, 8. Februar 1974, SUL/EC, 18. Siehe auch die Liste der NLS-Nutzer in Anhang C.

² Brooks 1975; Mahoney 1990.

³ McDonald 1965; Emerson 1965; Gruenberger 1966; Solomon 1966.

⁴ Clark 1990; Norberg et al. 1996, S. 104ff.

»Small stand-alone computers with teletypewriters (and even graphical displays) are now widely affordable, and the special-purpose packages now developed and in use make them the preferred instrument of control and analysis in many laboratories.«¹

sich Engelbart nicht aus ökonomischen Gründen für die Verwendung eines Time-Sharing-Computers entschieden, sondern wegen der Möglichkeit zur Kommunikation und Kooperation bei einem Computer, der gleichzeitig von einer Vielzahl von Benutzern verwendet wurde. Eine Abkehr vom Time-Sharing-Paradigma und die Verwendung von kleinen, vernetzten Einbenutzercomputern war also schon deswegen nicht ohne weiteres möglich, da die in Frage kommenden Computer in der Regel nicht die Leistungsfähigkeit besaßen, um neben dem normalen Betrieb auch noch den im ARPANET anfallenden Datenverkehr zu verwalten. Lokale Datennetze mit weniger aufwendigen Übertragungsmechanismen standen Engelbart noch nicht zur Verfügung. Sie wurden – auch als Reaktion auf die Erfahrungen am ARC – wenig später am *Xerox Palo Alto Research Center* (Xerox PARC) entwickelt, wo sie zu einem zentralen Bestandteil eines neuen Paradigmas wurden (vgl. Kapitel 6).

Das Problem wurde durch die Teilnahme des ARC am Aufbau des ARPANET weiter verschärft. Die Aufgaben des NIC waren eben nicht durch eine Vielzahl von kleinen Einbenutzermaschinen zu erfüllen, sondern nur durch einen Großcomputer wie den PDP-10, auf den schon bald mehr als 100 Nutzer täglich zugriffen.² Richard Watson formulierte diese Einschätzung noch 1976 folgendermaßen:

»... even when hardware costs decrease to the point where a user can perform 90 percent of his work using tools and information that operate in the processor in his work station, he will want to have access to a computer network to ... communicate in various forms with others ... «³

Insofern behinderte Engelbarts strikt evolutionärer Ansatz des Bootstrapping und seine Beteiligung bei der Entwicklung des ARPANET den Übergang zur Verwendung von kleinen Einbenutzermaschinen, wie er um 1970 von einer Reihe von jüngeren Computerwissenschaftlern verfochten wurde, insbesondere am nahegelegenen Xerox PARC.

Schließlich wurde auch die Weiterentwicklung des NLS-Softwaresystems zunehmend problematisch. Durch den Fortschritt bei der Hardware war es alle zwei

¹ Licklider 1968, S. 204.

² Engelbart and SRI-ARC Staff 1972b, S. 125.

³ Watson 1976, S. 359. Engelbart rechnete sogar damit, daß es mittelfristig kleine, tragbare Terminals geben würde, mit denen man über das Netz auf seinen »Heimatrechner« zugreifen könnte. Vgl. Engelbart et al. 1973, S. 12.

bis drei Jahre notwendig, das Online-System auf einen neuen Computer zu portieren. Beim Übergang vom SDS 940 auf den PDP-10 zeigte sich, daß für die Fortentwicklung der Software die Verwendung von Methoden des *Software Engineering* notwendig waren.¹ Das ARC konnte es sich nicht mehr leisten, für die Zeit der Portierung ganz auf ein funktionierendes Computersystem zu verzichten. Außerdem waren mittlerweile acht Programmierer allein mit der Portierung und Pflege des NLS beschäftigt, das einen Umfang von mehr als 100 000 Zeilen Quellcode erreicht hatte. Unter diesen Vorzeichen entschloß man sich 1973 zur vollständigen Neuimplementierung des NLS unter Verwendung einer systematischen, modularen Programmiermethode, die in Zusammenarbeit mit dem Xerox PARC entwickelt wurde.²

Die Entwicklung des NLS befand sich also in den frühen siebziger Jahren offensichtlich in einer Sackgasse. Die Komplexität des wachsenden Softwaresystems nahm in zunehmendem Maße die Entwicklungskapazitäten in Anspruch und führte zu dem Eindruck, das ARC habe seine Dynamik weitgehend eingebüßt. Andererseits behinderte die konsequent verfolgte evolutionäre Bootstrapping-Methodik und die Festlegung auf Time-Sharing einen grundlegenden Wandel, der Engelbarts Labor aus seiner technischen Erstarrung hätte befreien können.³

Die technischen bzw. methodischen Probleme von Engelbarts Projekt sind als Erklärung für dessen raschen Zusammenbruch allein allerdings nicht ausreichend. Hier kann man auf den sozialen und psychologischen Druck hinweisen, der als Folge von »Organisationsexperimenten« seit 1972 entstand. Er trug dazu bei, daß bis 1975 ein Großteil von Engelbarts Mitarbeitern das ARC verließ. So kam der Selbstbeobachtung in Engelbarts Konzept des *Bootstrapping* eine entscheidende Rolle für die Rückkopplung innerhalb des Entwurfszyklus zu. Als nach 1968 aus den individuellen Programmierern Entwicklungsgruppen wurden, mußte auch das Instrumentarium der Selbstbeobachtung modifiziert werden. Dazu wurde zunächst mit einer Veränderung der Arbeitsumgebung experimentiert. Die bisherigen Einzelbüros wurden durch ein gemeinsames Großraumbüro ersetzt, in dem durch halbhohe Stellwände Bereiche für Gruppen und Einzelpersonen flexibel eingerichtet werden konnten.⁴ Daneben wurde auch mit unterschiedliche Orga-

¹ *Software Engineering* war ein Schlagwort, unter dem seit 1968 Verfahren zur systematischen, ingenieurmäßigen Produktion von Softwaresystemen entwickelt wurden. Hintergrund dieser Entwicklung waren die immer häufigeren Probleme bei der termingerechten Fertigstellung komplexer Programme (z. B. des Betriebssystems für das IBM System/360). Vgl. dazu Naur and Randell 1969; Bauer 1993 sowie Mahoney 1990.

² Engelbart et al. 1973, S. 15f.; Irby 1974, S. 247; Watson 1974, S. 20ff.; Charles F. Dornbusch. »Paper about Conversion of NLS to run in MPS environment«, 26. Februar 1973, SUL/CHIP, 2-5; Irby, Charles H., »Report re. ARC's envisioned changes and extensions to NLS«, Journal #15 249, 25. Juli 1973, SUL/CHIP, 7-11.

³ Bardini and Horvath 1995, S. 52f.

⁴ Engelbart and SRI-ARC Staff 1970, S. 37ff.

nisationsstrukturen experimentiert, u. a. einer Matrixorganisation, bei der jeder Mitarbeiter in mehrere Projektgruppen eingebunden war.¹

Durch diese Maßnahmen sollte die Kommunikation und Kooperation zwischen den bislang isoliert arbeitenden Wissenschaftlern und die Vernetzung zwischen ansonsten unabhängigen Gruppen intensiviert werden. Da die Selbstbeobachtung bei Arbeitsgruppen ungleich problematischer war als bei Einzelpersonen und ohnehin nicht jedem Mitarbeiter gleich gut lag, wurden nach und nach einige Soziologen und Psychologen eingestellt, die als neutrale Beobachter fungieren sollten. Bei vielen der technisch orientierten Mitarbeiter Engelbarts, die sich teilweise bereits darüber beklagten, daß sie im Rahmen ihrer Tätigkeiten so viele Anträge und Berichte verfassen mußten, statt Programme zu schreiben, stieß die Überwachung ihrer Tätigkeit auf offenen Widerstand und verstärkte ihre unterschweligen Unzufriedenheit mit ihrer Arbeit am ARC.²

Schließlich muß man wohl feststellen, daß Engelbarts Projekt innerhalb des SRI zunehmend zu einem Fremdkörper wurde. Viele der jungen Wissenschaftler, die Engelbart in seinem Labor beschäftigte, weil sie brillante Programmierer oder Hardwareentwickler waren, brachten die Ideen und das Lebensgefühl der sogenannten Gegenkultur in das Projekt mit ein (vgl. Kapitel 7). Sie standen damit in krassem Gegensatz zum konservativen Selbstverständnis des SRI, das mit den wichtigsten Industrieunternehmen kooperierte und an vielen geheimen militärischen Entwicklungen partizipierte. Obwohl sich Engelbart bemühte, beiden Seiten gerecht zu werden, war er doch stets zwischen den unvereinbaren Gegensätzen gefangen.³

Insofern hatte Engelbart mit den typischen Problemen evolutionärer Technikentwicklung zu kämpfen, wie sie Herbert Simon in seinem Buch »Die Wissenschaft vom Künstlichen« (1969) beschrieben hat. Da jeder evolutionäre Prozeß nur neue Lösungen erzeugen kann, die ein wenig besser sind als die alten, wird das System zwar ständig verbessert, es ist aber unwahrscheinlich, daß ein irgendwie geartetes Optimum erreicht werden kann. Bei rationaler Entscheidung bietet eine evolutionäre Entwicklung auch kaum Möglichkeiten, vom einmal eingeschlagenen auf einen alternativen Entwicklungspfad zu wechseln, ohne das bereits Erreichte aufzugeben. Bei einem solchen Wechsel wären die anfallenden Kosten vielfach sehr viel höher als der kurzfristig zu erwartende Nutzen. In der biologischen Evolution, bei der rationale Entscheidungen und die effiziente Nutzung

¹ Irby, Charles H., »Suggestion for organization of NLS software people«, 29. Dezember 1971, SUL/CHIP, 6-2; Bardini 2000, Chapter 9.

² Matzorkis, Gus, »Some Reflections Of and On ARC«, Journal #11 732, 11. September 1972, SUL/EC, 62; Engelbart and Lehtman 1988 S. 246. Vergleiche auch Barnes 1997, S. 25 und Bardini 2000, Chapter 9.

³ Rheingold 1985, S. 202; Vallee 1982, S. 100f.

5. Douglas Engelbart und der Computer als Intelligenzverstärker

von Ressourcen belanglos sind, kann hingegen eine Vielzahl von Alternativen erzeugt und wieder verworfen werden. Selbst hochentwickelte, aber nicht mehr anpassungsfähige Spezies werden deshalb von anpassungsfähigeren Arten verdrängt und sterben aus.¹ Ohne die Analogie zu sehr verallgemeinern zu wollen, befand sich Engelbart mit seinem Projekt zu Beginn der siebziger Jahre in einer ähnlichen Situation. Obwohl er mit NLS ein hochentwickeltes technisches System zur Hand hatte, behinderte dessen Komplexität und die gewählte Entwurfsmethodik eine Anpassung an die Umweltveränderungen in Form der Fortschritte in der Mikroelektronik, der sich wandelnden Nutzung des Computers und der veränderten Einstellung des Menschen zur Computertechnik.

Es verwundert deshalb nicht, daß in den frühen siebziger Jahren viele der besten Wissenschaftler das ARC verließen, um an Xerox' neuem Forschungszentrum zu arbeiten, das nur wenige Kilometer von Menlo Park seinen Sitz hatte und viel mehr als das SRI die technischen und kulturellen Strömungen der siebziger Jahre repräsentierte.² Dort wurde den Ideen, die Engelbart während der sechziger Jahre entwickelt hatte, neues Leben eingehaucht.

5.9.4 Epilog

Mit Engelbarts Labor ging es Mitte der siebziger Jahre schnell zu Ende. Nachdem man anfangs noch versucht hatte, mit dem neuen Konkurrenten Xerox PARC zu kooperieren³, blutete das ARC seit 1974 finanziell und personell immer stärker aus. Das Management des SRI reagierte auf diese Entwicklung, indem sie Engelbart und James Norton im November 1975 von einem Tag auf den anderen ihrer Positionen enthoben, aber nicht entließen. Die Leitung des ARC übernahm Bertram Raphael, der lange Jahre dem SRI-Labor für Künstliche Intelligenz vorgestanden hatte. Dieser Schritt verbesserte die Lage des ARC allerdings nicht, da Engelbart dem ARC als informeller Leiter weiterhin erhalten blieb. So beschränkten sich die Aktivitäten bald darauf, das ARC in ein unabhängiges Unternehmen umzuwandeln oder an ein interessiertes Unternehmen zu verkaufen, während sich der Exodus des verbliebenen wissenschaftlichen Personals verstärkt fortsetzte.⁴

¹ Simon 1994, S. 38ff., 141f. Vergleiche auch Maturana und Varela 1997, S. 64ff., Kapitel 5, passim.

² Zu den ersten »Überläufer« gehörten so wichtige ARC-Wissenschaftler wie Engelbarts langjähriger Stellvertreter William K. English, der NLS-Chefprogrammierer Jeff Rulifson und der Hardwarespezialist Roger Bates. Später folgten auch William Duvall, Charles H. Irby, William Paxton, David C. Smith, Donald Wallace sowie Robert Belleville, die alle erheblichen Anteil an der Entwicklung des Online-Systems gehabt hatten.

³ Mitchell 1973.

⁴ Engelbart 1986/87, Interview 4, Tape 3, Side A; Bertram Raphael, »Status Report after five months in ARC«, Journal #40 218, 9. Mai 1977, SUL/EC, 61; Douglas C. Engelbart and James C. Norton, »Proposed AKW Business Definitions«, Journal #40 354, 17. Mai 1977, SUL/EC, 61; Dirk

Auch das einzige Produkt des ARC, das Online-System, das mittlerweile in einer Reihe von militärischen Organisationen für die tägliche Arbeit verwendet wurde, gab keinen echten Anlaß, das ARC weiter am Leben zu erhalten. Nüchtern beschrieb Bertram Raphael im März 1977 die reichlich negative Einschätzung der NLS-Nutzer:

»We still keep hearing that NLS is complicated, hard to learn, poorly documented. Whitaker: »If NLS were gone tomorrow, nobody in ARPA would really notice.« Krutz: »Fantastic for our engineers, but too hard for our secretaries.«¹

Trotz dieser schlechten Voraussetzungen wurden die verbliebenen Aktivitäten des ARC am 1. Januar 1978 an Tymshare, Inc. verkauft und in die Entwicklungsabteilung für Büroautomation integriert. Das Online-System wurde unter dem neuen Namen AUGMENT weiterhin wenig erfolgreich vermarktet. Engelbart wurde von Tymshare zwar weiter beschäftigt, war aber in den folgenden Jahren als *Senior Scientist* im wesentlichen zum Nichtstun verdammt. Daran änderte sich auch nichts als Tymshare 1984, vor allem wegen seines Computernetzwerks, vom Flugzeughersteller McDonnell Douglas übernommen wurde.²

Erst nach seiner Pensionierung 1990 konnte Engelbart, nun als einer der Erfinder von Hypertext und *Computer Supported Cooperative Work* international anerkannt³, wieder seinen eigentlichen Forschungsinteressen nachgehen. Er leitet heute das 1991 gegründete Bootstrap Institute in Fremont, Ca., das von der Logitech Corp., dem weltgrößten Hersteller von Computermäusen, beherbergt wird.

H. van Nouhuys, »Organization, Management, and Cherishing«, Journal #40 531, 28. Mai 1977, SUL/EC, 61; Robert L. Lieberman, »RLL's thoughts on the new company and recent events«, Journal #40 537, 29. Mai 1977, SUL/EC, 61.

¹ Bertram Raphael, »Trip & Contact Report: ARPA, NASA, RADC, DDR&E, AFOSR«, 3. Mai 1977, Journal #40 135, SUL/EC, 61.

² Engelbart 1986/87; Engelbart 1988, S. 229. Das *Network Information Center* wurde allerdings bis 1991 durch SRI International weiterbetrieben. Vgl. Williamson and Nobile 1991.

³ Engelbart hat in den letzten Jahren eine Vielzahl von Preisen und Ehrungen erhalten, darunter den ACM Software System Award (1990), den IEEE Computer Pioneer Award (1992), den Lemelson-MIT Prize (1997) und den ACM Turing Award (1998).

6. Xerox PARC und die Architektur der Information

Machines which do one thing only are boring.

*Alan C. Kay*¹

Der Aufstieg der Xerox Corporation mit ihrem Hauptsitz in Rochester, N.Y. (später in Stamford, Conn.) von einem unbedeutenden Hersteller von Fotopapier zu einem der größten Büromaschinenhersteller der Welt zwischen 1945 und 1970, ihr anschließender Niedergang bis an den Rand des Konkurses und ihr Wiederaufstieg seit 1983 ist eines der interessantesten Kapitel der amerikanischen Wirtschaftsgeschichte.² Bestandteil dieser Geschichte sind auch Xerox' Bestrebungen seit 1968, sich im Computermarkt zu etablieren und der übermächtigen IBM Marktanteile streitig zu machen. Ein wichtiges Mittel für diese Bemühungen war die Einrichtung eines Forschungslaboratoriums für Computerwissenschaften und Halbleitertechnik in Palo Alto, in unmittelbarer Nachbarschaft zur Stanford University am Rande des Silicon Valley. Dort wurden im Laufe der siebziger Jahre die Grundlagen für eine völlig neue Form der Computernutzung gelegt, das auf der Verwendung von kleinen, leistungsfähigen Computern im Einbenutzerbetrieb basierte, die über ein lokales Computernetzwerk miteinander verbunden waren. Obwohl Xerox damit Ende der siebziger Jahre alle Voraussetzung zur Markteinführung des ersten wirklichen Personal Computers besaß, erwies sich ihr – verspätet vorgestelltes – Produkt als ein wirtschaftlicher Mißerfolg.

6.1 Das Kopiererimperium im Computerzeitalter

6.1.1 Diversifizierung und das Büro der Zukunft

1947 erwarb die Xerox Corporation, damals noch unter dem Namen Haloid Co., die Rechte an Chester F. Carlsons Patent zur elektrofotografischen Reprodukti-

¹ Kay 1969, S. 9.

² Dessauer 1971; Jacobson and Hillkirk 1986; Smith and Alexander 1988; Kearns und Nadler 1993.

on. Unter dem patriarchalischen Hauptgeschäftsführer Joseph C. Wilson (1910–1971) und dem Entwicklungschef John Dessauer (1905–1993) investierte das Unternehmen in den nächsten 14 Jahren 75 Mio. Dollar in die Entwicklung der Xerographie. Nach zwei wenig erfolgreichen Vorgängerprodukten wurde 1959 der Kopierer Modell 914 auf den Markt gebracht, der wegen seiner Leistungsfähigkeit und einfachen Bedienbarkeit zu einem riesigen Erfolg wurde und Xerox zum Synonym für Fotokopierer werden ließ. In den folgenden Jahren erlebte Xerox in bezug auf Umsatz, Gewinn und Mitarbeiterzahl ein rasantes Wachstum, beherrschte unangefochten den Markt für Fotokopierer und gehörte bald zu den hundert größten Unternehmen des Landes.¹

Während dieser Wachstumsphase vollzog sich ein Generationenwechsel an der Spitze des Unternehmens. Joseph Wilson trat 1968 vom Posten des Hauptgeschäftsführers zurück und überließ diesen seinem Protegé Peter McColough (* 1923), der seit Anfang der fünfziger Jahre Xerox' Verkaufsorganisation aufgebaut hatte. Im gleichen Jahr trat auch John Dessauer als Chef der Forschung zurück, sein Nachfolger wurde Jacob E. Goldman, der zuvor die Forschung bei Ford geleitet hatte. McColoughs wichtigstes Ziel war es, Xerox' Wachstumsdynamik zu bewahren. Da der Erfolg des Unternehmens bis dahin auf einem einzigen Produkt basierte, entschloß sich McColough zur Diversifizierung des Unternehmens. Ein besonders lohnendes Betätigungsfeld schien die Datenverarbeitung zu sein, ohne die ein modernes Unternehmen im Bereich Bürotechnik und Telekommunikation künftig nicht mehr auskommen würde.²

Nachdem man Mitte der sechziger Jahre erfolglos versucht hatte, die *Digital Equipment Corporation* zu übernehmen, kaufte Xerox Anfang 1969 das in El Segundo, Ca. beheimatete Unternehmen *Scientific Data Systems*. SDS war ein im Vergleich zum Marktführer IBM kleines Unternehmen, das aber wegen seiner Geschäftsbeziehungen mit der NASA und wegen seiner Beschränkung auf Computer für wissenschaftliche und technische Anwendungen mit Gewinn arbeitete.³ Nach McColoughs Vorstellungen sollte sich *Xerox Data Systems* (XDS), wie das Unternehmen künftig hieß, nicht nur in den bisherigen Marktsegmenten betätigen, sondern die ganze Breite des Datenverarbeitungsmarktes abdecken. Damit

¹ Kearns und Nadler 1993, S. 25–41; Smith and Alexander 1988, S. 34ff. In der jährlich vom Wirtschaftsmagazin *Fortune* aufgestellten Liste der größten Industrieunternehmen wurde Xerox 1963 noch auf Rang 294 geführt, lag 1969 bereits auf Rang 71 und 1972 auf Rang 40.

² Kearns und Nadler 1993, S. 54ff.; Smith and Alexander 1988, S. 23ff.

³ Smith and Alexander 1988, S. 30. Der Kaufpreis von 900 Mio. \$ (in Xerox Aktien) sorgte allerdings in Finanzkreisen für einige Unruhe, da SDS 1968 bei einem Umsatz von 100 Mio. \$ einen Gewinn von 10 Mio. \$ erwirtschaftet hatte und setzte Xerox' jüngstes Tochterunternehmen unter einen enormen Erfolgsdruck.

begab sich Xerox in bewußte Konkurrenz zu IBM, mit dem Ziel, die Nummer Zwei in der Computerbranche zu werden.¹

Die Technologie, über die man durch den Aufkauf von SDS verfügte, hatte allerdings wenig Bezug zu Xerox' Kerngeschäft, der Büromaschinenteknik. Es bedurfte also einer Vision, an der sich Xerox' Computeraktivitäten in Zukunft orientieren konnten und eines neuen Forschungslabors, das sich mit der Umsetzung beschäftigen sollte.² Am 3. März 1970 stellte McColough in einer Rede an die *New York Society of Security Analysts* die ehrgeizigen Ziele vor, die er für Xerox und XDS gesetzt hatte. So wie Joseph Wilson einen »Kopiererfeldzug« gestartet hatte, wollte er das »Büro der Zukunft« entwickeln lassen. Dabei griff er Überlegungen auf, die um 1970 die Bedeutung der Information in Wirtschaft und Gesellschaft betonten und zum Rohstoff eines nachindustriellen Informationszeitalters erklärten.³ McColough gab aber zu bedenken, daß Information und Wissen nicht nur zur Basis einer neuen gewinnträchtigen Industrie wurde, sondern auch zu einem Problem werden konnte. Mit seiner Einschätzung, »[that] the hard reality ... is that in attempting to gather, process, absorb, and disseminate information and knowledge today, we find ourselves living more and more in the confusion of tied-up telephones, computer printouts, procedure manuals, stacked airplanes, unnecessary correspondence, meetings, mail, memoranda, and aging files marked ›must read‹ «⁴, beklagte McColough nichts anderes als das schon von Vannevar Bush und Douglas Engelbart identifizierte Informationsproblem, in diesem Fall allerdings nicht im Bereich der Wissenschaft, sondern der Wirtschaft. Seine Vorstellung war, daß die Forschung versuchen müsse zu verstehen, wie Information innerhalb eines Unternehmens organisiert ist und wie sie »funktioniert«. Darauf sollten neue Verfahren und Produkte zur Büroautomation aufbauen. Eines Tages, so stellte sich McColough vor, würde Xerox eine Reihe von Produkten herstellen, die miteinander verbunden sein würden, um den Informationsfluß zu beschleunigen und die Kommunikation zu verbessern. Deshalb, so kündigte McColough in der gleichen Rede an, werde man künftig einen Teil des jährlichen Forschungs-etats von 100 Mio. \$ in ein neues Forschungszentrum investieren. Dessen Aufgabe sollte es sein, zu untersuchen, wie Digitaltechnik und physikalische Grundlagenforschung zur sogenannten »Architektur der Information« beitragen konnten.⁵

Damit entsprach er auch dem Wunsch von Jack Goldman, der bei seinem ersten Besuch des zentralen Xerox-Forschungszentrums in Webster, N.Y. erstaunt festgestellt hatte, daß Xerox bislang keinerlei Forschung im Bereich der Digital-

¹ Flamm 1988, S. 102f, 130f.; Kearns und Nadler 1993, S. 58; Smith and Alexander 1988, S. 46ff.

² Kearns und Nadler 1993, S. 59.

³ Drucker 1969; Bell 1985.

⁴ C. Peter McColough, »Remarks prepared for delivery to The New York Society of Security Analysts, March 3, 1970«, Xerox Corporation.

⁵ Kearns und Nadler 1993, S. 104f.; Smith and Alexander 1988, S. 48.



Abbildung 65. Das 1975 fertiggestellte Gebäude des Xerox Palo Alto Research Centers in den Hügeln oberhalb von Palo Alto

und Computertechnik betrieb und enormen Nachholbedarf hatte. Das neue Forschungszentrum sollte deshalb auch die Zusammenführung von Xerox' hochentwickelter analoger Kopierertechnik und der Digitaltechnik realisieren.¹

6.1.2 Die Elite der Computerwissenschaftler

Da man aber einen ungesunden Konkurrenzkampf der alten und neuen Forschungsschwerpunkte um Ressourcen und Einfluß fürchtete, entschied man sich gegen eine Ansiedlung der Computerforschung am Firmensitz im Staate New York, sondern für einen Standort in Kalifornien, genauer in Palo Alto, 50 km südlich von San Francisco. Auf diese Weise suchte man die Nähe der Computerindustrie im Santa Clara County, dem Silicon Valley und der renommierten Universitäten Stanford und Berkeley und ignorierte bewußt diejenigen Stimmen, die sich für einen räumlichen engen Kontakt der Computerforschung mit dem restlichen Unternehmen aussprachen. Bereits Ende Juni 1970 wurde das *Palo Alto*

¹ Perry and Wallich 1985, S. 62, 67; Smith and Alexander 1988, S. 42f.



Abbildung 66. George E. Pake, um 1996 (links) und Robert Taylor, um 1972 (rechts)

Research Center gegründet, sein erster Manager wurde der 46jährige Physiker George E. Pake (Abb. 66, links).¹

Das Zentrum war in eine Reihe von Labors gegliedert, die jeweils bis zu 50 Wissenschaftler beschäftigten. Jeder der drei bis fünf Forschungsschwerpunkte pro Labor wurde von Gruppen mit acht bis zwölf Wissenschaftlern bearbeitet. Bis Ende 1971 hatte man insgesamt drei Labors eingerichtet: das *General Science Laboratory*, das vor allem physikalische Forschung betrieb, das *Computer Science Laboratory* (CSL), das neue Hardwarekonzepte entwickeln sollte und das *System Science Laboratory* (SSL).

Mit der Einrichtung des SSL wollte Xerox die Probleme umgehen, die bei der Entwicklung umfangreicher Computersysteme während der sechziger Jahre aufgetreten waren. Man war sich bewußt, daß auch die geplanten Büroinformationssysteme aus einer Vielzahl heterogener Hard- und Softwarekomponenten bestehen würden, die standardisierten Verfahren und Protokollen genügen mußten. So bestand beispielsweise Xerox' erster Laserdrucker nicht nur aus der neuentwickelten Hardware, sondern benötigte auch eine spezielle Software, die die Erzeugung der Zeichen und die Formatierung der Seiten übernahm, sowie ein Protokoll,

¹ Pake 1980, S. ix; Pake 1985, S. 54; Pake 1992, S. 59.

über das der Computer mit dem Drucker kommunizieren konnte. Obwohl die Entwicklung des Laserdruckers eine der Aufgaben des SSL war, entwickelte es sich im Laufe der siebziger Jahre zu einem zweiten Labor für Computertechnik, allerdings mit einem Schwerpunkt auf der Softwareentwicklung. Eine Trennung von SSL und CSL war ohnehin nur schwer möglich, da viele Projekte in Kooperation der beiden Labors durchgeführt wurden.¹

Die Leitung des *General Science Laboratory* übernahm George Pake persönlich, als Manager des SSL wechselte William F. Gunning von Astrodata zum PARC. Das CSL blieb zunächst ohne formellen Leiter.²

Pakes erste Aufgabe war es, ausgezeichnete und kreative Wissenschaftler zu finden, die das Schlagwort von der »Architektur der Information« mit Leben füllen konnten. Da Pake kein Computerwissenschaftler war, stellte er im September 1970 den ehemaligen Direktor des IPTO Robert W. (Bob) Taylor (Abb. 66, rechts) ein, der seit 1969 an der University of Utah arbeitete und nun den Aufbau der Computerforschung organisieren sollte. Durch seine Tätigkeit für die NASA und ARPA hatte Taylor einen ausgezeichneten Überblick über den Stand der Forschung und besaß beste Kontakte zu allen wichtigen Universitäten und Forschungszentren. Taylor kannte einige besonders talentierte Computerwissenschaftler, die er unbedingt an das PARC verpflichten wollte. Dazu gehörte unter anderem Alan C. Kay (* 1940), der 1969 an der University of Utah promoviert hatte und nun am renommierten *Stanford Artificial Intelligence Laboratory* unter der Leitung von John McCarthy beschäftigt war.³

Im Januar 1971 gelang es Taylor, den größten Teil des Personals der *Berkeley Computer Corp.* abzuwerben, darunter auch Butler Lampson (* 1943) und Charles P. (Chuck) Thacker (* 1943), die als Mitarbeiter des *Project Genie* das Time-Sharing-Betriebssystem für den SDS-940-Computer entwickelt hatten. Alan Kay entschloß sich im April 1971 für einen Wechsel zum PARC. Wenig später kam auch eine Reihe von Douglas Engelbarts Mitarbeitern zum PARC, darunter seine wichtigsten Softwareentwickler Jeff Rulifson und Bill Paxton sowie Engelbarts Stellvertreter und Hardwarespezialist William English. Weitere Mitarbeiter kamen von der Carnegie-Mellon University, von der University of Utah, vom MIT, von BBN und nicht zuletzt vom Xerox Forschungslabor in Webster. Auch die Fachöffentlichkeit nahm langsam zur Kenntnis, daß in Palo Alto etwas außergewöhnliches vor sich ging, und dieser Eindruck machte das PARC für

¹ Pake 1985, S. 55; Smith and Alexander 1988, S. 70. Pake und vor allem Taylor hatten mit Recht auch darauf spekuliert, daß das Budget für Computerforschung durch die Einrichtung zweier Labors mit scheinbar unterschiedlichen Aufgaben größer ausfallen würde, als bei Einrichtung eines einzigen Labors für Computertechnik. Vgl. Goldberg 1998, S. 55 und Email von Alan Kay an den Autor, 13. Mai 1998.

² Pake 1985, S. 56.

³ Smith and Alexander 1988, S. 65ff.



Abbildung 67. Charles P. Thacker (links) und Butler W. Lampson (rechts) im Januar 1986

Nachwuchswissenschaftler noch interessanter.¹ Taylor hatte es geschafft, die besten Computerwissenschaftler des Landes nach Palo Alto zu holen.² Obwohl man sich beim Aufbau des Palo Alto explizit an den Bell Laboratorien orientiert hatte, handelte es sich um ein im Vergleich zu den 20 000 Angestellten der Bell Labs sehr bescheidenes Forschungszentrum mit maximal 360 Beschäftigten.³

Es existierte jedoch noch das Problem, daß Robert Taylor zunächst nur verpflichtet worden war, um das CSL aufzubauen, nicht aber als dessen Leiter vorgesehen war. Obwohl Taylor fast zehn Jahre Erfahrung in der Forschungsförderung und Projektleitung besaß und dabei ein enormes Gespür für erfolgversprechende Projekte entwickelt hatte, war er in Pakes Augen nicht für die Leitung eines Labors geeignet, weil er nicht promoviert war. Da Pake sich von dieser Auffassung nicht abbringen ließ, einigten sie sich schließlich darauf, daß Taylor als *associate manager* für die innere Leitung des Labors verantwortlich sein sollte. Formeller Leiter des Labors wurde Dr. Jerry Elkind, den Taylor bereits als Berater für die

¹ Kay 1996, S. 525f.

² Alan Kay hat diese Tatsache in die griffige Formel gebracht, das PARC habe 76 der 100 besten Computerwissenschaftler beschäftigt. Dabei ignorierte er bewußt die Tatsache, daß das PARC zu Anfang der siebziger Jahre insgesamt weniger als 76 Beschäftigte hatte. Vgl. Smith and Alexander 1988, S. 76; Cringely 1993, S. 99.

³ Stross 1997.



Abbildung 68. *Typische Beanbag Conference.* Teilnehmer von links nach rechts: Jim Mitchell, Ed Fiala, Vikki Parrish, unbekannte Frau, Wesley Clark (externer Berater), Edward Taft

ARPA kennengelernt hatte und der zuvor für Bolt, Beranek and Newman gearbeitet hatte. Da Elkind Ambitionen hatte, ins Management von Xerox aufzusteigen, und die Position des CSL-Managers dafür ein geeignetes Sprungbrett darstellte, willigte er schließlich in das Arrangement ein. Es sollte sich als eines der Probleme des PARC herausstellen, daß eine so starke Persönlichkeit wie Taylor auf diese Weise niemandem außer ihm selbst verantwortlich war.¹

Taylor etablierte einen Führungsstil, der sich stark am Stil der ARPA orientierte und von der Managementlehre als Adhokratie bezeichnet wird.² Es gab nur eine sehr flache Hierarchie, bei der alle Mitarbeiter des Labors formell gleichgestellt waren. Taylor selbst verstand sich als *Primus inter pares*, der als Katalysator von neuen Ideen fungierte und für die Kommunikation zwischen den Mitarbeitern verantwortlich war. Bei dieser Aufgabe hatte er stets das Ziel im Auge, die Möglichkeiten des interaktiven und vernetzten Computerbetriebs voranzutreiben. Insgesamt organisierte und leitete Taylor das CSL so, als handle es sich um eines

¹ Smith and Alexander 1988, S. 71ff.

² Toffler 1983, S. 93ff.; Mintzberg 1992, S. 335ff.

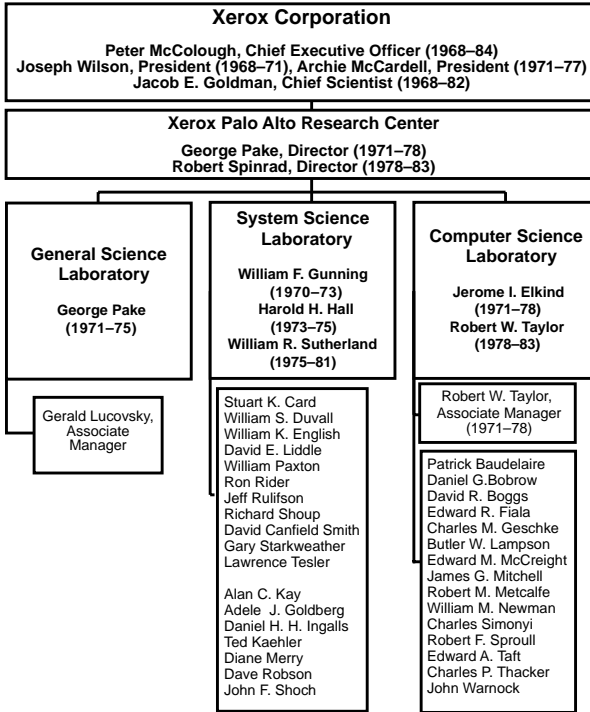


Abbildung 69. Gliederung des Xerox Palo Alto Research Center in den siebziger Jahren, Labors, Manager und ausgewählte Mitarbeiter.

der von der ARPA geförderten *Centers of Excellence*. Am prägnantesten zeigte sich dieser Stil in den wöchentlichen Besprechungen, die – ähnlich wie die regelmäßigen Treffen der ARPA-Wissenschaftler – in einer informellen Atmosphäre stattfanden und bei denen stets ein anderer Mitarbeiter die Diskussionsleitung übernahm und wie der Geber (*dealer*) beim Kartenspiel die Regeln bestimmt. Deshalb nannte man die Besprechungen *dealer meetings* oder auch nach den bevorzugten Sitzgelegenheiten *beanbag conferences* (Abb. 68).¹

6.2 Time-Sharing: Kontinuität am PARC

Die hochkarätige Gruppe von Computerwissenschaftlern, die Taylor 1970/71 zusammengestellt hatte, war zunächst weniger den ambitionierten aber noch sehr

¹ Smith and Alexander 1988, S. 78ff; Kearns und Nadler 1993, S. 109; Newell 1991a, Tape 6, Side A.

unschärferen Zielen verpflichtet, die Peter McColough mit dem neuen Forschungslabor verbunden hatte. Die meisten von ihnen hatten während der sechziger an einem der ARPA-Projekte mitgearbeitet, und für sie stellte das PARC die Möglichkeit dar, ihre Forschungsschwerpunkte auch in Zeiten rückläufiger militärischer Förderung ungehindert weiterzuführen.¹

Maxc

In Fortsetzung des ARPA-Forschungsprogramms der sechziger Jahre plante man umgehend den Aufbau eines eigenen Time-Sharing-Systems, das zu dieser Zeit für die Arbeit eines Computerlabors unabdingbar war. Dabei favorisierten die Mitarbeiter des CSL die Anschaffung eines PDP-10 mit dem TENEX Betriebssystem, um die Interlisp-Programmiersprache verwenden und um problemlos am entstehenden ARPANET teilnehmen zu können.² Die Anschaffung eines Computers eines konkurrierenden Unternehmens kam jedoch für das Management nicht in Frage. Sie schlugen vor, stattdessen den von XDS hergestellten Sigma-7 Computer einzusetzen. Da dieser nicht als Time-Sharing-Computer konzipiert war und die Entwicklung eines »vernünftigen Betriebssystems« nach der Einschätzung von Thacker und Lampson mindestens drei Jahre gedauert hätte, entschlossen sie sich, ihren eigenen Computer zu entwickeln, der kompatibel zum PDP-10 sein sollte und auf dem auch TENEX lauffähig sein würde.³

So beschäftigten sich die Mitglieder des CSL unter der Leitung von Lampson und Thacker bis Mitte 1972 vor allem mit dem Bau des *Multiple Access Xerox Computers* (Maxc)⁴ und der Anpassung bzw. Portierung von Systemsoftware und Programmiersprachen. Für die Realisierung des Hauptspeichers wurden im Gegensatz zum PDP-10 allerdings keine Kernspeicher, sondern Halbleiterspeicherchips verwendet, die sich um 1970 durchzusetzen begannen.⁵

Ebenfalls im Trend der Zeit lag die Mikroprogrammierbarkeit des Maxc. Dies bedeutet, daß die Maschinenbefehle des Computers nicht mehr durch ein festverdrahtetes, also in Hardware realisiertes Steuerwerk dekodiert werden, sondern jeder Befehl aus einer Kette elementarer (Mikro-)Operationen, dem sogenannten Mikroprogramm aufgebaut ist. So ist es möglich, den Befehlssatz eines Computers problemlos durch eine Veränderung des Mikroprogramms zu erweitern oder durch einen gänzlich anderen zu ersetzen. Dieses schon 1953 von Maurice Wilkes

¹ Email von Alan C. Kay an den Autor, 13. Januar 1998.

² Fiala 1978, S. 57; Memorandum von Chuck Thacker an Dave Allison über »Contribution to Research Report to Management, 21. April 1972, SUL/EC, 17-7.

³ Kay 1996, S. 526.

⁴ Angeblich entstand die Bezeichnung Maxc auch als sarkastische Anspielung auf den Leiter von XDS Max Palevsky, der sich lange Zeit gegen den Bau von Time-Sharing-Computern gewehrt hatte. Vgl. Smith and Alexander 1988, S. 146f.

⁵ Bylinsky 1971; Fiala 1978, S. 62.

entwickelte Konzept¹ erfreute sich seit den sechziger Jahren wachsender Popularität. Es wurde bei der Rechnerserie System/360 von IBM dazu verwendet, um bei den verschiedenen leistungsstarken Maschinen dieser Serie einen einheitlichen Befehlssatz zu realisieren.² Beim Maxc wurden beispielsweise neben dem PDP-10-Befehlssatz auch Befehle implementiert, die speziell auf die Anforderungen von Lisp abgestimmt waren.³ Das Resultat war ein Computer, der zwar kompatibel mit dem PDP-10 war, aber schneller und technologisch fortgeschrittener als sein Vorbild.⁴

Der Maxc war während der siebziger Jahre – auch nach der Entwicklung der Alto-Workstation – ein wichtiges Mittel zur zentralen Datenspeicherung und Kommunikation und ein wichtiges Arbeitsgerät bei der Entwicklung von Lisp-Programmen. Innerhalb des lokalen Rechnernetzes fungierten die beiden Maxc als zentrale Dateiserver und über ihn war das PARC auch an das überregionale ARPANET angeschlossen.⁵ In einer Zeit, da die Entwicklung von Time-Sharing-Systemen weitgehend abgeschlossen war und erste kommerzielle Systeme auf den Markt kamen, ging auch der Maxc-Computer des PARC nicht über den Stand der Technik hinaus.

POLOS

Während das *Computer Science Lab* mit dem Maxc seinen eigenen Time-Sharing-Computer entwickelte, wurde im *Systems Science Lab* seit 1971 ein erstes Büroinformationssystem entworfen, das *PARC On-Line Office System* (POLOS). Das Entwicklungsteam setzte sich vor allem aus ehemaligen Mitarbeitern Engelbarts zusammen und wurde von William English geleitet. Für sie war POLOS die Möglichkeit, ihre Auffassung von Engelbarts Vision in einem neuen Umfeld zu realisieren.⁶ So brachten sie nicht nur die Maus und die Einhandtastatur, sondern auch Engelbarts Konzept der hierarchischen Dokumentenorganisation beim PARC ein sowie die Überzeugung, daß das Time-Sharing-Konzept der richtige, wenn nicht sogar der einzig mögliche Realisierungsansatz sei. Da 1971 bereits offensichtlich war, daß ein komplexes System wie NLS bzw. POLOS nicht ohne Probleme auf einem zentralen Rechner installiert werden konnte, griff Englishs Gruppe auf das beim ARPANET praktizierte Verfahren zurück, bei der die Ressourcen in einem Netzwerk verteilt sind.

¹ Wilkes and Stringer 1953.

² Wilkes 1969; Klar 1983, S. 206f. Je leistungsfähiger und teurer ein Computer des Systems/360 war, desto mehr Befehle wurden durch (schnelle) Hardware realisiert.

³ Fiala 1978, S. 63f.

⁴ Fiala 1978, insb. S. 59.

⁵ Goldberg 1988, S. 338f.

⁶ Email von Bill Duvall an den Autor, 28. März 1998.

6. Xerox PARC und die Architektur der Information

POLOS bestand aus bis zu 15 miteinander vernetzten Nova 800 Minicomputern von Data General¹, die über sogenannte *Multiprocessor Communication Adapter* (MCA) sternförmig miteinander verbunden waren. Wie der Name bereits andeutet waren die von Data General als Standardperipheriegeräte verkauften MCAs nicht für den Aufbau eines Computernetzes, sondern eines Mehrprozessorsystems mit erhöhter Rechenleistung gedacht. Aus diesem Grund mußten die Novas mit dicken, 16-adrigen Kabeln verbunden werden, deren Länge auf 12 Meter beschränkt war.²

Innerhalb dieses Netzes war z. B. einer der Computer für die Speicherung und Verwaltung der Dokumente zuständig, ein anderer für die Formatierung von Texten, ein weiterer für die Druckaufbereitung, usw. Die Benutzer von POLOS saßen an hochauflösenden Bildschirmterminals und waren gleichzeitig mit *allen* Rechnern verbunden.³

Diese technisch alles andere als konservative Architektur stellte besonders hohe Anforderungen an das Betriebssystem. Bei einem herkömmlichen Time-Sharing-System hat das Betriebssystem die Aufgabe, die zur Verfügung stehenden Hardwareressourcen so zu verwalten, daß alle Benutzer reibungslos arbeiten können, ohne sich zu behindern. Bei POLOS war der Komplexitätsgrad der Aufgaben noch um einiges größer. Das Betriebssystem mußte nämlich nicht nur die Eingaben *verschiedener Nutzer* verwalten, sondern auch *verschiedene Computeroperationen* (z. B. Eingabe, Ausgabe, Speicher) und *verschiedene Büروفunktionen* (z. B. Editieren, Formatieren, Drucken). Obwohl diese Aufgabe von Bill Duvalls POGOS-Betriebssystem gemeistert wurde, war das Antwortverhalten von POLOS genauso wenig vorhersagbar oder steuerbar wie bei anderen Time-Sharing-Systemen. Auch die Zuverlässigkeit des Netzes war nicht so groß, wie man es sich für die Datensicherheit eines Büroinformationssystems wünschte.⁴

Schließlich hatte POLOS mit den gleichen Akzeptanzproblemen zu kämpfen wie NLS. Wie Engelbart war auch English davon überzeugt, daß die Arbeit mit hierarchischen Dokumenten letztlich bessere Memos, Briefe und Forschungsberichte hervorbringen würde. Und auch bei POLOS wurden wie bei NLS alle Systembefehle mittels Tastaturkürzel eingegeben.

Mit dem Verlagshaus Ginn & Company, einem Tochterunternehmen von Xerox, hatte man 1974 schließlich einen ersten potentiellen Nutzer von POLOS gefunden, der auf diese Weise die Durchlaufzeiten vom Eingang des Manuskripts

¹ Data General war erst 1968 vom ehemaligen DEC-Mitarbeiter Edson de Castro gegründet worden. Die Nova, ihr erstes Produkt, galt für seine Zeit als deutlich fortschrittlicher als die vergleichbaren Maschinen von DEC und war mit einem Einführungspreis von 8 000 \$ außerdem erheblich billiger. Vgl. Kidder 1984, S. 22; Saxenian 1994, S. 96.

² Metcalfe 1994, S. 83; Email von David Boggs an den Autor, 23. April 1998.

³ Smith and Alexander 1988, S. 109f.

⁴ Smith and Alexander 1988, S. 109; Duvall 1976; Metcalfe 1994, S. 83.

bis zum gedruckten Buch optimieren wollten. Aber schon bald nachdem mit der Anpassung von POLOS an die Belange von Ginn & Co. begonnen worden war, stellte sich heraus, daß die Textverarbeitung nicht den Anforderungen eines Verlags gerecht wurde, weil es zu kompliziert in der Bedienung war.¹

Mit der Ablehnung durch Ginn & Co. war auch Schicksal des POLOS-Projekts besiegelt: es wurde 1975 eingestellt. Auch wenn POLOS eine technische Sackgasse darstellte, fanden eine Reihe von Programmen und Konzepten nach 1975 Eingang in Xerox' spätere Büroautomationssysteme.²

Der Aufbau eines großen Time-Sharing-Systems war allerdings um 1970 nicht mehr die einzige ökonomisch sinnvolle Möglichkeit, einen interaktiven Computerbetrieb zu realisieren. Im Zuge der technologischen Entwicklung in der Halbleitertechnik eröffneten sich neue Möglichkeiten.

6.3 Das Dynabook: Ein Computer für Kinder in jedem Alter

Moores Gesetz und die Möglichkeiten der Mikroelektronik

1958/59 hatten Jack Kilby von Texas Instruments und Robert Noyce von Fairchild Semiconductors im Abstand von wenigen Monaten unabhängig voneinander die ersten integrierten Schaltkreise entwickelt. Dabei wurden erstmals in einem kontinuierlichen Prozeß die Bauelemente und die sie verbindenden Leiterbahnen einer Schaltung im Substrat des Halbleiters selbst aufgebaut. Damit wurde es möglich, vielverwendete, aber bislang diskret aufgebaute Grundschaltungen der Computertechnik (Flipflops, Register, Addierer, etc.) auf einer Fläche von nur wenigen Quadratmillimetern zu integrieren. Mit dieser Technologie entfiel dann auch die arbeitsintensive Verdrahtung der Schaltungen, durch die die Hardwareentwicklung um 1960 erheblich gehemmt wurde. Schließlich führten Fortschritte in der Produktionstechnik zu einer stetigen Verkleinerung der Bauelemente und zur ständigen Erhöhung der Packungsdichte, so daß Ende der sechziger Jahre die komplette Zentraleinheit eines Computers in Form eines Integrierten Schaltkreises realisiert werden konnte (vgl. dazu Kapitel 7).³

Nachdem die ersten Kinderkrankheiten, etwa das Problem der Reproduzierbarkeit, überwunden waren, wurde auch die Produktion von Halbleiterchips vom Mechanismus der Skalenökonomie bestimmt: Hohe Stückzahlen bei gleichzeitig sinkendem Ausschuß führten zu sinkenden Preisen. Gleichzeitig sank der Preis

¹ Smith and Alexander 1988, S. 105f., 110; Email von Larry Tesler an den Autor, 18. April 1998; Email von Tim Mott an den Autor, 18. Mai 1998.

² Email von Larry Tesler an den Autor, 18. April 1998

³ Kaiser 1996, S. 132ff. Zur Geschichte der Mikroelektronik vergleiche u. a. Braun and Macdonald 1982, Eckert und Schubert 1986, Molina 1989 und Morris 1990.

6. Xerox PARC und die Architektur der Information

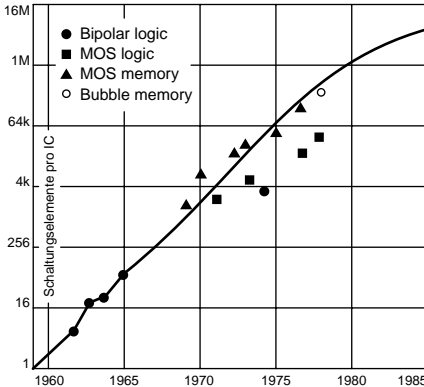


Abbildung 70. Gordon Moores »Gesetz« in einer Darstellung von 1979. Die vier Werte für die Jahre 1962 bis 1965 waren Grundlage für die erste Formulierung des »Gesetzes« im Jahre 1965.

pro Transistor bei immer höherer Integration weiter ab. Mitte der sechziger Jahre formulierte Gordon Moore, einer der Gründer von Intel, das nach ihm benannte ökonomische Gesetz der Mikroelektronik. Demnach kann man alle zwei Jahre eine Verdoppelung der auf einem Chip integrierten Transistorfunktionen erwarten (Abb. 70).¹ Auf die großen Computerhersteller wie IBM oder auch DEC und ihre Produktpalette hatte diese Feststellung zunächst keine große Wirkung. Für sie bedeutete Moores Gesetz, daß ihre Art von Computern noch leistungsfähiger werden konnten. Sie übersahen dabei die Möglichkeiten, die die fortschreitende Miniaturisierung beim Bau kleiner und kleinster Computer eröffnete.²

Auch den Wissenschaftlern am PARC war dieser Trend bekannt und so machte Robert Taylor im Frühjahr 1971 den unorthodoxen Vorschlag, das Time-Sharing-Konzept aufzugeben und jedem Anwender seinen *eigenen* Computer zur Verfügung zu stellen.³ Da Taylors Vorstellungen allerdings noch sehr vage und integrierte Schaltkreise noch sehr teuer waren, wurde sein Vorschlag zunächst zurückgestellt. Erst im Herbst 1972 begann man sich wieder ernsthaft mit der Idee auseinanderzusetzen, nachdem der Bau des Maxc weitgehend abgeschlossen war und die Entwicklung von POLOS mit erheblichen Problemen zu kämpfen hatte.⁴ Dabei konnte man nun auf Konzepte von Alan Kay zurückgreifen, der sich bereits seit Jahren mit der Frage beschäftigte, wie ein persönlicher Computer aussehen und was er leisten können sollte.

¹ Moore 1965.

² Kay et al. 1994.

³ Smith and Alexander 1988, S. 71; Kay 1971; Kay 1972b; Pake 1985, S. 56.

⁴ Email von Butler Lampson an den Autor, 9. Januar 1998.