



universität
wien

Diplomarbeit

Titel der Diplomarbeit

„Facing the Interface“

Die grafische Benutzeroberfläche als dominierendes Paradigma der
Mensch-Computer-Interaktion

Verfasserin

Ines Steixner

angestrebter akademischer Grad

Magistra der Philosophie (Mag. phil)

Wien, 2011

Studienkennzahl lt. Studienblatt: A 317

Studienrichtung lt. Studienblatt: Theater-, Film- und Medienwissenschaft

Betreuer: Univ.-Prof. Dr. Klemens Gruber

Inhaltsverzeichnis

1. Einleitung	1
2. Das Interface - terminologische Bestimmung	5
3. Das grafische User-Interface - „Desktop-WIMP“	8
4. Das Interface als Kategorie des Computerbildes	12
4.1. Interaktion mit Bildern bzw. interaktive Bilder.....	14
4.2. Bitmapping - Visualisierung digitaler Muster.....	17
5. Historische Voraussetzungen für die Entwicklung des grafischen User-Interface	21
5.1. Inspirationsquelle Vannevar Bush - „As we may think“	21
5.2. SAGE - der militärische Ursprung der Bedienkonsolen.....	23
5.3. Joseph Lickliders „Man-Computer-Symbiosis“ als impulsgebende Leitidee.....	25
5.4. ARPA - militärische Forschungsgelder für Grundlagenentwicklung	28
5.5. Time-Sharing	29
5.6. Ivan Sutherlands Sketchpad - grafische Revolution der Mensch-Computer-Interaktion.	32
5.7. Douglas Engelbart - „Augmenting Human Intellect“	35
5.7.1. Praktische Realisierung des Hardware-Interfaces - die Entwicklung der Maus.....	39
5.7.2. NLS - die Demonstration	43
6. Die Entwicklung der grafischen Benutzeroberfläche	46
6.1. Xerox PARC - die institutionelle Rahmenbedingung	46
6.2. Alan Kay - FLEX und das „Dynabook“	47
6.3. Seymour Papert und LOGO	50
6.4. Doing with Images makes Symbols.....	52
6.5. Der „Alto“	55
6.5.1. Das „interim Dynabook“ und „Smalltalk“.....	57
6.5.2. Entwicklung überlappender Fenster und der Beginn der Benutzerillusion.....	59
6.6. Xerox „Star“ - der Anfang der Automation	62
6.7. Apple Lisa und Macintosh	66
7. Interface-Kultur	73
8. Resumée.....	75
9. Quellenverzeichnis.....	80
10. Danksagung.....	87
11. Abstract.....	88

1. Einleitung

Wesentliche Aspekte der westlichen Kultur sind von den Errungenschaften digitaler Informations- und Kommunikationstechnologien durchdrungen und geprägt. Der Personal Computer affiziert - vor allem seit der Etablierung des World Wide Webs Anfang der 1990er Jahre - vehement subjektive und kollektive Erfahrungs-, Handlungs- und Wissensräume sowie unsere Kommunikationsmodalitäten. Insbesondere seit der Verbreitung von Laptops lässt sich eine intensive, fast schon intime Koppelung des Menschen an digitale Technologien diagnostizieren. Es scheint, als ob die Benutzung von Computern heutzutage schon so selbstverständlich ist wie Schreiben und Lesen - in manchen Fällen sogar selbstverständlicher. Die Schnittstellen zu diesen technischen Artefakten scheinen kaum noch als Schnitt, als Grenze wahrgenommen zu werden. Mensch-Computer-Interaktion ist weit mehr als nur ein wissenschaftlicher Terminus oder ein interdisziplinärer Forschungsbereich, nämlich ein beinahe schon automatisierter und soziokulturell höchst relevanter Gestus der digitalen Alltagskultur.

Angesichts der unüberschaubaren Menge an kultur- bzw. medienwissenschaftlichen Publikationen zur Informationsgesellschaft und zu neuen, elektronischen Medien verwundert die rare Auswahl an Werken, die sich dem Phänomen des Interface widmen. Die soziokulturelle Performanz von Computern und den diversen Dienstleistungen wie dem World Wide Web hängt jedoch wesentlich von den Möglichkeiten und Einschränkungen ihrer Benutzung ab. Wie die Luft zum Atmen, sind Interfaces, also die Schnittstellen zwischen Mensch und Computer, unerlässlich für die Rezeption aller Inhalte und Leistungen der digitalen Medienlandschaft. Sie transformieren die - sowohl haptisch wie auch intellektuell - unbegreifbaren Codes des digitalen, submedialen Raums in für den Benutzer verständliche und manipulierbare Informationen und bestimmen somit über die Poesie der Interaktion mit dem Computer und den verfügbaren Anwendungen.

Wirft man einen Blick zurück zum Ursprung des PCs als Massenmedium, so scheint die Entwicklung der grafischen Benutzeroberfläche eine fundamentale Triebkraft der Verbreitung und Akzeptanz des Computers in der Massenkultur, wenn nicht gar deren *conditio sine qua non* gewesen zu sein. Erst die Ikonografisierung von Rechenprozessen und Befehlsstrukturen sowie die Möglichkeit der direkten Manipulation von bildlich repräsentierten Bildschirmhalten - kurz, eine leicht erlernbare und intuitive Computerbedienung ließen den Computer massentauglich werden. Die Magie der Interaktion mit Bildern löste die komplizierte und nur Computerexperten vorbehaltene Steuerung und Kontrolle des Computers mittels schriftlicher

Befehle ab und lies diesen von einem technokratischen Laborgegenstand zu einem integralen Bestandteil der Medienkultur avancieren. Seitdem prägt die grafische Benutzeroberfläche mit ihrer standardisierten Ästhetik wesentliche Aspekte des digitalen Alltags. Seit gut zwei Jahrzehnten sind überlappende Fenster, bunte Icons und virtuelle Schreibtische ebenso präsent wie »click & point«, »drag & drop« oder Doppel- und Rechtsklick, die sich als technikgenuine sowie technikfunktionale Standardgesten einbürgerten. Alle Praktiken, in deren Mittelpunkt der Personal Computer steht, konvergieren also um dieselben virtuellen Metaphern und Benutzungsmodalitäten. Aus diesem Grund erscheint eine nähere Untersuchung des grafischen User-Interface als dominierendes Paradigma der Mensch-Computer-Interaktion angebracht. Die Arbeit zielt dabei weder auf eine detaillierte Beschreibung grafischer Benutzeroberflächen einzelner Anwendungen und deren Design-Strategien noch auf einen Vergleich unterschiedlicher Betriebssysteme ab, sondern interessiert sich vielmehr für deren gemeinsame Grundlagen und vor allem deren ideen- und technikgeschichtliche Genese, um somit das Bewusstsein für die Hintergründe eines alltäglichen und omnipräsenten Phänomens zu schärfen. Wie bei jeder Instrumentalisierung bzw. Externalisierung des Denkens verschmelzen wir mit den dafür zur Verfügung stehenden Werkzeugen innerhalb ihres Funktions- und Handlungsreiches, wir fließen in sie ein, wie sie umgekehrt unser Denken und unsere Deutungsschemata durchdringen. „Unser Schreibzeug arbeitet mit an unseren Gedanken“¹ lautet die bekannte Feststellung Friedrich Nietzsches. Die vorliegende Arbeit reflektiert in diesem Sinne auch die Voraussetzung ihrer eigenen Entstehung und versteht sich somit als Beitrag zur Diskussion über die technisch-mediale Bedingtheit von Wissen und Bedeutung.

Die forschungsleitenden Fragestellungen können folgendermaßen umrissen werden:

-Auf welchen gemeinsamen Fundamenten baut die Interaktion mit kommerziellen PCs bzw. Laptops auf? Was sind die konstitutiven Elemente heutiger User-Interfaces und auf welchen technischen Voraussetzungen und Ideen beruhen sie?

-Entlang welcher Diskurse entwickelten sich - die noch vor 60 Jahren raumfüllenden - technokratischen „Numbercruncher“ zu den heutigen benutzerfreundlichen multimedialen PCs? Was waren wegweisende Impulse und Errungenschaften, die zur Entwicklung der grafischen Be-

¹Brief von Friedrich Nietzsche an Heinrich Köselitz in Venedig, Februar 1882 (Typoskript), In: Nietzsche, Friedrich (2002): *Schreibmaschinentexte*, S. 18.

nutzeroberfläche und somit auch zu einer der einschneidendsten medialen Zäsuren beigetragen haben?

Zu Beginn der Arbeit, in Kapitel 1, erfolgt eine Klärung der terminologischen Voraussetzungen. Kapitel 2 widmet sich der Frage: Was ist ein Interface? Welche Funktionen übernimmt es und welche Probleme ergeben sich in der Diskussion über Interfaces? In Kapitel 3 wird die grafische Benutzeroberfläche als am weitesten verbreitete Form von Interfaces und dominierendes Interaktionsparadigma untersucht. Was sind deren konstitutive Bestandteile und welche Interaktionskonventionen bedingen dieselben? Kapitel 4 untersucht das Interface als eine Kategorie des Computerbildes. Die Dichotomie sowie das Zusammenspiel zwischen phänomenologischer Oberfläche und dem codiertem Hintergrund ist grundlegend für das Verständnis der Mensch-Computer-Interaktion. Die Annäherung an das Interface als Bild gewährt einen guten Einblick in deren Fundamente. Wie lässt sich die grafische Benutzeroberfläche als Computerbild fassen? Was unterscheidet sie von anderen Bildern? Worauf baut die Interaktion mit Bildern zur Steuerung des Computers auf und wie erfolgt die Visualisierung von immateriellen digital gespeicherten Informationen? Da sich die Untersuchung mit einem höchst präsenten Phänomen auseinandersetzt, welches erst in den letzten zwei Jahrzehnten entscheidend an Bedeutung gewonnen hat, erscheint ein Rückblick auf die historischen Wurzeln zugunsten einer Einbettung in aktuelle und zukünftige Diskussionen angebracht. Der Hauptfokus besteht daher in der Untersuchung der historischen Entwicklung von Computerbenutzung, deren vorläufiger Endpunkt die grafische Benutzeroberfläche darstellt.² Es wird versucht prägende Leitbilder, institutionelle und technische Rahmenbedingungen sowie konkrete Realisierungsprozesse zu parallelisieren. Kapitel 5 diskutiert die Anfangszeit interaktiver Computerbenutzung, deren Ursprung in der militärischen Luftraumüberwachung liegt. Die sonst übliche Verwendung des Computers beschränkte sich auf Stapelverarbeitung, die noch nicht interaktiv war. Die Forderung nach sowie die Entwicklung von Echtzeit-Systemen beruhte auf dem Bewusstsein, dass der Computer nicht nur bereits vorgegebene Rechnungen durchführen, sondern den Menschen auch beim Denken und Problemlösen aktiv unterstützen könnte und sollte, was jedoch einen dialogischen Austausch erforderte. Mit dem Aufkommen von Time-Sharing-Systemen in den 60ern bot sich zum ersten Mal die Möglichkeit einer interaktiven Kom-

²Publikationen, die sich explizit mit der Genese des grafischen Interface beschäftigen, halten sich noch in Grenzen. Die Autorin orientierte sich v.a. an zwei Werken: Michael Friedewalds *Der Computer als Werkzeug und Medium* (1999) und dem von Hans-Dieter Hellige herausgegebenen Sammelband *Mensch-Computer-Interface: Zur Geschichte und Zukunft der Computerbedienung* (2008). Es wurde weiters viel Wert auf Originalquellen gelegt, die großteils nur über digitale Archive zugänglich sind.

munikation. Unter diesen Bedingungen erfolgte die Entwicklung wegweisender soft- und hardwaretechnischer Innovationen, die großen Einfluss auf die Entstehung der grafischen Benutzeroberfläche hatten. Allerdings waren Computer noch immer Expertensysteme und die Bedienung alles andere als leicht. Anfang der 70er Jahre veränderten sich sowohl technologische Randbedingungen wie auch die Art und Weise, wie der Computer betrachtet und für wen er gestaltet wurde. Die Firma Xerox wollte ein elektronisches Büro der Zukunft entwickeln, Alan Kay hatte die Vision vom Computer als persönlichem (Meta)medium. Beide Ideen konvergierten um die Forderung einer intuitiven und leicht erlernbaren Benutzung. Kapitel 6 verdeutlicht die Hintergründe des Entstehens der grafischen Benutzeroberfläche, welche Gestaltungsprämissen ihr zugrunde liegen und wie diese nach und nach ihre Form annahm, bis hin zur endgültigen Kommerzialisierung des PCs. Die Entwicklung ist natürlich noch lange nicht abgeschlossen und es ist absehbar, dass die Gestaltung von Interfaces auch in Zukunft die Art und Weise der Rezeption und Distribution digital verwalteter Information bestimmen wird. Kapitel 7 will daher zum Abschluss den kulturtheoretischen Aspekt von Interfaces abermals akzentuieren um auf deren Wichtigkeit und Virulenz als Untersuchungsgegenstand hinzuweisen.

2. Das Interface - terminologische Bestimmung

Der Begriff Interface meint grundsätzlich „den Punkt einer Begegnung oder einer Koppelung zwischen zwei oder mehr Systemen und/oder deren Grenzen zueinander.“³ Ursprünglich, und zwar ab Ende des 19. Jahrhunderts, bezeichnet ein Interface die Phasengrenze zweier Komponenten in der Chemie.⁴ In der Informatik, wo sich der Begriff erst Mitte des 20. Jahrhunderts einbürgerte, markieren Hardwareinterfaces die physikalischen Verbindungspunkte zwischen Peripheriegeräten (z.B. Drucker, Scanner,...) und dem CPU bzw. zwischen zwei Hardwarekomponenten. Um den Menschen erweitert gerinnt das Interface zu einem User-Interface, welches alle ihm zur Verfügung stehenden Komponenten eines Gebrauchsgegenstandes zum Zweck einer zielgerichteten Benutzung umfasst. Der Kern des User-Interface ist daher die Austauschbeziehung zwischen dem Menschen und seiner artifiziellen Umwelt. Insofern gibt es User-Interfaces sowie User-Interface-Design schon ca. 2,4 Millionen Jahren - seit der ersten zielgerichteten Bearbeitung der Umwelt zum Zwecke einer Werkzeugherstellung.

Die Drehregler eines Radios erlauben eine direkte Verbindung der materiellen und mechanischen zu den elektronischen Komponenten, welche beispielsweise die Lautstärke oder die Funkwellenfrequenz regeln - Funktionalität und Benutzung sind beim Radio wie auch beim Faustkeil und beim Hammer klar definiert. Der Computer hingegen ist eine informationsverarbeitende programmierbare Maschine, Funktionalität und Benutzung sind nicht a priori festgelegt. Die Grenzen der Funktionalität eines Computers sind die Grenzen von Berechenbarkeit und den dazu nötigen Rechen- und Speicherkapazitäten. Mit den Prozessen der Maschinsprache, also jenem Niveau unfassbar schnell getakteter Folgen von logischen Operationen in Form binärer (bzw. hexadezimaler) Codereihen - der *lingua franca* des Computers - kann der Mensch nichts anfangen. Zwei unterschiedliche informationsverarbeitende Systeme, der Computer und der Mensch, müssen somit über ein geeignetes Interface aneinander angepasst werden. Erst durch eine Umwandlung der elektronischen Signale in Informationen, die für den Menschen wahrnehmbar sind, kann ein Informationsaustausch erfolgen. Um der ursprünglichen Konnotation als Grenzfläche gerecht zu werden, definiert Frank Hartmann das Interface als eine „Projektionsfläche, die sich zwischen Menschen mit ihren Wahrnehmungs-, Steuerungs- und Kommunikationsprozessen und die Maschinen mit ihren Datenverarbei-

³Halbach, Wulf R. (1994): *Interfaces. Medien- und Kommunikationstheoretische Elemente einer Interface-Theorie*, S. 168.

⁴Vgl.: Hellige, Hans-Dieter (2008): „Krisen- und Innovationsphasen in der Mensch-Computer-Interaktion“, S. 13.

tungsprozessen und Datenströmen schiebt.”⁵ Das Mensch-Computer-Interface konnotiert also generell jede Art der Interaktion des Menschen mit dem Computer. Es umfasst die physischen Komponenten möglicher Ein- und Ausgabegeräte (Bildschirm, Maus, Tastatur, Drucker, Kabel,...) wie auch den semantischen Kontext der Austauschbeziehung zwischen Mensch und Computer. Somit sichert es sowohl in technischer als auch in symbolischer Hinsicht den wechselseitigen Informationsaustausch. Mensch-Computer-Interaktion ist daher, kurz gesagt, Kommunikation mit dem System durch ein Interface. Als Übersetzungsinstant zwischen informationstechnischen Prozessen und der menschlichen Perzeptionsfähigkeit markiert es

„a contact point where software links the human user to the computer processors. This is the mysterious, nonmaterial point where electronic signals become information. It is our interaction with software that creates an interface.”⁶

Die Gestaltung des Interfaces definiert also die Nutzungsmöglichkeit und die Poesie der Interaktion zwischen Mensch und Computer. Im Forschungsfeld der HCI (Human Computer Interaction) konvergieren u.a. Erkenntnisse der Informatik, der Ergonomie, des Produkt- und Softwaredesigns, der Soziologie, der Arbeitswissenschaft sowie der Kognitionswissenschaft zu einer benutzergerechten Gestaltung von interaktiven Systemen und deren Interfaces. In diesem Sinne schlägt Thierry Bardini vor, das Interface als „the space in which user and designer meet” zu definieren, denn schließlich ist „the user's quest for the truth of the interaction [...] directed [...] by the designer of the interface.”⁷ Bardini erweitert somit das Interface von seiner Flächenkonnotation in den Raum, was angesichts der medialen und kommunikativen Eigenschaften des Computers durchaus sinnvoll ist. Insofern etabliert das Interface einen gemeinsamen Handlungskontext bzw. Handlungsraum, indem sowohl der Computer wie auch der Benutzer als Akteur fungiert.⁸

Es kann festgestellt werden, dass sich das Interface in seiner Funktion zwar einigermaßen eingrenzen lässt, jedoch mehrere Konnotationen bereithält, die je nach Kontext eine unterschiedliche Akzentuierung erlauben. Besonders deutlich wird dieser Sachverhalt, wenn der englische Terminus ins Deutsche übersetzt wird. Hans-Joachim Maass, der Übersetzer von Steve Johnsons *Interface-Culture*, schreibt zu diesem Problem:

„Der Begriff Interface lässt sich nicht ohne weiters übersetzen. Einerseits meint er die Mensch-Maschine-Schnittstelle, das Bindeglied, das uns Zugang zu den binären Daten der Computerwelt

⁵Hartmann, Frank (2006): *Globale Medienkultur: Technik, Geschichte, Theorien*, S. 191.

⁶Heim, Michael (1993): *The Metaphysics of Virtual Reality*, S. 77.

⁷Bardini, Thierry (1997): „Bridging the Gulfs. From Hypertext to Cyberspace”, <http://jcmc.indiana.edu/vol3/issue2/bardini.html#rref36> (08.11.2010)

⁸Vgl.: Laurel, Brenda (1991): *Computers as Theatre*, S. 7.

ermöglicht, andererseits benennt er das Medium, d.h. die Benutzeroberfläche, mit der wir auf Computerdaten zugreifen.”⁹

Maass Anmerkung verdeutlicht ausgezeichnet die Diskrepanz zwischen phänomenologischer Oberfläche - der sinnlichen Interaktionsebene des Benutzers mit dem System - und dem codierten Hintergrund - der technischen Ebene des Interfaces. Den verschiedenen Optionen auf der Darstellungsebene entspricht eine technische Kodierung der Daten im submedialen Raum. Natürlich sind diese stets mit konkreten Entscheidungen der Systementwickler verbunden - insofern lässt sich das Interface immer auch als eine mediale Inszenierungsstrategie verstehen. In einschlägiger Literatur findet sich eine Vielfalt an Übersetzungen, wie z.B. Benutzerschnittstelle, Schnittstelle, Nutzungsoberfläche, Benutzeroberfläche etc. Es gibt keine standardisierte Übersetzung. Außerdem scheint es, als ob der Interface-Begriff in der englischsprachigen Literatur, sofern nicht explizit ausgezeichnet, beinahe schon selbstverständlich als Synonym für das „graphical User-Interface“ gebraucht wird. Auch diese Arbeit verfolgt keine einheitliche Terminologie, sondern verwendet entweder bewusst den unscharfen, jedoch weiträumigeren Interface-Begriff bzw. macht sich die Möglichkeit einer kontextabhängigen Übersetzung zunutze. Der Terminus der Benutzeroberfläche scheint etwa dann angebracht zu sein, wenn primär von der phänomenologischen Erscheinung des Computers für den Benutzer die Rede ist, wobei die den Sinnen verborgene Ebene stets mitbedacht werden muss.

⁹Maass zit. in: Johnson, Steven (1999): *Interface-Culture. Wie neue Technologien Kreativität und Kommunikation verändern*, S. 15.

3. Das grafische User-Interface - „Desktop-WIMP“

Seit dem kommerziellen Durchbruch der grafischen Benutzeroberfläche durch Apples Macintosh 1984 (Microsoft folgte kurz darauf) dominiert diese die Mensch-Computer-Interaktion. Mit Lev Manovich lässt sich sogleich deren kulturelle Relevanz hervorheben und der Fokus dieser Arbeit auf die Fundamente und Genese der grafischen Betriebssystemoberflächen begründen:

„In semiotic terms, the computer interface acts as a code which carries cultural messages in a variety of media. When you use the Internet, everything you access - texts, music, video, navigable spaces - passes through the interface of the browser and then, in its turn, the interface of the OS [Operating System (Betriebssystem), d.A.].“¹⁰

Webbrowser bilden die Plattformen der Repräsentation und der Interaktion mit den Inhalten des World Wide Web. Obwohl sie als „herkömmliche“ Computerprogramme intrinsisch mit den Elementen der oberflächlichen Architektur des Betriebssystems verbunden sind, etablieren deren Interfaces zur Navigation in der elektronischen Informationslandschaft ein völlig anderes Interaktionsparadigma.¹¹ Es geht in dieser Arbeit jedoch nicht um die Differenzen der Logik unterschiedlicher Interfaces, sondern um deren gemeinsame Fundamente. Der Webbrowser dient ebenso wie das grafische Interface des Betriebssystems dazu, die unsichtbare und unverständliche Welt der Nullen und Einsen in eine Sprache zu übersetzen, welcher der Benutzer versteht - denn jede Beziehung zwischen Mensch und Computer ist semantischer Natur. Vor der Entwicklung des grafischen User-Interfaces war die Interaktion mit dem Computer auf getippte, schriftliche Kommandos limitiert, was dessen Benutzung auf einen engen Personenkreis einschränkte. Erst durch eine Ikonografisierung von Speicherinhalten, Rechenprozessen und Befehlsstrukturen - kurz, durch die Entwicklung einer leicht erlernbaren und intuitiven Computerbedienung, konnte der Computer zu einem integralen Bestandteil der Massenkultur avancieren. Frank Hartmann beschreibt die grafische Benutzeroberfläche daher treffend als eine „in die Apparatur integrierte Ikonizität, die ihre Funktionen erst allgemein verfügbar macht.“¹²

Konstitutive Elemente der visuellen Organisation grafischer Interfaces wie z.B. Aqua (Mac OS X), Aero (Windows) oder KDE bzw. GNOME (Linux/Solaris) sind Fenster, Icons und Menüs, die mithilfe der Maus (bzw. bei portablen Geräten mit Touchpads etc.) angesteuert

¹⁰Manovich, Lev (2001): *The Language of New Media*, S. 64.

¹¹Für einen näheren Vergleich der Interaktionsparadigmen von Web-Browsern und Desktop-Systemen siehe Müller-Prove (2008): „Vom Persönlichen Computer zum Sozialen Medium“, In: Hellige, Hans-Dieter (Hg.): *Mensch-Computer-Interface: Zur Geschichte und Zukunft der Computerbedienung*, S.173-196.

¹²Hartmann (2006): S. 191.

und manipuliert werden können. Fenster ergeben sich aus der Notwendigkeit die begrenzte Bildschirmfläche möglichst effizient zu nutzen und können, ihrem physischen Gegenstück gleich, geöffnet und geschlossen werden. Sie sind gerahmte, zweidimensionale rechteckige Bildflächen, die den visuellen Arbeitsbereich einzelner Programme bzw. der bearbeiteten Dokumente repräsentieren. Wird mit mehreren Programmen gleichzeitig gearbeitet, werden die Fenster überlappend dargestellt und können einzeln skaliert und repositioniert werden.¹³ Mit Ausnahme von Dialogfenstern, die eine situationsspezifische Eingabe des Benutzers erwarten (z.B. „Wollen Sie das Programm beenden?“), besitzen sie stets einen Titelposten mit den Schaltflächen für die Befehle minimieren, maximieren und schließen. In den meisten Fällen verfügen Fenster auch über Symbolleisten (Toolbars), die einen schnellen Zugriff auf verfügbare Steuerungsbefehle bzw. Benutzeraktionen ermöglichen. Diese werden häufig in Form von Icons repräsentiert und können durch Anklicken aktiviert werden. Icons dienen neben der bildlichen Darstellung von Befehlen außerdem der Repräsentation von Programmen und Objekten (wie z.B. Ordnern, Laufwerken).¹⁴ Menüs bieten dem Benutzer eine hierarchische Auswahl an vordefinierten Befehlen, die über Mausclick aktiviert werden können und sind in einer horizontalen Menüleiste am oberen Bildschirm- oder Fensterrand arrangiert. Wird auf die Schaltfläche einer Menükategorie geklickt, öffnet sich eine Liste mit den ausführbaren Befehlen - daher auch der Name Drop-down-Menü. Andererseits bieten sogenannte Pop-Up-Menüs, die über eine bestimmte Maustaste (in der Regel durch einen Rechtsklick) aktiviert werden, dem Benutzer je nach Cursorposition kontextsensitive Optionen an. Die eben erläuterten Elemente konvergieren in der allumfassenden Desktop-Metapher, der virtuellen Arbeitsumgebung, an in der sich der Schein der Oberfläche verdichtet. Ursprünglich nämlich treffend als Benutzerillusion bezeichnet, gewährt sie dem Benutzer eine Vorstellung von der Funktionsweise des Computers. Die technischen Innereien der algorithmischen Maschine verschwinden hinter einem simulierten Schreibtisch, der die Möglichkeiten und die Poesie des Interaktionsraums markiert. Die symbolische „Domestizierung“ des Computers bietet dem Benutzer erfolgreich die Illusion, trotz Unkenntnis der technischen Funktionsweise über die Maschine verfügen zu können.

Grafische Benutzeroberflächen vom Format „Desktop-WIMP“¹⁵ (windows, icons, menus, pointing device) bilden die Basis der heutigen Mensch-Computer-Interaktion und veränderten

¹³Zur näheren Erläuterung der Genese der überlappenden Fenster siehe Kap. 6.5.2.

¹⁴Zur näheren Erklärung von Icons und deren Entstehung siehe Kap. 6.4. sowie Kap. 6.6.

¹⁵Matthias Müller-Prove vermutet, dass diese Abkürzung von Unix-Anhängern stammen könnte, die als Benutzer von Kommandozeilen-Interfaces mit dem englischen Begriff „wimp“ (Feigling, Schlappschwanz) abschätzig auf Mausbenutzer rekurrerten. Vgl.: Müller-Prove (2008): S. 179.

sich seit ihrer Entwicklung in den 70er nur marginal. Zwar erfolgten durch die steigende Komplexität der Anwendungsmöglichkeiten sowie durch einen erhöhten Fokus auf Benutzerfreundlichkeit gewisse Weiterentwicklungen, wie etwa hierarchische Ordnerstrukturen, Kontextmenüs, der Windows-Task-Bar bzw. der Apple-Dock sowie Grenzerweiterungen und Strukturierungen des virtuellen Arbeitsbereiches (z.B. Spaces bei Mac OS X), die jedoch allesamt das ursprüngliche Modell lediglich erweitern und nicht grundsätzlich in Frage stellen. Andries van Dam, Computerwissenschaftler und Verfechter neuer Interface-Konzepte namens „Post-WIMP“¹⁶, resümiert:

“What WIMP GUIs have made possible is a de facto standard for the application interface that, compared to command line interfaces, gives us (relative) ease of learning, ease of use, and ease of transfer of knowledge gained from using one application to another because of consistencies in look and feel. ‘No one reads manuals anymore’ because by and large they don’t have to.”¹⁷

Van Dam liefert mit seiner Feststellung bereits eine Antwort auf die Frage, weshalb die Computerbenutzung zu einem alltäglichen und automatisierten Gestus avancieren konnte. (Relativ) schnelles Erlernen der, im Gegensatz zur Kommandozeilensteuerung, einfacheren Benutzung durch eine Konsistenz im Systementwurf. Sind etwa die Anordnung und die Befehle der Menüpunkte in der Menüleiste anwendungsunabhängig gleichbleibend, so wird die Struktur für den Benutzer vorhersagbar. Unter DATEI, in der Regel der erste Menüpunkt, finden sich stets Befehle zum Laden, Speichern und Drucken. Auch die drei Knöpfe an den oberen Ecken der Fenster meinen immer minimieren, maximieren und schließen. Visuelle Konsistenz verbindet sich mit der Gewissheit, dass eine bestimmte Aktion stets denselben Effekt hat. Die Bedienung ist somit, im Gegensatz zu schriftlichen Kommandos, eng an das räumliche Gedächtnis in Zusammenspiel mit der Motorik des Benutzers geknüpft.

Die prägnanteste Interaktionskonvention von „Desktop-WIMP“-Systemen ist die Ansteuerung und Kontrolle grafisch repräsentierter Daten mittels Selektion durch Mausclicks (bzw. Herumtappen mit den Fingerspitzen auf Touchpads oder ähnlichen Mausalternativen an mobilen Geräten). Diese spezifische Verschränkung haptischer und visueller Aktivitäten lässt sich unter dem Konzept „direct manipulation“¹⁸ subsumieren. „User of this type of system have the feeling that they are operating upon the data directly, rather than via an agent: like fetching a

¹⁶Das Ziel der Post-WIMP Interface-Designer ist die Minimierung mechanischer Operationen und der kognitiven Distanz zwischen der Absicht des Benutzers und der technologischen Ausführung derselbigen. Die „Post-WIMP“ Interfaces „don’t use menus, forms, or toolbars, but rely on, for exemplae, gesture and speech recognition for operand and operation specifications.“, van Dam, Andries (1997): „Post-WIMP User Interfaces“, <http://portal.acm.org/citation.cfm?id=253708> (23.11.2010), S. 64.

¹⁷van Dam (1997): S. 65.

¹⁸Der Ausdruck wurde 1983 von Ben Shneiderman geprägt. Vgl.: Shneiderman, Ben (1983): „Direct Manipulation: A Step beyond Programming Languages“, <http://www.computer.org/portal/web/csdl/doi/10.1109/MC.1983.1654471> (20.11.2010)

book from a library shelf oneself rather than asking someone to do it.”¹⁹ Die schriftliche Formulierung abstrakter Kommandos wurde abgelöst von einem simulierten physischen Kontakt mit grafisch visualisierten Daten. Ein typisches Beispiel ist etwa das Löschen eines Dokumentes oder einer Datei durch Ziehen in den Papierkorb. Durch die unmittelbare Präsenz der vom Benutzer ausgeübten Aktionen kann dieser sofort abschätzen, ob seine Handlungsweise zielführend ist und bei Bedarf korrigieren. Durch die intuitive Erkundung des Systems stellt sich ein schneller Lerneffekt, nach einer gewissen Einübungszeit gar ein selbstverständlicher Umgang mit dem Computer ein.

Der Übergang der Interaktionskonvention vom „manual“ zur „direct manipulation“ bildete also ein wesentliches Fundament digitaler Medienkultur. Hinter dieser scheinbaren Direktheit verbirgt sich jedoch ein paradoxer Sachverhalt. Obwohl die taktile Unmittelbarkeit die Illusion erweckt, dass sich die Informationen in direkter Reichweite befinden, trennt die visuell organisierte Struktur des grafischen Interfaces den Benutzer eigentlich noch mehr von den Daten in ihrer informationstechnologischen Existenz. Nicht desto trotz erlaubte erst das Verschwinden des programmgesteuerten Gesichtes des Computers hinter der Magie der Interaktion mit Bildern die Computerbedienung auch für Computerunkundige, die sich nicht mit einer erst mühsam zu erlernenden Befehlssyntax herumplagen wollen. Entsprechend der Idee, dass Worte trennen und Bilder verbinden, hat das grafische User-Interface somit Millionen von Menschen an Computer gebunden und in das globale virtuelle Informations-, Wissens- und Kommunikationsnetzwerk integriert.

¹⁹Johnson, Jeff et.al (1989): „The Xerox ‚Star‘. A Retrospective“, S. 57.

4. Das Interface als Kategorie des Computerbildes

Die in der Definition des Interface angeklungene Ambivalenz zwischen der phänomenologischen Erscheinung des Bildschirms und dem codierten submedialen Hintergrund gibt Anlass, die grafische Benutzeroberfläche vom Standpunkt des apparativ erzeugten Bildes her zu fassen. Damit soll einerseits versucht werden das Wesen des Interface aus bildtheoretischer Perspektive zu beleuchten, andererseits, jene Grundkonzepte der Mensch-Computer-Interaktion zu verdeutlichen, die sich anhand der Eigenschaften des Computerbildes ergeben.

Alle am Computermonitor erscheinenden Ausgaben haben ihren Ursprung im digitalen Prinzip der elektronischen Datenspeicherung und -verarbeitung. Durch die Zerlegung der Daten in eine Reihe zweiwertiger Zustände, also in binäre Form, lassen sich diese durch Schaltzustände automatisch verarbeiten. Das Spiel zwischen Präsenz und Absenz gewisser elektronischer Zustände erlaubt also die technische Verarbeitung von Information. Diese werden durch Programme, also die konkrete Umsetzung von Algorithmen und den dazugehörigen Datenbereichen in eine Programmiersprache, geregelt. Im Gegensatz zu analogen Geräten, etwa einem Abakus, geht der digitale Code mit einem Verlust von Anschaulichkeit einher. Der Computer operiert also mit abstrakten Entitäten, die vom Menschen nur über vermittelnde Schichten, Interfaces also, interpretiert werden können, allen voran als visuelle Repräsentation der Informationen durch die Bildschirmausgabe. Das durch den Computer erzeugte Bild macht somit keine optische Spur, keine Aufzeichnung einer phänomenologischen Seite des Realen sichtbar, sondern es visualisiert stets logisch-mathematische Modelle, also alles, was sich formalisieren und durch den Computer mithilfe von Algorithmen prozessieren lässt. Das, was am Monitor erscheint ist immer Ausgabe eines laufenden Programmes. Die Bildfläche des Computerbildschirms entpuppt sich somit stets als doppelt existent. Sie spaltet sich in die sinnliche, oberflächliche Erscheinung am Bildschirm und in den verborgenen, algorithmischen Untergrund. Das ist bezeichnend für alle Lichterscheinungen am Display. Friedrich Kittler schreibt über das Computerbild: „Manchmal nämlich als grafische Oberfläche neomodischer Betriebssysteme, zeigt es sich minder, ein andermal, nämlich als Bild im emphatischen Wortsinne, etwas mehr.“²⁰ In dieser Feststellung liegt auch die Möglichkeit der Differenzierung zwischen grafischer Benutzeroberfläche und anderen Computerbildern. Die Benutzeroberfläche erhebt keinen ästhetischen Anspruch im eigentlichen Sinne, wie beispielsweise ein Film oder

²⁰Kittler, Friedrich (1998): „Computergraphik. Eine halbtechnische Einführung“, S. 48.

ein Bild innerhalb eines Programmfensters, sondern sie stellt dem Benutzer bildlich repräsentierte Möglichkeiten dar, über die Oberfläche Einfluss auf die hintergründigen, verborgenen Steuerungsprozesse des Computers auszuüben.

Mit Jörg Schirra, Vertreter der Computervisualistik, einem Versuch der Konvergenz von Informatik und Bildwissenschaft, lässt sich eine weitere Differenzierung des Computerbildes vornehmen. Aus der Perspektive der Informatik stellt sich vor allem die Frage nach spezifischen Transformationsalgorithmen, „nämlich ob die hauptsächlich betrachteten Operationen eine Instanz von ‚Bild‘ in etwas anderes umwandeln, oder umgekehrt.“²¹ Schirra nennt drei Hauptbereiche der Computervisualistik:

-Algorithmen von Bild zu Bild: Dabei geht es v.a. um die Beschäftigung mit Prozessen der Bildverarbeitung (z.B. Kantenfinder, Kontrastverstärkung).

-Algorithmen von Bild zu Nicht-Bild: Dieses Gebiet umfasst v.a. Ansätze aus der Künstlichen Intelligenz, dem Computer visuelle Wahrnehmung, i.e. die Interpretation und das Verständnis von bildlich repräsentierten Daten beizubringen (z.B. Software zur Mustererkennung bzw. Handschriftenerkennung).

-Algorithmen von Nicht-Bild zu Bild: Hierbei können zwei Fachbereiche unterschieden werden. Einerseits Computergrafik, bei der es zunehmend um die exakte Modellierung dreidimensionaler geometrischer Modelle geht (z.B. Computeranimationen). Andererseits die Informationsvisualisierung, „eine bildhafte Umsetzung beliebiger anderer Datentypen, die insbesondere nicht-visuelle Komponenten in einem Zustandsraum‘ enthalten, für die eine Konvention der visuellen Darstellung (Farbcodes, Icons) verwendet oder gefunden werden muss.“²²

Die grafische Benutzeroberfläche ist ein prototypisches Beispiel für eine Vielzahl von Algorithmen von Nicht-Bild zu Bild, ist aber auch die Basis für die Implementierung und Anwendung der anderen zwei Kategorien. Sie ist also ein vom Computer selbst hervorgebrachtes bildliches Konstrukt bzw. die Illusion einer Entität heterogener grafischer Elemente für die Benutzung desselben. Als Bildfläche erscheint sie am Monitor in ihrer Grunddisposition formal fixiert. Sie beruht auf Konventionen bzw. Richtlinien der Mensch-Computer-Interaktion, die die visuelle Organisation der Bildfläche bestimmen und ist somit inhärenter Bestandteil des Betriebssystems. Das Interface als Kategorie des Computerbildes muss primär von seiner Funktionalität her gefasst werden, denn es ist schließlich immer *User-Interface*. Der medialen Logik gehorchend, verschwindet es großteils hinter dem, was es zur Schau stellt, z.B. andere

²¹Schirra, Jörg (2005): „Computervisualistik“, S. 268.

²²Ebda, S. 273.

Computerbilder, die Kittlers Bezeichnung des Bildes im emphatischen Wortsinn mehr entsprechen.

„Als technisches Bild ist das Interface auf eine seltsame Weise abstrakt und scheinbar kulturindifferent. Es öffnet einen Raum modellhafter Bezüge, bei denen der Referenzrahmen des Realen zugunsten einer neuen Form von Funktionalität verlassen wird: Die ‚virtuellen‘ oder ‚simulierten‘ Knöpfe und Regler, die uns die grafische Benutzeroberfläche anbietet, sind freilich nicht echt, aber sie erfüllen ihren Zweck wie ihre Vorläufer an mechanischen Geräten.“²³

Die Benutzeroberfläche reiht sich somit nicht in die Tradition apparativ erzeugter Bilder ein, die ihrer reinen Sichtbarkeit genügen, wie etwa die Fotografie, der Film oder das Fernsehbild. Sie ist vielmehr eine Ikonografisierung von Rechenprozessen, Speicherinhalten und Befehlsstrukturen, daher nicht nur ein technisches sondern primär ein technikfunktionales Bild.

4.1. Interaktion mit Bildern bzw. interaktive Bilder

Im Gegensatz zur Benutzung analoger mechanischer Instrumente erfordert der Verlust einer direkten haptischen Erfahrungsmöglichkeit aufgrund der Immaterialität elektronischen Informationen ein hohes Maß an visueller Rückkoppelung.²⁴ Die Möglichkeit der Berechnung und Darstellung von Bildschirminhalten in Echtzeit und deren Fähigkeit auf Eingabewerte des Benutzers zu reagieren, ist eine fundamentale Voraussetzung der Interaktion mit dem Computer und somit konstitutiv für die grafische Benutzeroberfläche. Fenster, Icons und Menüs sowie diverse andere Steuerungselemente sind die dominanten Elemente am Computerbildschirm. Die Maus verweist auf die leibliche Dimension des Benutzers, der durch seine, im Kontext der grafischen Benutzeroberfläche primär auf die Aktivität der Hände reduzierten, Handlungen das System affiziert, welches im Gegenzug ein instantanes visuelles Feedback liefert. Dieser Rückkoppelungsprozess ist essentiell für die Mensch-Computer-Interaktion. Der Dialog mit dem Computer folgt ebenso wie eine zwischenmenschliche Kommunikation dem Schema „Input-Output-auf diesen Output reagierenden neuen Input etc.“ und findet seine Bedingung im und seinen Ausdruck durch das Interface. Michael Heim verweist auf dessen etymologische Wurzel, das Prosopon²⁵:

„The archaic Greeks spoke reverently of prosopon, or a face facing another face. Two opposite faces make up a mutual relationship. One face reacts to the other, and the other face reacts to the other's reaction, and the other reacts to that reaction, and so on ad infinitum. The relationship then lives on as a third thing or state of being.“²⁶

²³Hartmann (2006): S. 191.

²⁴Auch akustisches Feedback ist eine wichtige Informationsquelle für den Benutzer. Standardisierte Klänge lenken die Aufmerksamkeit des Benutzers auf bestimmte Ereignisse, z.B. als Warntöne bei Systemfehlern oder als Benachrichtigung von eingehender elektronischer Post.

²⁵πρόσωπον: pros (zu, hin, gegenüber) und opopa (Auge, sehen)

²⁶Heim (1993): S. 78.

Um eine reibungslose, zufriedenstellende Kommunikation zu gewährleisten muss das System, im optimalen Fall ohne Zeitverzögerung, auf den Input des Benutzers reagieren, sodass „zu jedem Moment einer Handlung das System eine Rückmeldung darüber gibt, bis zu welchem Grad die Handlung ausgeführt ist, und damit den weiteren Handlungsverlauf steuert.“²⁷

Wurden Informationen bei Film und Fotografie noch chemisch bzw. beim Video magnetisch gespeichert und somit fest ins Trägermaterial eingebunden, ist die digitale Codierung nicht an vorgegebene Zusammenhangsbedingungen geknüpft. Daher lassen sich digital gespeicherte Informationen direkt adressieren und in beliebige Formate transformieren. Mediale Repräsentation ist erst durch die digitale Kodierung nicht mehr an eine a priori gegebene Präsentation gebunden. Mit Peter Weibel lässt sich das „digitale“ Bild wie folgt charakterisieren:

„Alle Parameter der Information, die zu einem Bild gehören und es konstituieren, sind bei der digitalen Speicherung im Computer sofort und unmittelbar, jederzeit zugänglich und veränderbar. Instantane Variabilität aufgrund der digitalen Speicherung der Information ist also das einzigartige Merkmal der Computerbilder.“²⁸

Also erst durch die digitale Informationsspeicherung ergibt sich auch die Möglichkeit der Interaktion mit dem Computer über das Interface. Ein interaktives Bild ist also stets ein Bild, welches auf der Grundlage digitaler Informationsspeicherung bzw. -verarbeitung beruht. Doch ein interaktiv erzeugtes „digitales“ Bild muss nicht gleich ein interaktives Bild sein. So bietet etwa ein Bildbearbeitungsprogramm die Möglichkeit durch Interaktion mit der Software ein statisches Bild zu erzeugen, das an sich keine charakteristischen Eigenschaften eines interaktiven Bildes aufweist, denn diese zeichnen sich eben dadurch aus, dass sie auf einen äußeren Input reagieren - wie die grafische Benutzeroberfläche. Der heutige Computerbenutzer tritt nämlich meist durch Manipulation vorwiegend grafischer Erscheinungen à la Icons und Fenster mit dem Computer in Kontakt. Jede Computerbenutzung ist daher interaktiv, ein dynamischer Prozess und in diesem Sinne tatsächlich vergleichbar mit einem zwischenmenschlichen Dialog. Doch den Blick des Interfaces, der ja ein zentrales Moment eines jeden Gesichtes ist, gibt es nicht wirklich. Es spürt als oberflächlicher Ausdruck eines programmierten kontextsensitiven Systems lediglich die Aktionen des Benutzers auf. „Der Rechner [...] ist in einem Zustand stetiger Bereitschaft, Überwachung und Aktion.“²⁹ Ein Programmkonstrukt, der Event Loop, zu deutsch Ereignisschleife, misst Änderungen des Systemzustands, wie z.B. eine Mausbewegung, einen Tastenanschlag bzw. ob ein externes Gerät angeschlossen oder eine DVD eingeschoben wurde. Bei jedem Ereignis, sei es das Drücken einer Taste oder die

²⁷Dinkla, Söke (1997): *Pioniere interaktiver Kunst von 1970 bis heute*, S. 48.

²⁸Weibel, Peter (1995): „Die Welt der virtuellen Bilder. Zur Konstruktion kontextgesteuerter Ereigniswelten“, S. 46.

²⁹Coy, Wolfgang (2008): „Auf dem Weg zum ‚Finalen Interface‘“, S. 314.

Änderung der Cursor-Position, wird auf der Hardwareebene ein Interrupt ausgelöst, also eine kurze Unterbrechung der Arbeit des Prozessors. Registriert das System ein Ereignis, so wird automatisch die für das Ereignis zuständige Funktion aufgerufen und ausgeführt. Wenn z.B. eine Taste gedrückt wird, schickt der Tastatur-Controller einen „Interrupt-Request“ an den Prozessor, der seine aktuelle Tätigkeit kurz, d.h. im für den Benutzer nicht wahrnehmbaren Millisekundenbereich, unterbricht und auf den Tastatur-Treiber verweist, der dafür zuständig ist, dass der Buchstabe gemäß standardisierter Codierungsvorschriften am Bildschirm erscheint. Das Programmierschema „Eingabe-Verarbeitung-Ausgabe“ bleibt dem Benutzer verborgen, es äußert sich ausschließlich in der Veränderung der Oberfläche, im Feedback des Systems. Der Ausdruck des Antlitzes der Virtualität ist somit im Grunde eine Mediation eines präkonfigurierten Programmbefehls. Ohne Input passiert nichts - leeres, ausdrucksloses Starren.

Erst die interaktiven Eigenschaften digitaler gespeicherter Information lassen den Computerbildschirm zu einem Handlungsraum gerinnen. Als funktionale, operative Bildfläche im Rahmen der Mensch-Computer-Interaktion hebt er sich damit deutlich von anderen Oberflächen ab, die in ihrer Funktion rein repräsentativ und passiv, also ohne Einflussmöglichkeiten sind, wie beispielsweise die Kinoleinwand oder der Fernschirmschirm.³⁰ Insofern ist es gerechtfertigt, wenn Frieder Nake das Zeigegerät als das eigentliche Interface verstehen will. Wie das Computerbild existiert auch dieses doppelt, nämlich in seiner physikalischen Ausformung, vorwiegend als Maus, und semiotisch als Cursor am Bildschirm.

„In diesem physikalisch-kognitiven Doppelpack findet durch ständig miteinander verschränkte Hand- und Augenaktivität alles seinen konzentrierten Ausdruck, was den Computer (das laufende Programm) vom Buch, vom Fernseher, ja gar vom Telefon unterscheidet.“³¹

Es ist eben dieser technikfunktionale „haptische Blick“, der den Umgang mit Computern prägt und gleichsam den bildtheoretischen Status des Interfaces beantwortet - Computerbenutzung ist eben kein Betrachten *von* Bildern sondern ein Tun *mit* Bildern (siehe Kap.6.4.). Dadurch gerinnt die Bildfläche des Monitors zu einem Handlungsraum, der sich entlang der systematisch aufgespürten Eigenbewegung der Hand erstreckt. Denn der Computer wird heutzutage nicht durch getippte Befehle sondern primär durch Zeigen und Klicken mit der Maus handhabbar gemacht. ‚Point and click‘, the hallmark of WIMP GUIs, has become part of mo-

³⁰Dagegen könnte eingewendet werden, dass beim digitalen Fernsehen die Möglichkeit besteht, das Bild zu pausieren bzw. Inhalte unabhängig von der Sendezeit abzurufen. Oder, dass die künstlerischen Video-Arbeiten Nam June Paiks gezeigt haben, dass das Medium sehr wohl manipulierbar ist. Jedoch liegt in beiden Fällen eine äußerst eingeschränkte Interaktivität vor, die keineswegs „kommunikative“ Eigenschaften aufweist.

³¹Nake, Frieder (2008): „Zeigen, Zeichnen und Zeichen. Der verschwundene Lichtgriffel“, S. 140.

dem culture"³², so auch van Dam. Das Konzept der direkten Manipulation bürgerte somit ein Repertoire standardisierter technikfunktionaler Gesten ein, durch die sich die ikonografische Ordnung der virtuellen Dinge strukturieren und manipulieren lässt. Doch die durch die grafische Benutzeroberfläche geordnete Erscheinung der Informationen hat natürlich nichts mit den Milliarden an elektronischen Impulsen zu tun, die durch die Siliziumchips strömen und sich, je nach Auftrag, zu bestimmten Formen zusammenraffen. Was zählt ist einzig die Erscheinung. „Außer der Repräsentation gibt es nichts. Stellen Sie es sich als existentielles WY-SIWYG [What You See Is What You Get, d.A., siehe Kap. 6.5.] vor."³³

4.2. Bitmapping - Visualisierung digitaler Muster

Anfang der 70er begünstigte eine Hardware-Revolution das Aufkommen des PCs und neuer Formen visueller Darstellungsmöglichkeiten. Mit der Erfindung des Mikroprozessors und integrierter Schaltkreise wurde einerseits ein neues Level der Miniaturisierung erreicht, andererseits fielen die Preise für Rechen- und Speicherkapazitäten, was sich v.a. auf die Auflösung der Bildschirme, also die zur Verfügung stehende Dichte an visueller Information, auswirkte. Es war eben auch zu jener Zeit, als sich die Ideen um die Funktionalität des Computers änderten. Der Computer wurde zunehmend als persönliches Medium gedacht und dessen Möglichkeit der Simulation rückte mehr und mehr ins Visier der Forscher (siehe Kap.6.2.). Der Bildschirmanzeige kam dementsprechend eine erhöhte Aufmerksamkeit zu, denn schließlich hängt die Verwendung des Computers wesentlich davon ab, wie dieser sich seinem Benutzer repräsentiert und welche Möglichkeiten des Ausdrucks zur Verfügung stehen.

In der Computergrafik lassen sich zwei grundlegende Möglichkeiten der Beschreibung und Erzeugung des Bildschirminhalts unterscheiden, Vektorgrafik und Rastergrafik. Unter einer Vektorgrafik versteht man ein Computerbild, welches aus geometrischen Primitiven, z.B. Linien, Kreisen oder Polygonen besteht. Die Darstellung eines Kreises etwa wird beschrieben durch Lage des Mittelpunkts und den Durchmesser. Gespeichert werden nur die Funktionsparameter zur Definition der Objekte, was auch deren Farbe, Füllung etc. beinhaltet. Die ersten Verbände aus Computer und Bildschirm konnten aufgrund der geringen Speicherfähigkeit nur vektorbasierte Grafiken darstellen. Ein Linie entstand dabei aus der Bewegung des Elektronenstrahls, der die im Speicher abgelegten Koordinaten des Anfangs- und des Endpunktes verband und so eine leuchtende Spur auf die Phosphorbeschichtung des Monitors zeichnete.

³²van Dam (1997): S. 65.

³³Laurel (1991): S. 15.

Der Eindruck eines stillstehenden Bildes kam dadurch zustande, dass dieses 30 mal pro Sekunde aufgefrischt wurde.³⁴ Auch die symbolische Ausdruckskraft der Bildschirmterminals früherer Großcomputer war meist begrenzt auf alphanumerische Zeichen. Mit einem Zeichengenerator wurden diese nach dem ASCII-Code³⁵ durch die Bewegungen des Elektronenstrahls als Linien und Winkel dargestellt. Doch, so Claus Pias, „bildet die Vektorgrafik gegenüber heute üblichen Bildschirmen ein ganz anderes Dispositiv.“³⁶ Diese operieren nämlich ausschließlich auf den Grundlagen der Rastergrafik, eine Technik die sich durch die zur Verfügung stehenden Speicherkapazitäten Anfang der 70er Jahre durchzusetzen begann. Denn im Gegensatz zu Vektorgrafiken erfordert diese Darstellungstechnik die Speicherung eines jeden einzelnen Bildpunktes. Der Wert eines jeden Bildpunktes, eines Pixels (Picture Element), wird dabei als Teil einer zweidimensionalen Matrix im (Bild)Speicher abgelegt. Dieser, auch Framebuffer genannt, enthält sozusagen eine digitale Kopie des gesamten Monitorbildes. Die gespeicherten Bildpunkte werden dann von einem Display Prozessor gelesen und durch die Anregung der Phosphorbeschichtung³⁷ bzw. heute vorwiegend von LCD-Elementen in Lichtsignale umgewandelt. Anders ausgedrückt, die „bits“ im Speicher werden zu „maps“ am Bildschirm - daher lautet die englische Bezeichnung für dieses Prinzip „bitmapping“.³⁸

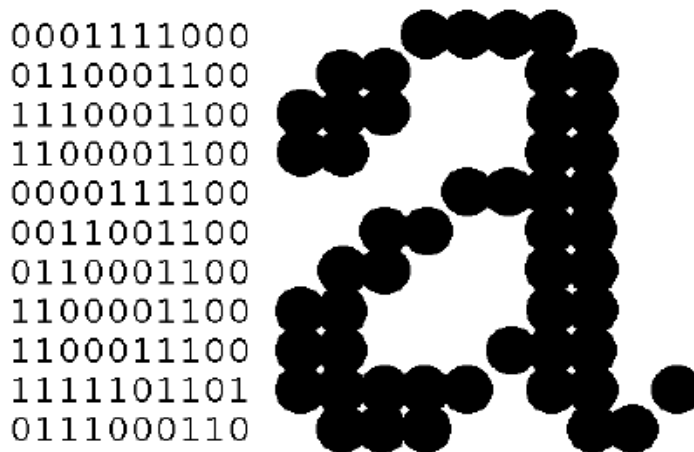


Abb.1: Darstellung des Buchstaben „a“ im Speicher und als Pixelmuster am Bildschirm

³⁴Heutige Computermonitore arbeiten, je nach Bildschirmgröße und Anwendungsbereich, von 60 Hertz aufwärts.

³⁵ASCII = American Standard Code for Information Interchange, 1963 als 7-Bit Zeichensatz entwickelt. Der ASCII-Zeichensatz als standardisierte Bit-Codierung spielt eine Schlüsselrolle in der Textdarstellung. Der Zeichengenerator dient der Transformation des Zeichencodes in eine für den Menschen wahrnehmbare Darstellung.

³⁶Pias, Claus (2001): „Punkt und Linie zum Raster“, S. 66.

³⁷Bei Röhrenmonitoren wurde der Elektronenstrahl durch eine Lochmaske, ein dünnes Metallgitter hinter dem Bildschirm, geschickt, welches die einzelnen Punkte definierte.

³⁸Vgl.: Hartmann (2006): S.196.

Die Breite und Höhe der gespeicherten Matrix entspricht dabei der Auflösung des Bildschirms. Dementsprechend sieht man sich bei einer Auflösung von 640 mal 480 theoretisch mit 307 200 einzelnen Bildpunkten konfrontiert. Bei einem Monochrombildschirm wird ein Bit pro Pixel visualisiert - ein Bildpunkt leuchtet daher oder eben nicht (Abb.1). Steht genügend Rechenleistung im Arbeitsspeicher zur Verfügung, lassen sich die Pixel farbig darstellen. Drei Byte - für Rot, Grün, Blau³⁹ - ergibt dementsprechend bei der Auflösung von 640 mal 480 256 Farben. Jedes Pixel ist somit durch einen Farbcode und, wie der Name Rastergrafik unschwer vermuten lässt, durch seine x und y-Position im virtuellen Koordinatensystem genau definiert. Es ist übrigens der Verdienst von Descartes analytischer Geometrie, dass Flächen- und Raumbeziehungen in Zahlenrelationen ausgedrückt werden können - Peter Weibel nennt es treffend „eine Digitalisierung bzw. Mathematisierung des Raums.“⁴⁰

Die Technik des „bitmapping“ wurde erstmals bei den im Xerox PARC entwickelten „Altos“ Anfang der 70er Jahre verwendet (siehe Kap. 6.5.). Diese markierten zugleich den Advent des multimedialen PCs und der grafischen Benutzeroberfläche heutiger Ausprägung. Rasterbildschirme, also die früher noch üblichen Braun'schen Röhren(monitore) mit Rasteransteuerung, brachten nämlich ein neues Repertoire der visuellen Organisation der von nun an durch Pixel definierten Bildschirmfläche mit sich.

„Die Notationen, die bei Vektorbildschirmen die Performance des Kathodenstrahls selbst zur Figur programmierten, wichen bei Rasterbildschirmen einer einzigen Bewegungsvorschrift, auf deren Bahn der Kathodenstrahl unterschiedliche Intensitäten durchlief. Mit ihm war somit die Fläche zu, Grundelement aller Darstellung geworden.“⁴¹

Die virtuelle Schreibtischfläche mit Fenstern, die durch einen ausgefüllten weißen Hintergrund Papier simulieren sollten (siehe Kap.6.4.), präzise editierbare Icons sowie Strukturen und Farben zur Betonung unterschiedlicher Bildschirminhalte wurden erst durch Rasterbildschirme ermöglicht. Die damit gewährleistete Manipulation von Schriftzeichen selbst zauberte weiters unterschiedliche Schriftsätze hervor, die sich kursiv oder fett darstellen ließen. Der menschlichen Fähigkeit, subtile visuelle Muster in einem größeren Informationszusammenhang zu erkennen, wurde dadurch stark zugearbeitet.

Neben den reichhaltigeren ikonischen Ausdrucksmöglichkeiten bedeutet die direkte Verbindung zwischen der Bildschirmanzeige und dem Speicherinhalt außerdem, dass eine Manipulation der Oberfläche den Speicherinhalt affizieren kann. Es ist eben jene direkte Relation zwi-

³⁹ Ein Farbton wird am Computermonitor durch die unterschiedliche Intensität von Rot, Grün, Blau definiert (additive Farbmischung). Unterschiedliche Farbmodelle dienen in der Computergrafik dazu, Farben als Zahlen darzustellen.

⁴⁰Weibel (1995): S. 36.

⁴¹Pias (2001): S. 68.

schen „bits“ und „maps“, also zwischen gespeicherter und prozessierbarer Information und deren Visualisierung, die dem Computermonitor seine Doppelsexistenz als Repräsentations- und Kontrollfläche verleiht und die direkte Manipulation von Bildschirminhalten als Computersteuerung ermöglicht. Die Bildfläche wird somit zur Benutzeroberfläche. Erst durch die Verbreitung von Rasterbildschirmen in Kombination mit der Maus konnte sich eine große Vielfalt an Software-Interfaces erschließen, was die Verbreitung von Anwenderprogrammen, in weiterer Folge die Akzeptanz des Computers in der Massenkultur enorm förderte,

„[d]enn durch bloßes Zeigen auf einen Punkt, eine Linie eine Fläche, ein Wort, ein Feld mit Schaltfunktion oder ein Symbol konnten die unterschiedlichsten Software-Prozesse angewählt und Aktionen ausgelöst werden.“⁴²

⁴²Hellige (2008): S. 48.

5. Historische Voraussetzungen für die Entwicklung des grafischen User-Interface

Computer entstanden einst aus der Notwendigkeit komplexe Rechenaufgaben, für die der Mensch viel zu lange brauchen würde, zu übernehmen. Noch vor 70 Jahren waren sie raumfüllende, kostspielige Rechenmaschinen, für deren alleinigen Betrieb viele Programmierer, Techniker, Mathematiker und Operateure zuständig waren. Anfangs wurden sie u.a. zur Decodierung feindlicher Funkcodes oder zur Berechnung von Flugbahnen und Abwurfhöhen von Atombomben verwendet. In den 60er Jahren gab es eine Handvoll dieser Geräte weltweit. Heute sind Computer populäre Konsumgüter und integraler Bestandteil unseres Alltags. Die Entwicklung des PCs mit grafischer Benutzeroberfläche in den 70ern baute auf Errungenschaften auf, die sich aus einer Neudefinition der Aufgabe, damit auch der Verwendung des Computers ergaben. Computer galten nicht mehr primär als stupide, vom Menschen abgesonderte Rechenmaschinen, sondern als Kommunikationspartner, Denkhilfen, ja sogar als Intelligenzverstärker. Dieser Bewusstseinswandel initiierte systematische Forschungen zu innovativen Interaktionskonzepten ab Beginn der 60er Jahre. Eine lineare Abstammungslinie der Mensch-Computer-Interaktion heutiger Ausprägung ist natürlich nicht auszumachen und würde auch jede Anerkennung der vielschichtigen Ebenen bzw. Akteure, die in einem technologischen Entwicklungsprozess beteiligt sind, untergraben. Das Ziel dieses Kapitels ist es vielmehr markante Meilensteine zu kontextualisieren, die sowohl in ideengeschichtlicher als auch in hard- und softwaretechnischer Hinsicht den Weg zur Entwicklung der grafischen Benutzeroberfläche ebneten.

5.1. Inspirationsquelle Vannevar Bush - „As we may think“

„Consider a future device for individual use, which is a sort of mechanized private file and library [...] A memex is a device in which an individual stores his books, records and communications, and which is mechanized so that it may be consulted with exceeding speed and flexibility. It is an enlarged intimate supplement to his memory.“⁴³

So formulierte der amerikanische Ingenieur, Manager und Politiker sowie Direktor des *Office of Scientific Research and Development*⁴⁴ Vannevar E. Bush 1945 im programmatischen Aufsatz „As we may think“ seine Vision einer Informationsarchivierungs- und -verarbeitungsmaschine, des Memory Extender- kurz Memex. In einer Zeit, als Computer noch raumfüllende

⁴³Bush, Vannevar (1945): „As we may think“, S. 45.

⁴⁴Während des zweiten Weltkrieges war das *Office of Scientific Research and Development* die zentrale Koordinationsstelle militärischer Forschungs- und Entwicklungsprogramme, u.a. auch des Manhattan-Projekts zur Entwicklung der Atombombe.

„Numbercruncher“ waren, betonte Bush deren Funktion als Speicher- und Arbeitseinheit für Wissen und Information. Außerdem verstand er den Memex, im Gegensatz zur damaligen Vorstellung des Computers als eine zentrale Datenverarbeitungsmaschine, als ein Gerät für die individuelle Benützung, was bereits die Idee des Personal Computing vorwegnahm. Der Hintergrund von Bushs Idee lag in der Beobachtung, dass Wissenschaftler angesichts der steigenden Anzahl von Publikationen, mangels struktureller und technischer Voraussetzungen einer Datenverwaltung und -ordnung, das vorhandene Wissen nicht effizient nutzen könnten. Mit dem Memex sollte eine revolutionäre Arbeitsumgebung geschaffen werden, um den „growing mountain of research“⁴⁵ effizient verwerten zu können. „There is a keyboard, and sets of buttons and levers. Otherwise it looks like an ordinary desk“ jedoch mit „slanting translucent screens, on which material can be projected for convenient reading.“⁴⁶ Die Dokumente sollten auf Mikrofilm gespeichert und laufend um neue Publikationen erweitert werden. Revolutionär war die zumindest angedachte Möglichkeit, dass der Wissenschaftler durch Querverbindungen zwischen Dokumenten eigene Denkpfade konstruieren könnte, indem die Informationen durch ein assoziatives Pfadsystem gespeichert werden. Schließlich funktioniere auch das menschliche Gehirn „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 combinations possible.“⁴⁷ „As we may think“ wird meist in Zusammenhang mit Hypertext zitiert und zeigt, wie aktuell Bushs Forderungen auch heute noch sind. Wann wird die bereits vor 65 Jahren angedachte Möglichkeit persönlicher Denkpfade im Internet standardmäßig implementiert werden? Bushs Vision verdeutlicht jedoch auch den Beginn eines entstehenden Bewusstseins, genuin menschliche Denkmechanismen auf Maschinen zu übertragen, die als Werkzeuge eines effizienteren Umgangs mit Informationen und somit als Unterstützer des menschlichen Intellekts dienen sollen. Friedewald konstatiert zu Recht, dass „As we may think“ das erste konzis formulierte Konzept einer interaktiven Mensch-Computer-Kommunikation war.⁴⁸ Der Memex konnte trotz einiger Anläufe nicht realisiert werden, und auch später wurden keine Versuche unternommen, Bushs Ideen weiterzuführen, was u.a. der Verdrängung der Analogtechnik durch die zunehmende Hinwendung zu Digitalrechnern ab 1946 geschuldet ist.⁴⁹ Nichts desto trotz waren seine Gedanken, v.a. über die Ver-

⁴⁵Bush (1945): S. 37.

⁴⁶Ebda, S. 45.

⁴⁷Ebda, S. 44.

⁴⁸Vgl.: Friedewald, Michael (1999): *Der Computer als Werkzeug und Medium. Die geistigen und technischen Wurzeln des Personal Computers*, S. 70f.

⁴⁹Vgl.: Friedewald (1999): S. 38.

wendung des Computers als informationsverarbeitende Technologie, um den Umgang mit Wissen auf eine neue Stufe zu heben, eine reichhaltige Inspirationsquelle für folgende Generationen von Computeringenieuren.

5.2. SAGE - der militärische Ursprung der Bedienkonsolen

1952 entwickelte IBM auf der Grundlage von Whirlwind, dem ersten Digitalcomputer zur Echtzeitinteraktion, die AN/FSQ-7 Rechner (Whirlwind II) als Basis für ein computergestütztes Luftverteidigungssystem - SAGE⁵⁰. Da der Computer kontinuierlich Daten - z.B. von Radarstationen - empfing, wurden hier bereits Magnetkernspeicher, Interrupt-Signale und Zwischenablagen von Daten verwendet, um einen interaktiven Computerbetrieb, im



Abb.2: SAGE - interaktive Bildschirmeingabe mit der Lightgun

Gegensatz zur Lochkartensteuerung, zu ermöglichen. Die damals und bis weit in die 70er Jahre übliche Eingabe von Programmen und Daten über Lochkarten, das sogenannte „Batch-Processing“, eignete sich nämlich nicht für „zeitkritische“ militärische Aufgaben, da alle benötigten Daten bereits vor der Abarbeitung eines Auftrag, in Lochkarten gestanz, vorliegen mussten. Die NORAD (North American Aerospace Defense Command) unterhielt von 1963 bis 1983 insgesamt 23 SAGE-Befehlszentren zum Aufspüren, Verfolgen und Abfangen von feindlichen Militärflugzeugen. Das System war allerdings bereits zum Zeitpunkt der Inbetriebnahme - strategisch gesehen - unbrauchbar, da Interkontinentalraketen mit geringer Vorwarnzeit längst die Bedrohung durch Bomber abgelöst hatten.⁵¹ Nichts desto trotz war SAGE wegwei-

⁵⁰SAGE = Semi-Automatic-Ground-Environment

⁵¹Vgl.: Friedewald (1999): S. 95.

send für die Entwicklung von Datennetzen⁵² und vor allem, was hier von Interesse ist, von Bedienkonsolen und grafischer Echtzeit-Interaktion mit dem Computer. Da die „User“ keine Computerexperten, sondern Soldaten waren, findet sich bei SAGE die erste komplexe Mensch-Maschine-Schnittstelle mit einer Bedienkonsole, die in den Grundzügen, wenn auch in weiterentwickelter Form, bis heute prototypisch für den Computergebrauch ist. Kathodenstrahlröhren dienten als Monitore, auf denen mit der Lightgun⁵³ zum ersten Mal Daten/Informationen in Echtzeit manipuliert werden konnten. Die Braun'sche Röhre wurde bereits seit Anfang der 30er bei Fernsehempfängern und ab den 40ern als Radarbildschirm verwendet. Bei SAGE diente der Monitor jedoch nicht nur dafür entfernte Ereignisse anzuzeigen, sondern zur aktiven Überwachung und zum Eingriff in Realvorgänge des durch Radartechnologie erfassten Luftraumes. Die Charactron-Röhre, eine Kathodenstrahlröhre mit einer in den Strahlengang eingesetzten Metallschablone mit 64 Symbolen (stencil), bildete als „erstes visuelles dynamisches interaktives Display“⁵⁴ das Kernstück der grafischen Luftraumdatenrepräsentation. Küstenlinien wurden aus einzelnen Strichen (Teile des Zeichensatzes) unterschiedlicher Neigungswinkel, zusammengesetzt. Wollte der „Operator“ ein repräsentiertes Flugobjekt auswählen um dem Computer mitzuteilen, dass er dessen Flugbahn als Graphen abbilden sollte bzw. um zusätzliche Informationen wie etwa Freund-/Feinderkennung, Höhe und Geschwindigkeit abzurufen, so markierte er es direkt am Bildschirm mit der Lightgun.⁵⁵ Diese wurde ursprünglich nicht für die militärische Freund-/Feindidentifizierung entwickelt, sondern schon 1948 um die fehleranfälligen Speichersysteme des Whirlwind in den Griff zu bekommen. Um dessen Speicherröhren zu testen wurde ein Programm entwickelt, welches die einzelnen Röhren auf einem Oszilloskopdisplay (ebenfalls eine Kathodenstrahlröhre) als Lichtpunkte repräsentierte. Das Programm tat nichts anderes, als jede Speicherzelle einzeln auszulesen und eine 1 zu schreiben - programmatisch hieß es „Waves of One“. Konnte keine 1 geschrieben werden, wurde das Programm gestoppt. Um in diesem Fall die betreffende Speicherzelle identifizieren zu können, wurde die Lightgun entwickelt. Norman Taylor schildert die Erfindung:

⁵²Für die Echtzeitübertragung von Daten (die analogen Radardaten wurden in digitale Signale konvertiert) über Telefonnetze wurden die ersten Modems und eine Reihe von Codierungs- und Datenkompressionsverfahren entwickelt. Vgl.: Friedewald (1999): S. 98.

⁵³Die Lightgun bestand aus einem Tubus mit eingebauter Linse und Fotozelle. Wurde das Gerät auf den Monitor gerichtet, so registrierte die Fotozelle immer dann einen Lichtimpuls, wenn ein Bildpunkt in ihrem Blickfeld neu geschrieben bzw. aufgefrischt wurde. Das an den Computer weitergeleitete elektronische Signal diente der Bestimmung der Bildschirmkoordinaten.

⁵⁴Dinkla, Söke (1997): *Pioniere Interaktiver Kunst von 1970 bis heute*, S. 51.

⁵⁵Die Benennung bezieht sich entweder auf die „electron gun“, die Fokussierungsvorrichtung der Kathodenstrahlröhre, oder - was wahrscheinlicher ist - auf das Aussehen und die Funktionalität: die der Gestalt einer Pistole ähnliche „Light Gun“ fungierte nämlich zum symbolischen Beschuss verdächtiger Flugobjekte. Damit wurde dem Computer signalisiert, welche Flugbahn er zu verfolgen und darzustellen habe. Vgl.: Hellige (1998): S. 34.

„We were asking how we can identify the address of that spot. So Bob Everett, our technical director, said ‚we can do that easily‘. All we need is a light gun to put over the spot that stops and we’ll get a readout as to which one it is. So he invented the light gun that afternoon and the next day we achieved man-machine interactive control of the display - I believe for the first time. This was late ‘48 or early ‘49.”⁵⁶

Erst im sogenannten „Situation Display“ von SAGE verschmolz die taktile Selektion mit militärischer Taktik, denn die grafische Repräsentation ermöglichte dem geübten Offizier schließlich „eine Lagebeurteilung *auf einen Blick*“⁵⁷. Bereits hier deutet sich die Überlegenheit bildlich dargestellter und direkt manipulierbarer Information an, die sich bei SAGE insbesondere durch die simultane Präsenz mehrerer Daten des repräsentierten Luftraumes auszeichnete.

Das SAGE-Projekt war nicht nur die Quelle von Ein- und Ausgabegeräten der Mensch-Maschine-Interaktion, sondern löste auch das Bewusstsein nach einem Bedarf an neuen Interaktionskonzepten und interaktiven Mensch-Computer-Interfaces aus. Für den Paradigmenwechsel, der sich Anfang der 60er Jahre abzuzeichnen begann war ein komplexes Zusammenspiel aus neuen Leitideen und institutionellen Rahmenbedingungen, nötig. Joseph Licklider war u.a. eine prägnante Schlüsselfigur für die konzeptionelle Initialzündung neuer Formen der Mensch-Computer-Interaktion.

5.3. Joseph Lickliders „Man-Computer-Symbiosis“ als impulsgebende Leitidee

Lange Zeit wurden die raumfüllenden und klimatisierten Rechnergiganten ausschließlich im sogenannten „Batch-Modus“ betrieben. Die Eingabe von Daten und Programmen erfolgte über Lochkarten, als Ausgabe der Ergebnisse dienten Drucker. Der Computer selbst und die Ein- und Ausgabegeräte waren in unterschiedlichen Räumen der Rechenzentren untergebracht. Dort wurden vorwiegend technische, naturwissenschaftliche bzw. mathematische Berechnungen sequentiell Karte für Karte prozessiert. Außerdem herrschte eine strikte Arbeitsteilung. Die anspruchsvolle Aufgabe der Programmierstellung, das monotone Anfertigen von Lochkarten (typische Frauenarbeit), die nicht weniger spannende Programm- und Dateneingabe sowie die, nur wenigen Privilegierten vorbehaltenen, Operatoraufgabe an der Rechenanlage. Einzig wer Zugang zu den abgeschotteten, brummenden und blinkenden Rechenautomaten hatte, konnte Programme ausprobieren und im Falle des Nicht-Funktionierens auch gleich

⁵⁶Taylor, Norman et. al. (1989): „Retrospectives: The early Years in Computer Graphics at MIT, Lincoln Lab and Harvard“, <http://portal.acm.org/citation.cfm?id=77277.77279> (12.11.2010), S. 20.

⁵⁷Friedewald (1999): S. 99.

„through a very detailed set of instructions“⁵⁸ debuggen⁵⁹. Von Interaktion konnte keine Rede sein, die vorprogrammierte Befehlsgewalt über den Computer lag in den Lochkarten mit ihrer binären Codierung Loch bzw. Nicht-Loch. Joseph Licklider beschreibt die damals vorherrschende „Benutzung“ des Computers sehr treffend:

„You formulate your problem today. Tomorrow you spend with a programmer. Next week the computer devotes 5 minutes to assembling your program and 47 seconds to calculating the answer to your problem. You get a sheet of paper 20 feet long, full of numbers that, instead of providing a final solution, only suggest a tactic that should be explored by simulation.“⁶⁰

Zumal es keine User im eigentlichen Sinne gab, kann hier auch noch nicht von einem User-Interface gesprochen werden. Der fehlende Direktzugang zu Rechnern sowie deren komplizierte Steuerung über monologische Befehlssequenzen, die ohnehin nur wenigen Experten zur Verfügung stand, führte vermehrt zur Forderung nach dialogischen Interfaces „to facilitate the transmission of general ideas as in a conversation, between a human and a computer, so that the maximum use of their respective capabilities can be made.“⁶¹ So der Computerwissenschaftler Douglas Ross (1929-2007). In ähnlicher Weise argumentierte Joseph Licklider (1915-1990), treibende Kraft hinter der Time-Sharing-Bewegung, des ARPANET⁶² und wegweisender Ideengeber für die Neugestaltung der Mensch-Maschine-Interaktion. Lickliders akademischer Hintergrund war die Psychologie, er arbeitete jedoch auch eng mit Ingenieuren im Bereich der Elektrotechnik zusammen. Nathan Rosenberg und W.E. Steinmüller prägten dafür den Begriff des „gatekeepers“⁶³, dem durch die Vermittlung der Zusammenarbeit unterschiedlicher Disziplinen eine entscheidende Rolle in der Verbreitung neuer Technologien zukommt. Sein interdisziplinärer Hintergrund erlaubte Licklider eine konzeptionelle Weitsicht zur Neuformulierung und Ausarbeitung von interaktiver Computerbenutzung, was klarerweise auch mit einer Änderung des Interfaces einherging. So formulierte Licklider etwa: „A new concept is achieved, not by creating a new schema ab initio, on a custom basis, but by adapting an old schema, or if necessary, arranging several refurbished schemata into a new, complex structure.“⁶⁴ Ein gewisses Maß an Eklektizismus und ein neues Arrangement bereits bestehender Errungenschaften unter veränderten Leitideen und technologischen Rahmenbedingungen, zie-

⁵⁸Carr, J.W. (1962): „Better Computers“, S. 158.

⁵⁹Debugging bezeichnet das Fehlersuchen in einem Programm.

⁶⁰Licklider, Joseph (1960): „Man-Computer Symbiosis“, S. 75.

⁶¹Ross, Douglas (1956): „Gestalt Programming: A new Concept in Automatic Programming“, <http://portal.acm.org/citation.cfm?id=1455410.1455414> (20.11.2010), S. 5.

⁶²Das 1969 realisierte ARPANET gilt als der Beginn des Internets. Es vernetzte die einst nur über lokale Netzwerke verbundenen Computer amerikanischer Forschungseinrichtungen, überregional.

⁶³Vgl.: Rosenberg, Nathan/Steinmüller, Edward W. (1982): „The economic implications of the VLSI revolution“, S. 189 ff.

⁶⁴Licklider, Joseph (1965): *Libraries of the Future*, S. 3.

hen sich durch die gesamte Genealogie des Interfaces. 1960 veröffentlichte Licklider seinen richtungsweisenden Artikel „Man-Computer-Symbiosis“, in dem er seine Visionen einer fruchtbaren Zusammenarbeit von Mensch und Computer durch innovative Dialogsysteme skizzierte, was schnell zu einem einflussreichen Leitbild avancierte. Licklider forderte eine Vorstellung der Mensch-Computer-Interaktion „in the same way that you think with a colleague whose competence supplements your own.“⁶⁵ Diese Vision entsprang aus Lickliders Selbstbeobachtung während seiner Arbeitszeit am MIT, einer „time-and-motion-study analysis of the mental work of a person engaged in a scientific or technical enterprise“.⁶⁶ 85% der Zeit, so musste er feststellen, verbringe er damit Informationen aus seinen Unterlagen zusammenzutragen, zu ordnen um sie dann zu grafischen Modellen aufbereiten zu können (die erst recht falsche Annahmen aufzeigten) - kurz: organisatorische Büroangelegenheiten, die der intellektuellen Arbeit in die Quere kommen. Daraus leitete er ab, dass es vorteilhaft wäre, routinemäßige Arbeiten, die leicht in Verarbeitungsvorschriften formuliert werden können, dem Computer zu überlassen. In den unterschiedlichen Modalitäten der Informationsverarbeitung lag für Licklider also das Potential des Computers zur Unterstützung geistiger Arbeit:

„[M]en are noisy, narrow-band devices, but their nervous systems have very many parallel and simultaneously active channels [...] computing machines are very fast and very accurate, but they are constrained to perform only one or a few elementary operations at a time“⁶⁷

Das Ziel der Symbiose zwischen Mensch und Computer, als ein „expected development in cooperative interaction between men and electronic computers“, sei einerseits „to let computers facilitate formulative thinking as they now facilitate the solution of formulated problems“ und weiters, eine Forderung, die vor dem Hintergrund des „Batch-Processing“ zu verstehen ist, „to enable men and computers to cooperate in making decisions and controlling complex situations without inflexible dependence on predetermined programs.“⁶⁸

Der Computer sollte nicht nur zur Berechnung von Problemen, sondern auch zu einer interaktiven Modellierung von Modellen und zur Simulation komplexer Probleme in Echtzeit beitragen. Dadurch ersetzte er die prophetische Vorstellung des Computers als Rechenmaschine durch die einer partnerschaftlichen Beziehung. Mensch und Maschine sollten zu einer symbiotischen Problemlösungseinheit verschmelzen, was mit der Hoffnung einherging „that in not too many years, human brains and computing machines will be coupled together very tightly, and that the resulting partnership will think as no human brain has ever thought.“⁶⁹ Die

⁶⁵Licklider (1960): S. 75.

⁶⁶Ebda, S. 76.

⁶⁷Ebda.

⁶⁸Ebda, S. 74.

⁶⁹Ebda.

Analogisierung der Mensch-Computer-Interaktion mit der Idee einer partnerschaftlichen Konversation, die durch einen synergetischen Effekt bis dahin unerreichte Höhenflüge des menschlichen Intellekts hervorbrächte, war eine zentrale Leitidee in der Anfangszeit interaktiver Systeme. Die naive Hoffnung an die Realisierbarkeit einer „speech communication between human operators and computing machines“⁷⁰ - Licklider spekulierte mit den nächsten fünf Jahren - zeigt, dass die Vorstellung einer Konversation mit dem Computer auch tatsächlich wörtlich genommen wurde. Schreiben und Lesen, später auch Sehen und Manipulieren von Bildern erwies sich schlussendlich gegenüber der Vision des gesprochenen Dialoges tatsächlich realisierbar. André Reifenrath argumentiert sogar, dass die Symbole und die künstlichen Welten des Bildschirms das Scheitern einer einst angestrebten intellegiblen Verständigung reflektieren würden.⁷¹

5.4. ARPA - militärische Forschungsgelder für Grundlagenentwicklung

Die *ARPA (Advanced Research Project Agency)*⁷² wurde Anfang 1958 unter der Regierung Eisenhower als Agentur des amerikanischen Verteidigungsministeriums gegründet. Als unmittelbare Antwort auf den Sputnik-Schock sollte sie alle landesweiten Forschungsbestrebungen koordinieren und eine technologische Lücke gegenüber der Sowjetunion verhindern. Dominierte Förderungsbereiche in den ersten Jahren waren Raketenabwehr, die Entwicklung von Verfahren zur Detektion von Atomwaffentests und Raumfahrt.⁷³ Gemäß der Umorientierung der militärischen Strategie unter Kennedy, wurde das Konzept der „massiven Vergeltung“ durch die neue Militärdoktrin der „flexiblen Antwort“ ersetzt, was eine vermehrte Förderung von interaktiven Computersystemen nach sich zog. Licklider wurde 1962 der erste Direktor des *Information Processing Techniques Office (IPTO)* der *ARPA*, eine einflussreiche Position, die ihm erlaubte, Forschungen zu seiner Vision der Mensch-Computer-Symbiose voranzutreiben. Primäres Ziel der *IPTO* war die Weiterentwicklung von militärischen Entscheidungsunterstützungs- bzw. Führungssystemen.⁷⁴ Die von Licklider formulierten Ziele und die Erfordernisse für militärische Computeranwendungen, nämlich kurze Antwortzeit und flexibler Umgang mit dem System, waren beinahe deckungsgleich. Bei näherer Betrachtung der Organisation der *IPTO* lässt sich außerdem feststellen, dass zwar, vor allem seitens des ameri-

⁷⁰Ebda, S. 80.

⁷¹Vgl.: Reifenrath, André (2000): „Die Geschichte der Simulation“, S. 112.

⁷²Heute unter dem Namen DARPA (Defense Advanced Research Project Agency) bekannt.

⁷³Auch die NASA wurde vor demselben Hintergrund gegründet.

⁷⁴Diese Art von militärischen Computersystemen werden auch als „Command and Control Systems“ bezeichnet.

kanischen Kongresses, die Wichtigkeit zur Erforschung interaktiver Mensch-Computer-Systeme für konkrete militärische Zwecke herausgestellt wurde, während die Wissenschaftler jedoch alle Freiheiten hatten, der Entwicklung ihrer Visionen nachzugehen.⁷⁵ Grundlagenforschungen und militärische Verwendung standen somit in wechselseitiger Beeinflussung und Stärkung. Die Entwicklung des Time-Sharing als technische Lösung zur Mensch-Computer-Interaktion in „Echtzeit“, interaktive Computergrafik, wie Ivan Sutherlands Sketchpad sowie anfangs auch Douglas Engelbarts Projekt - alles innovative Beiträge zur Entwicklung neuer Interaktions- und Interfacekonzepte - wurden mit militärischen Forschungsgeldern der ARPA realisiert.

5.5. Time-Sharing

Das Aufkommen von Time-Sharing, die Verteilung der damals noch äußerst kostbaren Rechenzeit unter mehreren Benutzern, zu Beginn der 60er markierte eine große Wende in der Computergeschichte. Das Time-Sharing-Prinzip revolutionierte die Qualität und den Modus der Interaktion mit dem Computer, denn anstatt binären Code zur sequentiellen Abarbeitung ohne Eingriffsmöglichkeit über Lochkarten einzugeben, wurde eine symbolische Ebene geschaffen, über die mit dem Computer in Echtzeit interagiert werden bzw. sein Rechenverhalten durch Eingabe von Kommandos direkt beeinflusst werden konnte. Jeder Benutzer arbeitete an einem eigenen Terminal, das an den Zentralrechner angeschlossen war. Durch die unmittelbare Repräsentation der Daten und die (relativ) kurzen Antwortzeit des Computers bei der Programmausführung, entstand die Illusion einer persönlichen und kontinuierlichen Interaktion mit dem Computer. Vor allem die Erstellung und das Debuggen von Programmen wurden dadurch erheblich erleichtert.

„The ability to program on-line frees the programmer from having to concern himself with all the formalities of punched card accounting. With experience and facility, he programs online directly from his thoughts or, for more difficult problems, directly from a flow diagram, circumventing such timeconsuming tasks as program-coding-sheet preparation, key punching, card sorting, editing, and prestoring. The time saved by the programmer can be applied to other coding tasks or to quality review of his current code.“⁷⁶

Durch fortschrittliche Dialogsprachen änderte sich auch bald die Grundhaltung der „Benutzer“ zu Computern, was u.a. dazu führte, dass sich viele auf die Fähigkeit der Beherrschung des Computers spezialisierten und das Interesse zur Codierung von Programmen

⁷⁵Vgl.: Friedewald (1999): S. 125-127.

⁷⁶Schwartz et.al.(1964): „A General-Purpose Time-Sharing System“, <http://portal.acm.org/citation.cfm?id=1464122.1464163> (12.11.2010), S. 406.

selbst abnahm.⁷⁷ Jörg Pflüger vermutet außerdem, dass die Unmittelbarkeit der Kommandostruktur, die beim „Batch-Processing“ schließlich nicht gegeben war, häufig mit Führung und Kontrolle identifiziert wurde.⁷⁸ Jedoch war die Interaktion, so innovativ sie damals erlebt wurde, beschränkt auf die Eingabe von Kommandos und Daten - somit auf eine Art Literatur, die nicht für den Benutzer selber Sinn machte (etwa als Textverarbeitung), sondern nur für den Computer. Technische und wissenschaftliche Berechnungen waren immer noch dominierende Anwendungen. Außerdem waren die Systeme aufgrund der fehlenden Möglichkeit symbolische bzw. bildliche Informationen schnell und intuitiv zu manipulieren, was schließlich ein wesentliches Merkmal geistiger Arbeit ist, noch weit davon entfernt der Vision gerecht zu werden. Daher forderte Licklider 1962:

„[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 spend a large part of its time maintaining the displays. The entire device should be inexpensive enough for incorporation into a remote control.”⁷⁹

Die Interaktion zwischen dem Computerwissenschaftler Fernando J. Corbato⁸⁰ und dem Mainframecomputer am MIT 1963 soll zur Veranschaulichung einer typischen „Konversation“ dienen.⁸¹ Das folgende Beispiel ist zwar ein ziemlich einfaches Programm, doch so Corbato „it’s illustrative of man-machine-interaction. I think it’s very clear that this is the technique that would work, let’s say in airline reservation or in a small business asking about a payroll problem or something in this sort.“⁸²



Abb.3: Fernschreiber-Terminal am MIT Computation Center 1963; Bildschirmterminals wurden erst später entwickelt.

⁷⁷Vgl.: Friedewald (1999): S. 135.

⁷⁸Vgl.: Pflüger, Jörg (2008): „Interaktion im Kontext“, S. 332.

⁷⁹Licklider, Joseph /Clark, Welden E. (1962): „On-Line Man-Computer Communication“, <http://portal.acm.org/citation.cfm?id=1460847> (20.11.2010), S. 121.

⁸⁰Corbato leitete ab 1961, noch bevor Time-Sharing durch Licklider massiv propagiert wurde, am MIT eines der ersten Time-Sharing-Systeme (CTSS).

⁸¹Corbato, Fernando J. (1963): „1963 Timesharing: A Solution to Computer Bottlenecks“, Interview von John Fitch mit Fernando J. Corbato am MIT Computation Center am 9.05.1963; online-Video, <http://www.youtube.com/watch?v=Q07PhW5sCEk> (20.11.2010) Das Beispiel ist eine Abschrift der im Video gezeigten Interaktion.

⁸²Ebda, 20’ 40’’ff. (Abschrift der Autorin)

Start.

Aktivierung des Programms durch Knopfdruck am Terminal; die Koordination einzelner Benutzeranfragen wird von einem Supervisor-Programm übernommen. Der Computer reagiert ohne merkbare Zeitverzögerung und schreibt:

WAIT, HELLO. I DO SQUARE ROOTS AND COMPUTE HYPOTENUSES. PLEASE TYPE EITHER ROOT OR TRIANGLE.

root

PLEASE TYPE A NUMBER UP TO 6 DIGITS, WITH DECIMAL POINT.

49

THE SQUARE ROOT IS 7.00 ... DO YOU WISH TO CONTINUE

sure

PLEASE TYPE YES OR NO.

Das Programm erwartet zwar eine präzise Eingabe, toleriert jedoch fast alles; durch einen Trial-Error-Prozess lernten die meisten Benutzer mit dem System umzugehen; schlimmstenfalls spuckte der Computer je nach Programmierung ein „Eh?“ oder ähnliches aus.⁸³

yes

PLEASE TYPE EITHER ROOT OR TRIANGLE

triangle

GIVE ME TWO SIDES AND I WILL GIVE YOU THE HYPOTENUSE. TYPE UP TO 6 DIGITS, WITH DECIMAL POINT. THE FIRST SIDE IS ...

Das Programm wartet auf eine Eingabe; während des Denkprozesses kann der Prozessor die Berechnungen anderer Benutzer ausführen.

1.0

AND THE OTHER SIDE

12

THE HYPOTENUSE IS 12.04....DO YOU WISH TO CONTINUE

no

EXIT CALLED. PM MAY BE TAKEN. (Ausdruck des Dialogs)

Bei gravierenderen Problemen als in diesem einfachen Beispiel bzw. der Entwicklung von Programmen diente die Ausgabe des Computers als Anschlussbedingung für den nächsten Schritt. In den Pausen zwischen Ausgabe und erneuter Eingabe, erfolgte die innerliche Gedan-

⁸³Vgl.: Pflüger (2008): S. 329.

kenarbeit des Benutzers und der Computer prozessierte unterdessen die Anfragen anderer Benutzer. Räumliche Orientierung erschien zu diesem Zeitpunkt noch nicht nötig, denn es wurde stets nur an einem Problem gearbeitet. Kooperative serielle Problemlösung durch einen alternierenden getippten „Dialog“ mit einem Computer, der die Probleme mehrerer Benutzer durch geschickte Koordination der Rechenzeit gleichzeitig abfertigte - das war also der anfängliche Status Quo der visionierten „Mensch-Computer-Symbiose“. Die Einführung und Weiterentwicklung von Time-Sharing-Systemen, die bis zum Advent des Personal Computing in den 70ern den Computerbetrieb dominierten, sowie die Entwicklung befehlsorientierter Dialogsprachen gaben der interaktiven Computerbenutzung einen gewaltigen Aufschwung.

5.6. Ivan Sutherlands Sketchpad - grafische Revolution der Mensch-Computer-Interaktion

Ebenfalls am MIT wurde durch Ivan Sutherlands Pioniersystem Sketchpad (1962/63) ein gewichtiger Beitrag zur grafischen Mensch-Computer-Interaktion und wegweisende Konzepte für die Entwicklung der grafischen Benutzeroberfläche gelegt.

„The Sketchpad system makes it possible for a man and a computer to converse rapidly through the medium of line drawings. Heretofore, most interaction between men and computers has been slowed down by the need to reduce all communication to written statements that can be typed.“⁸⁴

Mit diesen Worten beginnt Ivan Sutherlands (*1938), von Claude E. Shannon betreute, Dissertation „Sketchpad: A Man-Machine Graphical Communication System“, in deren Rahmen er das erste interaktive Grafikprogramm entwickelte. Der Übergang des Computers von einer numerischen zur geometrischen bzw. bildlichen Maschine „by eliminating typed statements“ durch eine „picture language perfectly natural to the human“ bezeichnet eine einschneidende Wende der Mensch-Computer-Interaktion. Sutherland schreibt sogar von Zeichnungen als „a novel communication medium for a computer.“⁸⁵ Das Einsatzspektrum von Sketchpad reichte von elektrischen, mechanischen, wissenschaftlichen, mathematischen bis zu „animated drawings“⁸⁶, die jedoch im Vergleich zu den ingenieurwissenschaftlichen Zwecken eine zu vernachlässigende Rolle einnahmen. Die vektorbasierten Zeichnungen konnten direkt am Bildschirm des, stets noch raumfüllenden TX-2 (damals einer der fortschrittlichsten und leistungsfähigsten Computer) mithilfe eines Lichtgriffels, dem verbesserten Nachfolger der Lightgun, erstellt und manipuliert werden. Insofern kann das Sketchpad-System nicht nur als Beginn in-

⁸⁴Sutherland, Ivan (1963): „Sketchpad: A Man-Machine Graphical Communication System“, elektronische Neuauflage 2003, <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.10.4290> (18.11.2010), S. 17.

⁸⁵Ebda, S. 9; S. 17; S. 9.

⁸⁶Ebda, S. 9.

teraktiver Computergrafik, sondern auch des Prinzips der „direct manipulation“ sowie als erstes grafisches User-Interface angesehen werden.⁸⁷ Am Bildschirm erschien jedoch noch keine Benutzeroberfläche mit Menüs oder Scrollbars zum Steuern der Grafiken. Kommandos an das System erfolgten ausschließlich über die rund 40 „command buttons“, die Operationen wie



Abb.4: Ivan Sutherland an der Konsole des TX-2 am MIT: Die Bedienung von Sketchpad erfolgte mit einem Lichtgriffel und Kommandoknöpfen (unter der linken Hand). Die vier schwarzen Knöpfe unterhalb des Bildschirms regelten Größe und Position der „Sketches“.

DRAW, MOVE, DELETE etc. ausführten bzw. Bedingungen auf bereits gezeichnete Objekte anwendeten, z.B. „to make lines vertical, horizontal, parallel, or perpendicular.“⁸⁸ Außerdem war es möglich Zeichnungen zu skalieren bzw. über den sichtbaren Bereich des Bildschirms hinaus zu verschieben. Sketchpad war damit das erste System, so Alan Kay, „to have clipping and zooming windows- one ‚sketched‘ on a virtual sheet about 1/3mile square.“⁸⁹ Sutherland erklärt den Vorgang des Zeichnens eines Striches:

„If we point the light pen at the display system and press a button called “draw”, the computer will construct a straight line segment which stretches like a rubber band from the initial to the present location of the pen [...] A sudden flick of the pen terminates drawing.“⁹⁰

Anstelle eines „sudden flick“, also einer ruckartigen Wischbewegung mit dem Lichtgriffel, konnte auch der STOP-Button gedrückt werden, um dem Computer den Endpunkt der Linie zu signalisieren. Wollte man z.B., dass die Linie horizontal oder vertikal ausgerichtet wird, so

⁸⁷Vgl. ebda: Alan Blackwell und Kerry Rodden im Vorwort der elektronischen Neuauflage.

⁸⁸Ebda, S. 25.

⁸⁹Kay, Alan (1993a): „The Early History of Smalltalk“, <http://portal.acm.org/citation.cfm?id=155364> (07.11.2010), S. 71.

⁹⁰Sutherland (1963): S.18.

übernahm dies der Horv-Knopf. Um einen Kreis zu zeichnen,

„we place the light pen where the center is to be and press the button ‚circle center‘, leaving behind a center point. Now, choosing a point on the circle (which fixes the radius) we press the button ‚draw‘ again, this time getting a circle arc, whose angular length only is controlled by the light pen position.“⁹¹

Der Benutzer zeigte, der Computer zeichnete. Die brillianteste Eigenschaft von Sketchpad war also die Softwarestruktur, die gemeinhin als Vorläufer objekt-orientierte Programmierung angesehen wird und erheblichen Einfluss auf Alan Kays „Smalltalk“ - ein wesentliches Fundament der grafischen Benutzeroberfläche - hatte. Jede Zeichnung in Sketchpad wurde als Objekt, also als Instanz einer Klasse bzw. einer „Urzeichnung“ gehandhabt. In der Klasse Kreis war beispielsweise festgelegt, dass alle Punkte des Objektes den gleichen Abstand zum Mittelpunkt haben müssen. Außerdem stellte die Bibliothek bereits viele Befehle wie drehen, verzerren, skalieren, gruppieren etc. bereit. Die grafischen Symbole am Bildschirm repräsentierten somit Programmteile, die individuell selektiert und manipuliert werden konnten. Dies führte u.a. dazu, dass „relatively new users with no programming knowledge could produce simple drawings with the system if a skilled user (myself) prepared the building blocks necessary.“⁹² So berichtet Sutherland etwa über die Entwicklung eines vektorbasierten Alphabets durch eine Sekretärin.⁹³ Der tatsächliche Benutzerkreis von Sketchpad war jedoch äußerst klein und nur den Wenigen vorbehalten, die Zugang zum TX-2 hatten. Umso größer jedoch war Sutherlands Verdienst der tatsächlichen Realisierung eines neuartigen Interaktionskonzeptes und des Aufzeigens bis dahin unbekannter Benutzungsmöglichkeiten des Computers.

Zur selben Zeit, an der anderen Küste von Nordamerika, nämlich am SRI (Stanford Research Institute), formulierte Douglas Engelbart (*1925) - laut Kay „a prophet of Biblical dimensions“⁹⁴ - sein großangelegtes Ziel: „Augmenting Human Intellect“⁹⁵. Die darauf folgenden Forschungen zogen, neben dem wohl beständigsten Relikt - der Maus - und dem von seinem Team des ARC (Augmenting Research Center) realisierten OnLineSystem⁹⁶ - das erste elaborierte (Hyper-)Textverarbeitungssystem - grundlegende Innovationen in der Computernutzung und gewichtige Ideen zur Ausgestaltung des Interfaces nach sich.

⁹¹Ebda.

⁹²Sutherland (1963): S. 35.

⁹³Ebda.

⁹⁴Kay (1993a): S. 72.

⁹⁵Vgl.: Engelbart, Douglas (1962): „Augmenting Human Intellect: A conceptual Framework“, <http://www.doungengelbart.org/pubs/augment-3906.html> (10.11.2010)

⁹⁶Online bezieht sich in diesem Fall auf das direkte, interaktive Arbeiten mit dem Computersystem über den Monitor als Ein- und Ausgabegerät im Gegensatz zum (damals üblichen) Offline/Batch-Betrieb; die Bedeutungsver-schiebung von „online“ als Verbindungsstatus mit einem Netzwerk fand erst eine Dekade später statt.

5.7. Douglas Engelbart - „Augmenting Human Intellect“

„Digital technology could help make this a better world. But we've also got to change our way of thinking [...] Being a real dreamer is hard work.“⁹⁷

Das World Wide Web bietet einen reichlichen Fundus an Artikeln von Engelbart bzw. Interviews mit ihm, die einen interessanten Einblick in die visionären Beweggründe jenes Mannes bieten, der nicht allzu selten nur als Erfinder der Maus genannt wird, obwohl er ja eigentlich angetreten war, „to bring significant improvement to the real-life problem-solving effectiveness of individuals.“ Ein Ziel von mindestens der gleichen sozialen Signifikanz wie „research toward harnessing thermonuclear power, exploring outer space, or conquering cancer.“⁹⁸ Engelbart war während des 2. Weltkriegs als Radartechniker im Südpazifik stationiert. Nach seiner Rückkehr setzte er sein begonnenes Studium der Elektrotechnik fort, beendete es 1948 und war daraufhin bei einer Vorgängerorganisation der NASA als Elektroingenieur tätig. Alles lief soweit gut, er war verlobt, hatte einen sicheren Job. Es war allerdings auch die Zeit des Koreakrieges und der nuklearen Bedrohungen. Engelbart beschloss, neue Herausforderungen zu suchen und etwas „Sinnvolles“ zu leisten. Er erinnerte sich u.a. an Bushs Artikel „As we may think“ (siehe Kap. 5.1.), den er noch während seiner Stationierung in Lahiti in einer Bibliothek des Roten Kreuzes gelesen hatte und resümiert:

„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.“⁹⁹

Als einstiger Radartechniker wusste, der an Computerwissenschaft und Programmierung ziemlich uninteressierte, Engelbart um die Möglichkeit von Kathodenstrahlröhren „to shape symbols from any kind of information“ und dass „the computer could manipulate the symbolic logic, and really help you in the kind of reasoning that is formal enough to employ that kind of symbolism.“¹⁰⁰ Um seinem Ziel näher zu kommen, erwarb er in Berkeley seinen Dokortitel, musste jedoch auch feststellen, dass „[t]he kind of things I was thinking about didn't turn them on, because the pressure behind building computers in those days was to support

⁹⁷Engelbart, Douglas (1995b): „Dreaming of the Future“, <http://web.archive.org/web/20080111155845/http://byte.com/art/9509/sec15/art1.html> (23.11.2010)

⁹⁸Engelbart, Douglas (1961): „Program on Human Effectiveness“, <http://sloan.stanford.edu/mousesite/Archive/Post68/PrHumanEffectiveness.html#1>. (23.11.2010)

⁹⁹Engelbart, Douglas (1995a): „Toward Augmenting Human Intellect and Boosting our Collective IQ“, <http://www.doungelbart.org/pubs/augment-133150.html> (23.06.2010), S. 3.

¹⁰⁰Engelbart, Douglas (1986/87): Interview 1, Tape 1, Side B. <http://www-sul.stanford.edu/depts/hasrg/histsci/ss-voval/engelbart/engfmst1-ntb.html> (23.11.2010)

numeric calculations.”¹⁰¹ Deshalb wechselte er 1957 ans SRI (Stanford Research Institut), zumal er das Gefühl hatte, dies sei der richtiger Ort um unabhängig von kommerziellen Zwängen und zeitraubenden Lehraufträgen seine Forschungen zu verfolgen. Engelbart kanalisierte seine Gedanken im kurzen Artikel „Program on Human Effectiveness” (1961) und in „Augmenting Human Intellect: A conceptual Framework”(1962), einem 130 seitigen Abschlussbericht seiner Forschungen zur Projektförderung. Schon die programmatischen Titel reflektieren Engelbarts Ziel, nämlich „an integrated man-machine working relationship, where close, continuous interaction with a computer avails the human of radically changed information-handling and -portrayal skills.”¹⁰² Ähnlich wie bereits Bush und Licklider erhoffte sich auch Engelbart durch die Koppelung von Mensch und Computer eine Effektivierung intellektueller Arbeit und bessere Ansätze zur Problemlösung. Doch Engelbart setzte völlig neue Akzente, zumal er den Kern menschlichen Denkens in „the purposeful manipulation of concepts” sieht, wobei sich ein Großteil der intellektuellen Kapazitäten des Menschen durch die Fähigkeiten und Mittel zur „external manipulation of symbols”¹⁰³ organisieren. Entschieden formulierte er die Vorstellung eines neuartigen intellektuellen Reichtums durch die symbolmanipulierende Maschine die einen mehrdimensionalen Gestaltungs- und Visualisierungsraum symbolischer Repräsentation eröffne, wobei er jedoch, im Vergleich zu den Entwicklern der grafischen Benutzeroberfläche, primär die sprachliche bzw. textliche Repräsentation von Konzepten fokussierte. So zitiert er z.B. eine These der ethnolinguistischen und sprachphilosophischen Forschungen von Benjamin Lee Whorf, nach der die Weltsicht einer Kultur durch die Struktur ihrer Sprache limitiert sei.¹⁰⁴ Engelbart betont außerdem: „The other important part of our ‚language‘ is the way in which concepts are represented--the symbols and *symbol structures*.”¹⁰⁵ Engelbarts Schlussfolgerung, die er als „Neo-Whorfian-Hypothesis” formuliert, ist, dass die jeweils verfügbaren Medien zur Kontrolle und Manipulation von Symbolen in direktem Verhältnis mit der Sprache einer Kultur (somit auch deren Weltsicht) sowie mit den intellektuellen Fähigkeiten des Formulierens und Generierens von Ideen stehen. Mit dem digitalen Computer, der das Potential völlig neuer Repräsentations- und Manipulationsmodalitäten bereitstellen könnte, sieht Engelbart die nächste Stufe menschlicher Intelligenz anbrechen:

„In this stage, symbols with which the human represents the concepts he is manipulating can be arranged before his eyes, moved, stored, recalled, operated upon according to extremely complex ru-

¹⁰¹Ebda.

¹⁰²Engelbart (1961)

¹⁰³Ebda.

¹⁰⁴Vgl.: Engelbart (1962): 2c4f.

¹⁰⁵Ebda, 2c5c4.

les--all in very rapid response to a minimum amount of information supplied by the human, by means of special cooperative technological devices.”¹⁰⁶

Gegenüber der Vorstellung einer symbiotischen Konversation, eines Dialogs mit dem Computer über seriell abgearbeitete Probleme forderte Engelbart von Anfang an dezidiert eine kreative Freiheit in der Verfertigung der Gedanken mittels externer dynamischer Symbolmanipulation - eine dynamische Wissensumgebung. Der Computer gerinnt in seinem Systementwurf zu einem „clerk“¹⁰⁷, einem Sekretär, der dem Benutzer, den Engelbart wenig überraschend als „knowledge worker“¹⁰⁸ visioniert, einen effektiveren Umgang mit komplexen Informationsstrukturen bieten soll. Ähnlich wie Licklider erkannte er im synergetischen Zusammenspiel der Möglichkeiten des digitalen Computers und den bereits existierenden kognitiven Fähigkeiten und Möglichkeiten des Menschen, „all of which appear to be but extensions of means developed and used in the past to help man apply his native sensory, mental, and motor capabilities“¹⁰⁹, ein Intelligenzverstärkendes Potential. Das medien- bzw. kulturanthropologische Argument, Geräte und Techniken als Erweiterung limitierter sensomotorischer und kognitiver Fähigkeiten des Menschen zu begreifen, sollte Marshall McLuhan zwei Jahre später in *Understanding Media*, das bezeichnenderweise den Untertitel *Extensions of Man* trägt, popularisieren.¹¹⁰ Engelbarts Vorstellung vom menschlichen Repertoire zur Erweiterung natürlich gegebener sinnlicher, motorischer und mentaler Fähigkeiten, kulminieren im H-LAM/T-System, „Human using Language, Artifacts and Methodology in which he is trained.“¹¹¹ Jeder Mensch besitze ein bestimmtes Repertoire an Fähigkeiten, die er in Problemlösungsprozessen anwendet. Geistige Arbeit sei daher vergleichbar mit einem „tool kit“ und „just as the mechanic [...] the intellectual worker must know the capabilities of his tools and have good methods, strategies, and rules of thumb for making use of them.“¹¹² Das Zusammenspiel der individuellen Fähigkeiten und den zur Verfügung stehenden Hilfsmitteln bedinge die Effektivität menschlichen Handelns und Denkens. So polemisierte Engelbart gegen die damaligen Bestrebungen der AI-Forschung, „the development of ‚artificial intelligence‘ has been going on for centuries.“¹¹³ In seinem soziotechnischen Systementwurf sind der Mensch und „computers, and computer-controlled information-storage, information-handling, and information-display

¹⁰⁶Ebda, 2c4j.

¹⁰⁷Ebda, 3b10d.

¹⁰⁸Engelbart, Douglas et.al. (1973b): „The Augmented Knowledge Workshop“, <http://www.doungelbart.org/pubs/augment-14724.html> (23.11.2010), 4e3b.

¹⁰⁹Engelbart (1962): 1a3.

¹¹⁰Vgl.: McLuhan, Marshall (1964): *Understanding Media. The Extensions of Man*.

¹¹¹Engelbart (1962): 2a13.

¹¹²Ebda, 2a12.

¹¹³Ebda, 2c2c.

devices”¹¹⁴ die zwei zentralen physischen Komponenten, die es nun im Sinne einer möglichst effizienten Interaktion aneinander anzupassen gilt. Der Austausch zwischen den beiden Systemkomponenten, der sich in der hybriden Entität des Interface ereignet, basiert auf einer Koppelung von Prozessen, die explizit der menschlichen und der artifiziellen Seite zugerechnet werden können. Das Interface ist jener Raum, in dem sich diese zwei strukturell unterschiedlichen Agenten mit ihren genuinen Möglichkeiten treffen, um durch geeignete Repräsentationsmittel miteinander in Kontakt zu treten.

„Where a complex machine represents the principal artifact with which a human being cooperates, the term ‚man-machine interface‘ has been used for some years to represent the boundary across which energy is exchanged between the two domains. However, the ‚man-artifact interface‘ has existed for centuries, ever since humans began using artifacts and executing composite processes.”¹¹⁵

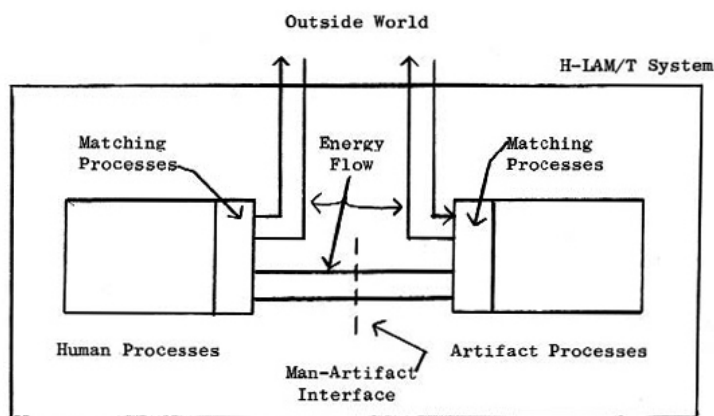


Abb.5: Engelbart's H-LAM/T-System

Zumal Engelbart's Ziel, ein System zum „besseren Denken“ in den frühen 60ern aus dem Nichts zu erschaffen, einigermaßen utopisch war, musste notwendigerweise eine effiziente Entwurfsmethodik gewählt werden um sich großen Problemstellungen schrittweise, durch gezieltes Lösen von Einzelproblemen anzunähern. Engelbart nannte seine Vorgehensweise Bootstrapping. Bootstraps sind Schnürriemen, die es einem erlauben, ähnlich wie Münchhausen, sich an den eigenen Stiefeln aus dem Sumpf zu ziehen. Kern dieses Konzeptes ist es, Werkzeuge und Methoden zu entwickeln, um auf deren Grundlage bessere Methoden und Werkzeuge zu entwickeln und sich so schrittweise dem Gesamtziel anzunähern.¹¹⁶ „The better we

¹¹⁴Ebda, 2a5.

¹¹⁵Ebda, 2c3b.

¹¹⁶Die Methode des Bootstrapping findet sich nicht nur in den Computerwissenschaften; so verweist etwa Friedewald auf Karl Marx, der den Werkzeugmaschinen in der industriellen Revolution eine entscheidende Rolle zuwies, da mit ihnen wiederum leistungsfähigere Werkzeuge und Maschinen hergestellt werden konnten. Vgl.: Friedewald (1999): S. 159.

get, the better we get at getting better.”¹¹⁷ Und selbst wenn das Ziel weit in der Zukunft liegt, müsse beim vorhandenen Wissen und verfügbaren Mitteln angesetzt werden.

5.7.1. Praktische Realisierung des Hardware-Interfaces - die Entwicklung der Maus

Die praktischen Arbeiten Engelbarts und seiner Mitarbeiter begannen daher mit der Erforschung von effizienten Aus- und Eingabegeräten, welche die sensomotorischen Fähigkeiten des Menschen am Besten bedienten. Dass nur ein Monitor als Ausgabemedium geeignet war, stand außer Frage. Zur damaligen Zeit war die interaktive Textverarbeitung direkt am Monitor noch ein Fremdwort.¹¹⁸ Da jeder Benutzer über seinen eigenen Arbeitsplatz¹¹⁹ verfügen sollte, ein 17 Zoll-Computermonitor mit entsprechend guter Auflösung und Flimmerfreiheit, jedoch ca. 20 000 Dollar teuer war, fand John Yarborough 1964 in der Konvergenz von Fernseh- und Computertechnik einen Kompromiss. An den Hauptrechner wurden zwölf hochauflösende kleine Vektormonitore angeschlossen, deren Bilder von einer Fernsehkamera zu herkömmlichen schwarz-weiß Videomonitoren der Arbeitsplätze weitergeleitet wurden. Das Verwenden von Videosignalen brachte außerdem die Möglichkeiten mit sich, das Bild eines entfernten Gesprächspartner zu übertragen (das, was man heute Video-Konferenz nennt), sowie das gemeinsame Bearbeiten desselben Dokuments, was haargenau in Engelbarts Konzept einer kollektiven Problemlösung passte (das, was heute als „computer-supported collaborative work“ definiert wird).

Für die Dateneingabe wurden mehrere Varianten vorgeschlagen, z.B. eine Zehnfingertastatur oder ein Datenhandschuh mit druckempfindlichen Sensoren. Tatsächlich realisiert wurde eine (Fünffinger)Einhandtastatur, die zwar in Engelbarts Forschungsgruppe neben der herkömmlichen Schreibmaschinentastatur verwendet wurde, sich aber kommerziell nie durchsetzen konnte. Auch spätere Anläufe zur Etablierung ergonomischer Einhandtastaturen, etwa von Xerox und IBM scheiterten alle aufgrund der zu hohen Einübungszeit. Für Engelbart stellte der vergleichsweise hohe Trainingsaufwand jedoch kein Hindernis dar, zumal die Effektivität

¹¹⁷Engelbart (1995b)

¹¹⁸Dies hatte auch ökonomische Gründe; so weist Friedewald darauf hin, dass das Editieren von Text an einigen Universitätsrechenzentren bis in die 80er explizit verboten war, Rechenzeit war nämlich knapp und teuer. Vgl.: Friedewald (1999): S. 175.

¹¹⁹Wie Friedewald schreibt, sahen die meisten Bildschirmterminals zu dieser Zeit wie „Gefechtsstände in einem U-Boot aus, bei denen Bildschirm und Tastatur fest in einen Metallschrank eingebaut waren“. [Friedewald (1999): S. 199.] Engelbart brach mit dieser Tradition und legte viel Wert darauf, dass der Benutzer einen eigenen Arbeitsplatz hatte, den er sich nach seinen Vorlieben einrichten konnte. Am SRI erfreuten sich beispielsweise, der Hippie- und Gegenkulturbewegung entsprechend, spezielle „Yoga-Workstations“, bei denen im Lotussitz gearbeitet wurde, einer großen Beliebtheit.

nach der Lernphase deutlich höher war. Das ist ein gutes Beispiel für Engelbarts Idee einer Koevolution im H-LAM/T- System. Ihm ging es nicht in erster Linie darum, eine intuitive Bedienbarkeit zu erreichen, sondern um die Verbesserung der menschlichen Fähigkeiten, komplexe Probleme besser in den Griff zu bekommen. Dazu müsse man sich eben auf neue Strukturen einlassen.

Der entscheidende Faktor, weshalb sich die Schreibmaschinentastatur als historisch gewachsener Standard durchsetzte, lag letztendlich in der wirtschaftlichen Relevanz, die sich durch die bereits weitverbreitete Fähigkeit, mit ihr umzugehen, ergab.¹²⁰ Daher bestimmt auch heute noch großteils die topografische Anordnung der Tasten à la QWERTY(Z), die bei mechanischen Schreibmaschinen, durch eine möglichst weit auseinanderliegende Platzierung häufig auftretender Buchstaben der Sprache, das Verhaken von Typenhebeln verhindern sollte, die Choreographie der Fingerspitzen.¹²¹ Und das, obwohl bereits in Sachen Ergonomie und Schreibgeschwindigkeit weitaus effizientere Vorschläge, wie z.B. die Dvorak-Tastatur, vorliegen würden.

Neben der Dateneingabe war die Möglichkeit einer direkten Manipulation von Bildschirmobjekten von großer Bedeutung, was in enger Verbindung mit der Orientierungsfähigkeit im virtuellen dynamischen Informationsraum steht. Einzelne Buchstaben, Wörter, Sätze oder Paragraphen müssen, um einer dynamischen Wissensumgebung gerecht zu werden, präzise angesteuert und ausgewählt werden können. Daher experimentierte das Labor mit verschiedenen Geräten, wie z.B. dem bereits etablierten Joystick und dem Lichtgriffel (Abb.7). Zusätzlich wurde am Institut 1964 die Maus als alternatives Eingabegerät entwickelt. Sie bestand aus einem hölzernen Gehäuse, indem zwei Räder zur horizontalen und vertikalen Bewegung mit Potentiometern verbunden waren, welche die Bewegung in digitale Signale transformierten, die im zweidimensionalen Koordinatensystem des Monitor die x und y Position des Cursors definierten. Außerdem wurde eine Knieststeuerung (Abb.6) entworfen, die sich am Beispiel der Steuerung des Autos durch Pedale orientierte. Der Cursor, damals hieß er noch „bug“¹²², wurde dabei durch einen, oberhalb des Knies angebrachten Metallbügel mit Hebung und Senkung

¹²⁰Nimmt man an, es gäbe 100 Millionen Menschen, welche mit alternativen Eingabegeräten umgehen könnten und deren Ausbildung und Training im Erwerb dieser Fähigkeiten überschlagsmäßig auf 10 \$ pro Stunde käme, so besitzt die herkömmliche Schreibmaschinentastatur einen Wert von 100 Milliarden Dollar. Vgl.: Friedewald (1999), S. 175.

¹²¹Der *Sholes & Glidden Typewriter*; die erste Schreibmaschine mit dieser Tastaturbelegung, wurde 1874 von einer amerikanischen Waffen- und Nähmaschinenfabrik auf den Markt gebracht. Die strikt alphabetische Anordnung der Tasten verursachte nämlich bei schnellen Typen von zwei nebeneinanderliegenden Buchstaben (wie die häufige Kombination DE, EF) ein Verhaken der Typenhebel.

¹²²English, William et al.(1967): „Display-Selection-Techniques for Text Manipulation“, <http://www.doungengelbart.org/pubs/augment-133184.html> (13.11.2010), 1c3.

des Beines vertikal, durch seitlichen Bewegungen horizontal bewegt. Diese Geräte wurden am SRI hinsichtlich ihrer Effizienz zu einer schnellen Ansteuerung von „targets“¹²³, so wurden die Buchstaben bzw. Wörter am Monitor genannt, in Versuchsanordnungen miteinander verglichen. Am Beginn der Textverarbeitung stand somit die Zielerfassung, wie einst bei SAGE. „Die Feindakquisition fand sich als Maus auf einem normalen Schreibtisch wieder“¹²⁴, schreibt Axel Roch. Es stimmt zwar, dass der Lichtgriffel ein direkter Nachfahre der Lightgun ist, der einst zur taktischen Kontrolle des Luftraums diente, dennoch muss eine monokausale Rückführung der Maus auf Kriegstechnologie relativiert werden, zumal Engelbarts Ziel der Unterstützung bzw. Verbesserung des menschlichen Intellekts nicht von militärischem Interesse war.

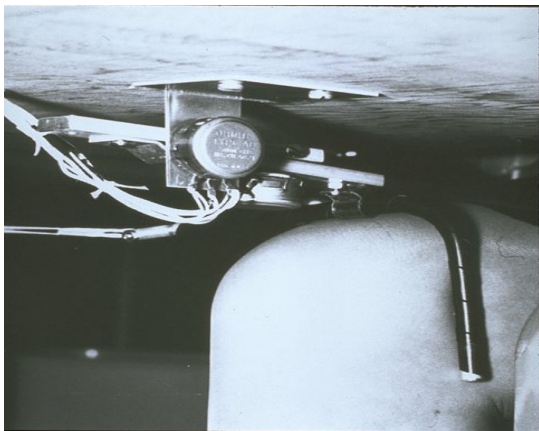


Abb.6: Experimentelle Kniestuerung

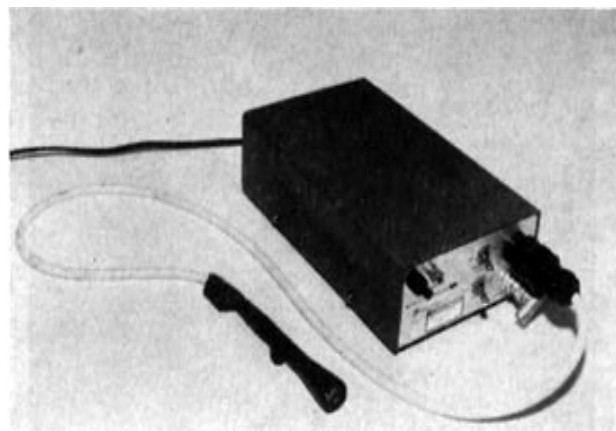


Abb.7: Lichtgriffel

Die Versuchspersonen (geteilt in geübt und ungeübt) mussten mithilfe des Eingabegerätes die am Bildschirm an einer zufälligen Stelle erscheinenden „targets“ ansteuern und auswählen. Nach dem Erscheinen musste eine Taste betätigt werden, die dem Computer signalisierte, dass mit der Ansteuerung begonnen wird. Nach der Erfassung wurden die „targets“ durch einen Knopfdruck auf dem Eingabegerät markiert. Eine Statistikroutine im Hintergrund berechnete die Zeit, die die Versuchsperson zum Griff zum Eingabegerät, dann bis zum Ansteuern brauchten, sowie die Fehlerquote der Auswahl. Ganz im Sinne arbeitswissenschaftlicher Bewegungsstudien wurden die Handlungen der Personen genau vermessen und elementarisiert. Anhand der Effizienz der Steuerung des „bugs“ geronnen die Versuchspersonen zu denjenigen, die „debugged“ wurden. Die Maus schnitt, knapp gefolgt von der Kniestuerung am besten ab. Der Lichtgriffel wurde wegen der Möglichkeit der direkten Markierung am Bildschirm

¹²³Ebda.

¹²⁴Roch, Axel (1998): „Die Geschichte der Computermaus. Von der elektrischen zur taktischen Feuerleitung“, <http://www.heise.de/tp/r4/artikel/2/2440/1.html> (19.11.2010)

zwar von ungeübten Benutzern bevorzugt, doch das Ausstrecken des Arms führte zu einer schnellen Ermüdung, was ergonomisch unvorteilhaft ist. Engelbart fasst die Vorteile der Maus zusammen:

„[I]t stays put when your hand leaves it to do something else (type, or move a paper), and re-accessing proves quick and free from fumbling. Also, it allows you to shift your posture easily, which is important during long work sessions with changing types and modes of work.“¹²⁵

1970 wurde die Maus patentiert, ihre massenweite Ausbreitung als dominantes Eingabemedium, was natürlich mit technischen Verbesserungen einherging, erfolgte erst ab den 80ern. Die Einführung der Maus war ein wegweisender Schritt in der Mensch-Computer-Interaktion und in Richtung grafischer Benutzeroberfläche. Als nahtlose Extension der Hand etabliert sie eine direkte Relation zwischen dem topographischen Raum der Benutzeroberfläche und der menschlichen Gestik. Weiters ist die Entwicklung der Maus ein frühes, exzellentes Beispiel für die Erforschung kognitiver Fähigkeiten des Benutzers¹²⁶, die in die Gestaltung der Interaktion mit dem Computer überführt wurden.



Abb.8: Bildschirmarbeitsplatz, an dem 1963/64 am SRI die Experimente mit grafischen Eingabegeräten durchgeführt wurden; rechts: Maus

¹²⁵Engelbart, Douglas (1973a): „Design Considerations for Knowledge Workshop Terminals“, <http://www.doungengelbart.org/pubs/augment-14851.html> (23.11.2010), 2b4b.

¹²⁶Die Maus ist ein typisches Beispiel der Materialisierung des Fitt'schen Gesetzes, welches in den 50ern vom Experimentalpsychologen Paul Fitts als eine mathematische Relation ausgedrückt wurde. Es beschreibt den Zusammenhang zwischen der Bewegungszeit, der Weite und der Genauigkeit gezielter Bewegungen. Die Reaktionszeit ist also abhängig von der Entfernung und Größe des Zielgegenstandes. Die Maus erwies sich bezüglich der Präzision und Schnelligkeit der Zielerfassung als beste Annäherung an das natürliche Zeigen mit dem Finger, was auf die Mensch-Computer-Interaktion umgemünzt bedeutet, dass die Begrenzungen der Zielerfassung mit der Maus jenen des menschlichen Nervensystems entsprechen.

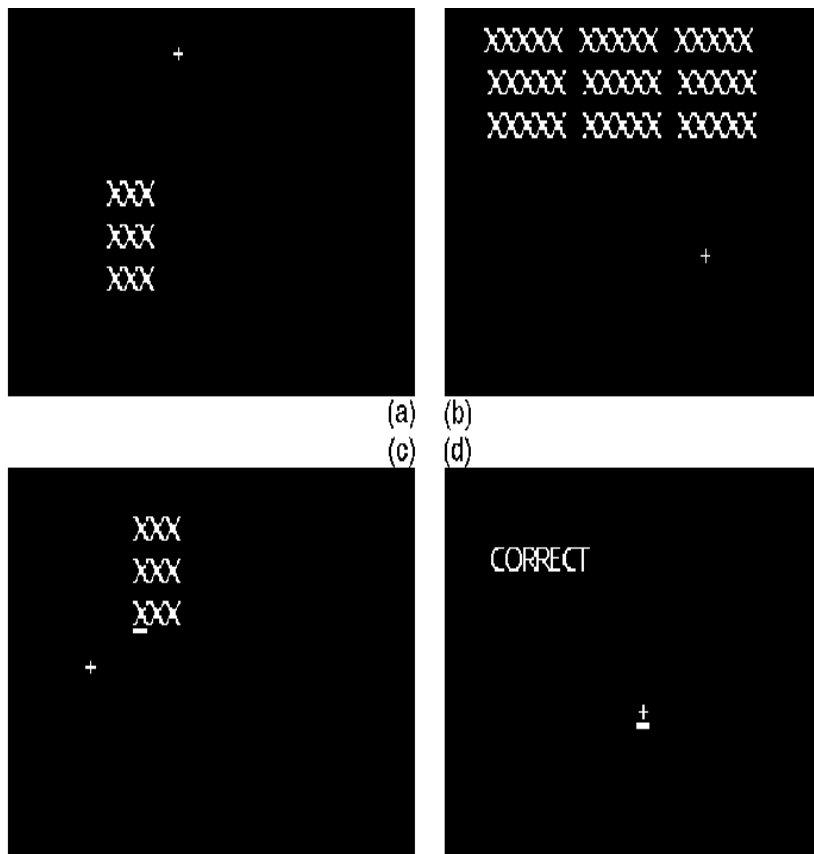


Abb.9: Versuchsanordnung der Textauswahl-Experimente

- a) „Character mode“: das mittlere X muss ausgewählt werden
- b) „Word mode“: die mittleren fünf X sind das Auswahlziel
- c) fehlerhaft ausgewähltes Zeichen wird unterstrichen
- d) korrekte Auswahl: Position des Ziels wird durch den unterstrichenen Cursor markiert

5.7.2. NLS - die Demonstration

Am 9.12.1968 fand eine der spektakulärsten Demonstrationen der Computergeschichte in San Francisco auf der Fall Joint Computer Conference statt. Durch gemietete Telefonleitungen und Richtfunkstrecken wurde eine Verbindung zwischen San Francisco und dem Computersystem am SRI hergestellt. William English dirigierte in San Francisco hinter der Bühne das Abmischen der Computer- und Videosignale. Die technische und inszenatorische Glanzleistung konnte von den Besuchern auf einer 6 Meter hohen Leinwand mitverfolgt werden.¹²⁷ In 90 Minuten präsentierte und erklärte Engelbart die wichtigsten Funktionen und die Bedienung des OnLineSystems, kurz NLS, das praktische Ergebnis seiner Forschungen. Die Demonstration zeigte eine zuvor unbekannte Form der interaktiven Zusammenarbeit zwischen

¹²⁷Engelbart selbst spricht von ca. 800 Besuchern, manche spätere Darstellungen von 2000 bis 3000. Wie Friedewald vermutet, wollten sich nachträglich anscheinend mehr Leute als Zeugen verstanden wissen. Vgl.: Friedewald (1999): S. 215.

Mensch und Computer. „The functions are [...] a set of controls easily moving one around in an information space and allowing one to adjust the scope, format, and content of the information seen.”¹²⁸ Die Steuerungsbefehle des NLS erfolgten durch Eingabe von Kurzkommandos, meist aus zwei Buchstaben (z.B.: DW: delete word, IW: insert word) bzw. Spezialkommandos (z.B. QI: automatic renumbering turned on, QJ: automatic renumbering turned off).¹²⁹ Die Einhandtastatur diente einer schnellen Eingabe der Befehlskombinationen, ähnlich wie Akkorde auf einem Klavier. Mit der Maus konnten Bildschirmobjekte schnell angesteuert und selektiert werden. Weiters zeichnete sich bereits die Strukturierung des Bildschirms in einen funktionellen, operativen Teil (zur Eingabe von Textkommandos) und einen inhaltlichen Teil (z.B. der Ansicht eines Dokuments) ab. Außerdem konnten durch das Splitten des Monitors in mehrere Arbeitsbereiche Informationen übersichtlich dargestellt und miteinander verglichen werden. Engelbart fasste dies unter „multiple windows“¹³⁰ zusammen. Um den pragmatischen Wert seines Systems zu betonen, verwendete er zur Demonstration eine Einkaufsliste. Dabei ließen sich einzelne Elemente, z.B. die Wörter Banana, Paper etc. zu Gruppen kategorisieren, die sich verbergen und aufklappen ließen. Es konnte angegeben werden, wie viel und was man von dem jeweiligen Dokument sehen wollte. Auch konnten einzelne Elemente wie Wörter, Sätze, Paragraphen u.ä. miteinander verlinkt werden, was somit, referierend auf Vannevar Bush, eine neue Art der Sichtweise auf die Verbindungen der Informationen innerhalb komplexer Strukturen ermöglichte. Und nicht zuletzt wandelte Engelbart die Strecke seiner Einkaufsbesorgungen in eine visuelle Darstellung um, einen einfachen Graphen, der Punkte (Orte) mit Linien verband, und auch gleich diejenigen Produkte anzeigte, die er bei den spezifischen Orten zu besorgen hatte. Im NLS war es also bereits möglich, unterschiedliche Ansichten derselben Informationsstruktur zu sehen. Neben dem radikal neuen, flexiblen und interaktiven Umgang von Informationen, versetzte Engelbart seine Zuschauer, darunter auch Alan Kay, abermals ins Staunen, als er mit seinem Kollegen, der am Institut des SRI vor seinem Terminal saß, über den Bildschirm Angesicht zu Angesicht kommunizierte und an einem gemeinsamen Dokument arbeitete - die früheste Form von „computer-supported collaborative working“ und zugleich die Verdichtung des Computers zu einem Kommunikationsmedium. Der Monitor avancierte hier zum ersten Mal zu einem dynamischen Interaktions- sowie zu einem intersubjektiven Kommunikations- und Kollaborationsraum. Die Innovationen waren

¹²⁸Engelbart (1973b): 4c1a.

¹²⁹Vgl.: Friedewald (1999): Übersicht der NLS-Befehle, S. 415-421.

¹³⁰Engelbart (1973b): 4c4a.

enorm. Dennoch neigte sich Engelbarts Projekt 1974 dem Ende zu.¹³¹ Die Gründe für das Scheitern des Projekts waren vielfältig. Das OnLineSystem verschwand wegen des hohen Lernaufwands und der Komplexität des Systementwurfes von der Bildfläche, obwohl es diese zumindest in Hinsicht der Mensch-Computer-Interaktion revolutionierte. Zwar war es hochentwickelt und in vielerlei Hinsicht eine Meisterleistung, doch für computerunerfahrene „knowledge worker“ nicht wirklich benutzerbar, weil zu kompliziert. Außerdem kam zunehmend Widerstand von seinen Mitarbeitern, jungen Wissenschaftlern, die geprägt von den ideologischen Prämissen der damaligen Gegenkultur, mit der konservativen Arbeitseinstellung am SRI, welches in militärische Forschungen und enge Kooperation mit führenden Industrieunternehmen eingebunden war, nichts anfangen konnten.¹³² Es ist daher nicht verwunderlich, dass viele von Engelbarts Kollegen Anfang der 70er zu Xerox PARC, das nur ein paar Kilometer vom SRI entfernt lag, wechselten. Weiters wurde das Prinzip des Time-Sharing zunehmend uninteressant. Der einstige Zweck war es ja, die teure Rechenleistung auf mehrere Benutzer aufzuteilen, zumal eine alleinige Benutzung von Computern ökonomisch nicht realisierbar war. Durch die Veränderung technologischer Randbedingungen - die Entwicklung integrierter Schaltkreise, fortschrittliche Halbleitertechnik und beginnende Miniaturisierung - begannen auch die Preise für Rechenleistung zu sinken. Die Zukunft gehörte also dem Personal Computer.



Abb.10: Screenshot der NLS- Demonstration 1968; der Bildschirm wurde in mehrere, nicht überlappende Bereiche unterteilt (tiled windows); rechts: Douglas Engelbart

¹³¹Engelbarts Vision lebt in dem 2006 fertiggestellten HyperScope 1.0 weiter, dem der ambitionierte Versuch zugrunde liegt, Hypertextfunktionen des NLS in einen Webbrowser zu integrieren. Vgl.: <http://hyperscope.org> (20.11.2010)

¹³²Vgl. Friedewald (1999): S. 233ff.

6. Die Entwicklung der grafischen Benutzeroberfläche

Während der 70er, im kalifornischen Xerox PARC, nahm das grafische User-Interface unter dem Einfluss neuer Leitideen der Mensch-Computer-Interaktion langsam seine Gestalt an. Das Ziel von Xerox, die Entwicklung eines papierlosen Büros der Zukunft, und die Vision junger Computerwissenschaftler vom Computer als ein persönliches Medium konvergierten um dieselbe Herausforderung: die Entwicklung eines leicht erlernbaren und intuitiv zu bedienenden Systems. Dieser Perspektivenwechsel ebnete den Weg zur Kommerzialisierung des PCs mit grafischer Benutzeroberfläche und katapultierte den Computer endgültig von einem technokratischen Element der Expertenkultur zu einem festen Bestandteil der Medienkultur. Das folgende Kapitel will wesentliche Rahmenbedingungen und die wichtigsten Einflüsse offenlegen, die den Nährboden der Entwicklung der grafischen Benutzeroberfläche bildeten und dem Gesicht des Computers seine heutigen Konturen verliehen.

6.1. Xerox PARC - die institutionelle Rahmenbedingung

„Raster graphics-based networked workstations and ‚point-and-click‘ WIMP GUIs (graphical user interfaces based on windows, icons, menus, and a pointing device, typically a mouse) are the legacy of Xerox PARC that we’re still using today.“¹³³

Von der 1906 gegründeten Firma Xerox stammt das Verfahren der Xerographie bzw. der Elektrofotokopie, das 1938 der Startschuss für die kommerzielle Etablierung von Fotokopierern war. Rund 30 Jahre später sah sich der führende Konzern für Bürotechnologien in seiner Marktherrschaft von aufstrebenden japanischen Firmen gefährdet, was neue Geschäftskonzepte herausforderte. Ein Expertenteam sollte deshalb in einer Denkfabrik ein papierloses Bürosystem für die Zukunft designen - denken statt kopieren lautete die Devise. Damit einher gingen auch Bestrebungen, sich in der Branche der digitalen Datenverarbeitung zu etablieren, und somit bewusst in Konkurrenz mit dem damals übermächtigen IBM zu treten. Daher gründete Xerox 1970 ein Forschungslaboratorium für Computerwissenschaft und Halbleitertechnik in unmittelbarer Nähe der Stanford University am Rande des Silicon Valley - das Palo Alto Research Center, kurz PARC. Um dem von Xerox formulierten Ziel einer „architecture of information“¹³⁴ gerecht zu werden, wurden viele exzellente und kreative Wissenschaftler angeworben, die in unterschiedlichen Schwerpunkten ihre Ideen einfließen lassen sollten.

¹³³van Dam (1997): S. 63.

¹³⁴Lampson, Butler (1986): „Personal Distributed Computing: The Alto and Ethernet Software“, <http://portal.acm.org/citation.cfm?id=12186> (10.11.2010), S. 102.

Darunter waren u.a. Alan Kay, einer der einflussreichsten Pioniere des grafischen User-Interfaces, David Canfield Smith, der Namensgeber der Icons, Butler Lampson, Charles Thacker sowie eine Reihe von Engelbarts Mitarbeitern wie etwa seine wichtigsten Softwareentwickler Jeff Rulifson und Bill Paxton sowie William English, Engelbarts engster Mitarbeiter und Mitentwickler der Maus. Im Xerox PARC wurden während der 70er und den frühen 80er Jahren die Grundelemente des Personal Computing ausgearbeitet, was Hand in Hand mit der Entwicklung der grafischen Benutzeroberfläche ging. Lange bevor die eigentliche PC-Revolution durch geschickte Vermarktung des Produktes begann, nämlich schon Anfang der 70er, wurde die Leitidee des PCs geprägt, deren Realisierung sich zunächst im „Alto“ (1973), dem ersten PC mit grafischer Benutzeroberfläche, später im Xerox Star (1981) niederschlug. Obwohl Xerox bereits alle wesentlichen Voraussetzungen hatte, gelang der Firma nie der Sprung in den Massenmarkt. Erst Apple, später auch Microsoft sollten das Erbe von Xerox PARC in kommerziell erfolgreich Produkte überführen. Sven Stillich schreibt dazu:

„Xerox hat seine große Chance vertan. Ihre Ideen finden sich heute in jedem Rechner, aber auf keinem dieser Rechner steht ihr Name. ‚Sie hatten den größten Sieg der Computergeschichte vor Augen‘, sagt Steve Jobs, ‚sie hätten das IBM oder das Microsoft der Neunziger werden können und heute die Industrie dominieren.[sic!]Aber sie wurden geschlagen‘ - von einem charismatischen Hippie mit einer Vision und einem Studienabbrecher, den der PC zu einem reichsten Menschen [sic!] der Welt gemacht hat.“¹³⁵

6.2. Alan Kay - FLEX und das „Dynabook“

„The best way to predict the future is to invent it!“¹³⁶

Alan Kay (*1940), einer der einfluss- und ideenreichsten Denker und Praktiker in der Computergeschichte, erwarb seinen Bachelor in Biologie und Mathematik und wechselte Mitte der 60er als Doktorand der Computerwissenschaften auf die University of Utah, wo 1968 niemand geringer als Ivan Sutherland lehrte. Damals arbeitete er gemeinsam mit Edward Cheadle, noch mit finanzieller Unterstützung der ARPA, an der FLEX-Maschine - einer frühen Variante eines persönlichen Desktop-Computers. Kay verfolgte zwar theoretisch ähnliche Ziele wie bereits Licklider und Engelbart nämlich „to design an interactive tool which can aid in the visualization and realization of provocative notions“¹³⁷, doch sein Fokus lag von Anfang an

¹³⁵Stillich, Sven (2008): „Weltherrschaft verschlafen“, http://einestages.spiegel.de/static/topicalbumbackground/3046/weltherrschaft_verschlafen.html (18.11.2010)

¹³⁶Kay, Alan (1990): „User Interface. A Personal View“, S. 206.

¹³⁷Kay, Alan (1969): „The Reactive Engine“, <http://www.mprove.de/diplom/gui/kay69.html> (23.11.2010)

auf einem Systementwurf, der auch computerunkundigen Benutzern entgegenkam. So schreibt Butler Lampson retrospektiv:

„Kay was pursuing a different path to Licklider's man-computer symbiosis: the computer's ability to simulate or model any system, any possible world, whose behavior can be precisely defined. And he wanted his machine to be *small, cheap, and easy for non-professionals to use*. Like Engelbart, he attached great importance to a high-quality, rapidly-changing display.“¹³⁸

Der programmatische Name FLEX, ein Akronym für „Flexible Extendable“¹³⁹, referiert auf die Programmierung der „Reactive Engine“ - so der Name der Dissertation, in deren Rahmen Kay das Projekt vorstellte. Erst die Software nämlich „gives form and purpose to a programmable machine.“¹⁴⁰ Kay legte großen Wert darauf, dass sie in ihrer Funktionalität für verschiedene Anwendungen erweitert werden konnte, denn, so Kay, „[m]achines which do one thing only are boring.“¹⁴¹ Zusammenfassend formulierte er drei wichtige Anforderungen an die Gestaltung eines Systems, die zugleich auch als Grundlage des Personal Computing begriffen werden können: „The communication device must be available (in every way) as a slide rule. The service must not be esoteric to use (It must be learnable in private). The transactions must inspire confidence (‘Kindness’ should be an integral part).“¹⁴² Im Gegensatz zu Engelbart, der die Mensch-Computer-Interaktion als eine Koevolution von Computer und Benutzer begriff, postulierte Kay von Anfang an eine dezidierte Anpassung des Computers an die menschlichen Bedürfnisse. Die FLEX verfügte über eine Tastatur und ein Grafiktablett als Eingabegeräte sowie über einen hochauflösenden text- und grafikfähigen Bildschirm. Trotz dieser exzellenten hard- und softwaretechnischen Voraussetzungen schien das User-Interface des, immerhin noch 160 kg schweren Gerätes, die potentiellen Endbenutzer jedoch eher abzuschrecken - es ergab einfach kein einheitliches Ganzes.¹⁴³

Als Mitglied der ARPA-Gemeinde hatte Kay einen guten Einblick in die neuesten technologischen Forschungen, die bei gemeinsamen Treffen anhand aktueller ARPA-Projekte präsentiert wurden. 1968 führten Studenten der University of Illinois den Prototyp eines Plasmaflachbildschirms vor. Kay war der enorme Wert bewusst, den dieses „1’ square lump of glass and neon gas in which individual spots would light up on command“¹⁴⁴ für die Realisierung eines PCs hatte. Auf den Erkenntnissen von Moore’s Gesetz stellte er Berechnungen an, wie lange es dauern würde, bis die Elektronik der, immer noch wuchtigen, FLEX hinter einem solchen

¹³⁸Lampson (1986): S.101 f. Hervorhebung, d.A.

¹³⁹Kay (1969)

¹⁴⁰Kay, Alan (1984): „Computer Software“, S. 41.

¹⁴¹Kay (1969)

¹⁴²Ebda.

¹⁴³Vgl.: Kay (1990): S. 192.

¹⁴⁴Kay (1993a): S. 73.

Bildschirm verschwinden könnte. Anfang 80er, so spekulierte Kay „we could all have an inexpensive powerful notebook computer- I called it ‚personal computer‘ then, but I was thinking *intimacy*.“¹⁴⁵ Bereits Ende der 60er geisterte also in Kays Kopf die Vision eines Laptops herum, die von Anfang an mit dem Gefühl bzw. gar der Forderung der Intimität verknüpft war, wie es sich später bewahrheiten sollte. Neben diesem technologischen Impuls, der tatsächlich erst nach den 80ern Realität zu werden begann, waren es auch die Gedanken Marshall McLuhans, die Kays Vorstellungen, von dem was ein Computer denn eigentlich sei, grundlegend beeinflussten. Durch die Lektüre von *Understanding Media* kam er zur Einsicht, dass der Computer, von dem er immer eher als ein Werkzeug gedacht hatte, eigentlich ein

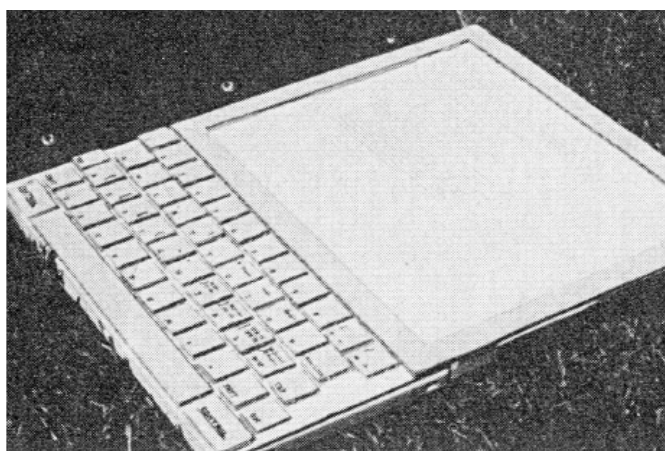


Abb.11: Alan Kays Kartonmodell vom Dynabook

Medium sei, welches dem Buch gleich, das Potential besitze „thought patterns of an entire civilization“¹⁴⁶ zu ändern. Außerdem erschien ihm „[t]he intensely interactive and involving nature“ des Computers einerseits als ein notwendiger Gegenpol zur passiven TV- Berieselung, „a remedievalizing tribal influence at best.“¹⁴⁷ Andererseits betonte auch Kay, wie bereits Engelbart, das Potential des Computers, die statische Repräsentationsebene des Papiers durch dynamische Simulation zu ersetzen, erweiterte diesen Aspekt jedoch um die Simulation aller bereits existierenden medialen Formate.

„Although digital computers were originally designed to do arithmetic computation, the ability to simulate the details of any descriptive model means that the computer, viewed as a medium itself, can be *all other media* if the embedding and viewing methods are sufficiently well provided.“¹⁴⁸

Kay formulierte damit explizit zum ersten Mal die multimedialen Möglichkeiten des Computers. In seiner Vorstellung gerinnt der Computer zu einem persönlichen „metamedium, whose

¹⁴⁵Kay (1990): S. 192.

¹⁴⁶Ebda, S. 193.

¹⁴⁷Ebda.

¹⁴⁸Kay, Alan/Goldberg, Adele (1977): „Personal Dynamic Media“, S. 393 f.

content would be a wide range of already-existing and not-yet invented media.”¹⁴⁹ Die Idee des Dynabook, eine Fortführung und Erweiterung des statischen Mediums des Buches, das konzeptuelle Vorgängermodel heutiger Laptops, war geboren. Das Dynabook sollte, in der Tradition der FLEX, eine intuitive Benutzung und Programmierung erlauben, eine hochwertige Grafikausgabe besitzen und ein für jedermann erschwingliches, tragbares „dynamic medium for creative thought“¹⁵⁰ sein.

„Imagine having your own self-contained knowledge manipulator in a portable package the size and shape of an ordinary notebook. Suppose it had enough power to outrace your senses of sight and hearing, enough capacity to store for later retrieval thousands of page-equivalents of reference materials, poems, letters, recipes, records, drawings, animations, musical scores, wave forms, dynamic simulations, and anything else you would like to remember and change.”¹⁵¹

6.3. Seymour Papert und LOGO

Kurz nach der medientheoretischen Erleuchtung machte Kay Bekanntschaft mit Seymour Papert, der gemeinsam mit Marvin Minsky das *Artificial Intelligence Laboratory* am MIT gründete. Papert, ein Mathematiker der lange Zeit mit Jean Piaget, dem Urvater der kognitivistischen Psychologie in Genf zusammenarbeitete, leitete zu dieser Zeit ein Projekt, welches sich intensiv mit den Möglichkeiten des Computers zum selbstständigen Lernen für Kinder auseinandersetzte. Papert war stark geprägt durch Piagets Pionierleistungen auf dem Gebiet der Entwicklungs- und Erziehungstheorie. Eine der wichtigsten Annahmen Piagets ist, dass das Denken des Kindes ab der Geburt von haptisch und sinnlich dominierter Wahrnehmung zunehmend zu immer differenzierteren abstrakt-begrifflichen Grundlagen fortschreite. Er unterteilt diesen Entwicklungsprozess in verschiedene, aufeinanderfolgende Stufen, in denen unterschiedliche Rezeptionsmodelle die Wahrnehmungsweise des Kindes prägen und das Denken bestimmen - die sensomotorische, die prä-operationale, die konkret-operationale und die formal-operationale Phase. Erst in der Letzteren, die Piaget ca. um das 12 Lebensjahr ansiedelt, entwickle sich die Fähigkeit zum logischen, abstrakten Denken. Papert folgerte daraus, dass es unnötig, wenn nicht gar kontraproduktiv ist, Kindern vor Erreichen dieser Phase, mathematische Konzepte in herkömmlicher Weise (wie z.B. durch Lösen von Aufgaben in Schulbüchern ohne praktischen Bezug) beibringen zu wollen. Statt Lernen als einen gerichteten Übertragungsprozess von Lehrern auf Kinder zu sehen, verfolgte Papert den Ansatz konstruktivistischer Lerntheorien. Eine Kernannahme ist, dass menschliches Lernen und Erleben ein akti-

¹⁴⁹Ebda, S. 403.

¹⁵⁰Kay/Goldberg (1977): S. 394.

¹⁵¹Ebda.

ver, selbstorganisierter Prozess ist, der sowohl durch eigene Wertigkeiten und Erfahrungen als auch durch eine ständige Auseinandersetzung mit der Umwelt geprägt ist. Papert wollte die Fähigkeiten des Lernens und des Problemlösens der Kinder durch einen spielerischen Umgang mit dem Computer fördern. Dazu entwickelte er mit seinen Mitarbeitern LOGO, eine Programmiersprache und Entwicklungsumgebung, mit deren Hilfe Grundschul Kinder einfache Programme schreiben bzw. bereits existierende Vorgaben kreativ weiterentwickeln konnten. Eine computergesteuerte Schildkröte etwa „with its local coordinate system (like the child, it is always at the center of its universe) became a highly successful `microworld` for exploring ideas in differential geometry.“¹⁵² Die Kinder lernten somit mathematische Konzepte, indem sie diese auf etwas anwendeten, was sie wirklich interessierte und was ihnen Spaß machte, z.B. Zeichnen mit einer Roboterschildkröte. Kay war tief beeindruckt davon, mit welcher Freude sich Kinder Computerkenntnisse aneigneten indem sie bereits existierende Programme anwendeten und diese, geleitet von ihren eigenen Interessen, weiterentwickelten. War er früher noch der Meinung gewesen, dass jeder Besitzer eines PCs über Programmierkenntnisse verfügen müsste, schien LOGO jetzt das Gegenteil zu beweisen. Die Kinder waren kreative Anwender und zugleich Entwickler von Software. Kay erinnert sich: „I was possessed by the analogy between print literacy and LOGO.“¹⁵³ Kays Dynabook-Konzeption erinnert zwar sehr stark an Laptops heutiger Ausprägung, doch es gibt einen entscheidenden Unterschied, der in eben jener „literacy“ kulminiert. Begreift man das Buch, respektive die Schrift als Medium, so sind, laut Kay, zwei Fähigkeiten erforderlich, tatsächlich Schriftkenntnisse zu besitzen und sich des Mediums zu bedienen. Einerseits muss man lesen können, um die Ideen anderer zu verstehen, andererseits schreiben, um seinen eigenen Ideen Ausdruck zu verleihen. Das Dynabook sollte deshalb schon fertige Anwenderprogramme bereitstellen, jedoch dem Benutzer auch die Möglichkeit bieten, das System nach seinen Vorstellungen selber gestalten zu können. „Thus a great deal of effort has been put into providing both endless possibilities and easy tool-making through the Smalltalk programming language.“¹⁵⁴ Wenn Kay, wie so oft, als Vater des PCs bzw. des Laptops gelobt wird, sollte dieser kritische Punkt, der heutzutage - wo die meisten PCs bereits mit fast allem, was ein Endnutzer braucht, ausgestattet sind - mehr oder weniger vergessen zu sein scheint, nicht übersehen werden. Kreativer Ausdruck durch den Computer ist heute meist beschränkt auf das Erstellen von Kunstwerken, welcher Art auch immer, mit End-User-Programmen. Kay ging es allerdings nicht darum, dem Benutzer

¹⁵²Kay (1990): S. 194.

¹⁵³Ebda, S. 193.

¹⁵⁴Ebda, S. 404.

Programmierkenntnisse aufzuzwingen, sondern er sah (und sieht immer noch) im Programmieren eine Chance, neue Denkkonzepte zu explorieren und kreatives Denken zu stimulieren.

„[L]ike the reading and writing of books, it wasn't just about getting and conveying information, but the very act of learning and doing them expands one's horizons and adds new ways of thinking about the world. In other words, programming could be *good* for people, and thus some effort should be put into designing systems that would have pedagogical benefit for both children and adults.”¹⁵⁵

6.4. Doing with Images makes Symbols

„[T]he actual dawn of user interface design first happened when computer designers finally noticed, not just that the end users had functioning mind, but that a better understanding of how those minds worked would completely shift the paradigm of interaction.”¹⁵⁶

Durch die Erfahrungen mit LOGO verlagerte sich Kays Fokus zunehmend auf Kinder als Endbenutzer. Schließlich müsse ein Medium sowohl von Kindern als auch von Erwachsenen einfach erlernt und benutzt werden können. Programmatisch nannte er das Dynabook „[a] personal computer for children of all ages”.¹⁵⁷ Die Erfahrungen mit LOGO überzeugten Kay, dass die Gestaltung eines kinderleicht zu bedienenden Systems eine Hinwendung zu Mechanismen der Wahrnehmung und des Lernen implizierte. Denn erst durch die Kenntnis der mentalen Operationen der User kann deren Externalisierung in ein entsprechendes Interfacedesign stattfinden. Infolge dessen beschäftigte sich Kay intensiv mit lern- und entwicklungstheoretischen Studien. Für die Gestaltung des User-Interface erschienen ihm vor allem die Forschungen des amerikanischen Psychologen Jerome Bruner, der auf den Pionierarbeiten von Piagets Experimenten und Erkenntnissen aufbaute und diese erweiterte, als ein fruchtbarer Ansatz. Mit Referenz auf Bruners *Toward a Theory of Instruction* (1966) bemerkt Kay:

„Our mentalium seems to be made up of multiple separate mentalities with very different characteristics. They reason differently, have different skills, and often are in conflict. Bruner identified a separate mentality with each of Piaget's stages: he called them *enactive, iconic, symbolic*.”¹⁵⁸

Die menschliche Entwicklung beginnt laut Bruner mit der enaktiven Phase. Fähigkeiten werden durch Handeln bzw. Imitation von Handlungen erworben. Der Wissenserwerb ist durch eine Auseinandersetzung mit konkreten Gegenständen gekennzeichnet. In der ikonischen Phase kommt die Fähigkeit hinzu Informationen zu verarbeiten, die bildlich repräsentiert sind. Erst in der dritten Phase entwickelt sich die Fähigkeit mit abstrakten Begriffen zu operieren,

¹⁵⁵Kay, Alan (1993 b): Vorwort in Cypher, Allen: *Watch what I do, Programming by Demonstration*, S. xii.

¹⁵⁶Kay (1990): S. 192.

¹⁵⁷Vgl.: Kay, Alan (1972): „A Personal Computer for Children of All Ages”, <http://www.mprove.de/diplom/referencesKay.html> (13.11.2010)

¹⁵⁸Kay (1990): S. 194 f.

die, wie in der Mathematik z.B. durch formale Regeln beschrieben bzw. geändert werden können. Jede Mentalität ist dabei für spezifische Aufgaben zuständig: „*enactive*: know where you are, manipulate; *iconic*: recognize, compare, configure, concrete; *symbolic*: tie together chains of reasoning, abstract.”¹⁵⁹

Entscheidend ist, dass alle unterschiedlichen mentalen Operationen, auch beim Erwachsenen, jederzeit zur Verfügung stehen. Doch da sie, so die Annahme, in unterschiedlichen evolutionären Phasen ausgeprägt wurden „there’s little probability that they can intercommunicate and synergize in more than the most rudimentary fashion.”¹⁶⁰ Kay kam daher zu dem Entschluss, dass die Bedienung wie auch die Programmierung eines Computermediums wie des Dynabooks nicht nur auf symbolischer Ebene erfolgen dürfte, sondern, dass ein gutes Interface-Design auch die enaktive und die ikonische Mentalität bedienen müsse - „the best strategy would be to try to gently force synergy between them [den unterschiedlichen Mentalitäten, d.A.].”¹⁶¹

Die symbolische Ebene hat, im Gegensatz zu deren expliziter Fokussierung im bisherigen Interface-Design, nur noch die Aufgabe „to stay with a context and to make indirect connections.”¹⁶² Hier deutet sich ein radikaler Perspektivenwechsel an. Die Rede ist weniger von einem Synergieeffekt der unterschiedlichen Fähigkeiten von Mensch und Computer, sondern davon eine Lernumgebung zu schaffen, welche die verschiedenen mentalen Operationsweisen des Menschen vereinige. Vor allem der visuellen Kommunikation kommt von da an erhöhte Aufmerksamkeit zu. Denn das, was sich dem Auge gleichzeitig und als ein Nebeneinander anbietet, gewährt dem Betrachter einen schnellen Überblick und erlaubt, Vergleiche und Relationen zu ziehen. Die Hauptaufgabe der ikonischen Mentalität ist nämlich „to be interested in everything in a scene, to dart over it as one does with a bulletin board, to change context.”¹⁶³ Kay war außerdem davon überzeugt, dass Prozesse des Problemlösens und kreativen Denkens stark in der ikonischen und sensomotorischen Mentalität angesiedelt sind. So verweist er z.B. auf Jacques Hadamards Studie über mathematische und wissenschaftliche Kreativität, die eindringlich belegt, dass

„creativity [...] is not at all linked to symbolic mentality (as most theories of teaching suppose), but that the important work in creative areas is done in the initial two mentalities - most in the iconic (or figurative) and quite a bit in the enactive.”¹⁶⁴

¹⁵⁹Kay (1990): S. 196.

¹⁶⁰Ebda, S. 195.

¹⁶¹Ebda.

¹⁶²Ebda, S. 196.

¹⁶³Ebda, S. 196.

¹⁶⁴Ebda.

Albert Einsteins aufschlussvolle Introspektion in seine Denkprozesse bestätigt diese Annahme:

"The words of the language, as they are written or spoken, do not seem to play any role in my mechanism of thought. The psychical entities which seem to serve as elements in thought are certain signs and more or less clear images which can be 'voluntarily' reproduced and combined [...] This combinatory play seems to be the essential feature in productive thought - before there is any connection with logical construction in words or other kinds of signs which can be communicated to others [...]The above mentioned elements are, in my case, of visual and some of muscular type. Conventional words or other signs have to be sought for laboriously only in a secondary stage, when the mentioned associative play is sufficiently established and can be reproduced at will."¹⁶⁵

Kay brachte seine Vorstellung der „integrativen Interaktion“¹⁶⁶, wie Pflüger dieses Konzept treffend formuliert, im Slogan „doing with images makes symbols“¹⁶⁷ auf den Punkt.

DOING	mouse	<i>enactive</i>	know where you are, manipulate
with IMAGES	icons, windows	<i>iconic</i>	recognize, compare, configure, concrete
makes SYMBOLS	Smalltalk	<i>symbolic</i>	tie together long changes of reasoning, abstract

Abb.12: Doing with images makes symbols - Alan Kays Gestaltungsmaxime für das User-Interface

Konkret bedeutet dieses Konzept, dass die Maus, physisch in der Hand und als virtuelle Extension der Hand als Cursor am Bildschirm, Icons und Objekte „greifen“ und manipulieren kann. Diese dienen zur grafischen, konkreten Repräsentation ansonsten abstrakter Konzepte. Unter der Programmier- und Entwicklungsumgebung „Smalltalk“ (siehe Kap. 6.5.1.) lassen sich damit abstrakte Ideen formulieren. „[T]his led to a 90 degree rotation of the purpose of the user interface from ‚access to functionality‘ to ‚environment in which users learn by doing‘.“¹⁶⁸

Mit der Dynabook-Vision sowie der neuen Gestaltungsprämisse waren die theoretischen Fundamente des PCs und der grafischen Benutzeroberfläche gelegt. Ein persönliches dynamisches Medium für kreatives Denken, mit dem primär via Bilder kommuniziert werden sollte, erforderte jedoch höchste Flexibilität in der Kommunikation, unmittelbare visuelle Reaktion des Systems auf die vom Benutzer ausgeführten Aktionen sowie eine hochauflösende Grafikausgabe. Hinsichtlich der Realisierung eröffneten sich zwei Wege:

„These ‚civilized‘ desires for flexibility, resolution, and response lead to the conclusion that a user of a dynamic personal medium needs several hundred times as much power as the average adult

¹⁶⁵Hadamard, Jacques (1945): *The Psychology of Invention in the Mathematical Field*, S. 142 f.

¹⁶⁶Vgl.: Pflüger (2008): S. 353.

¹⁶⁷Kay (1990): S. 196.

¹⁶⁸Kay (1993a): S. 84.

now typically enjoys from timeshared computing. This means that we should either build a new resource several hundred times the capacity of current machines and share it (very difficult and expensive), or we should investigate the possibility of giving each person his own powerful machine.”¹⁶⁹

Kay und sein Team entschieden sich für den letzteren Weg, zumal die bis dahin noch üblichen Time-Sharing-Systeme vom kreativen Standpunkt gesehen, nicht mehr als eine „slow control of crude wire-frame green-tinted graphics and square-wave musical tones“¹⁷⁰ zu bieten hatten.

6.5. Der „Alto“

Basierend auf Kays Anregung und der externen Vorgabe der Entwicklung eines „electronic office“¹⁷¹ wurde 1973 im Xerox PARC der „Alto“, ein Desktop-Computer für den Einnutzerbetrieb, der über ein lokales Netzwerk, das Ethernet, verbunden werden konnte, entwickelt. Der „Alto“ war von Anfang an als ein Experiment der Vision des PCs und nicht etwa als ein zu vermarktendes Produkt gedacht. „If our theories about the utility of cheap, powerful personal computers are correct, we should be able to demonstrate them convincingly on Alto. If we are wrong, we can find out why.“¹⁷² So Butler Lampson, einer der Entwickler des „Alto“ und neben Kay und einer Handvoll anderer Computerwissenschaftler, einer der wenigen, die die alleinige Benutzung eines Computers in einem Zeitalter, als Time-Sharing auf Minicomputern noch üblich war, favorisierten. Der „Alto“ gilt als erster Computer mit grafischer Benutzeroberfläche. Er bestand im wesentlichen aus einem Prozessor, der weniger auf reine Rechenleistung als mehr auf die schnelle Ausführung vom Befehlen hin ausgelegt war, einem Plattenlaufwerk, einer Tastatur, einer Maus sowie aus einem Grafikbildschirm, dem Herzstück des „Alto“. Alle Bestandteile waren so arrangiert, dass sie auf einem Schreibtisch Platz hatten. Da ein Bildschirm für Rastergrafiken sehr speicherintensiv ist, muss an dieser Stelle auch der Wechsel der technologischen Rahmenbedingungen mitbedacht werden. Integrierte Schaltkreise lösten Anfang der 70er die Magnetkernspeicher ab, die bis dahin neben Elektronenröhren als Datenspeicher dienten. Das wirkte sich sowohl auf die Größe des Speichers, wie auch auf den Preis aus. Im Gegensatz zu Engelbarts Labor konnte sich Xerox die Ausstattung der „Altos“ mit Rasterbildschirmen leisten.¹⁷³ Diese waren die Voraussetzung für die Darstellung des

¹⁶⁹Kay/Goldberg (1977): S. 394.

¹⁷⁰Ebda.

¹⁷¹Lampson (1986): S. 102.

¹⁷²Lampson, Butler (1972): „Why Alto“, <http://www.digibarn.com/friends/butler-lampson/> (10.11.2010)

¹⁷³1972 kostete der Speicher vom Alto noch 7000 \$, es wurde jedoch im Sinne des Moore'schen Gesetzes damit gerechnet, dass 128 Kbyte Speicher in wenigen Jahren nur noch 35 \$ kosten würden. Vgl.: Friedewald (1999): S. 271.

elektronischen Papiers mit elektronischer Tinte des Büros der Zukunft und konnten auch Kays Ansprüchen der Nachahmung bereits existierender Medien und Erschließung neuer Darstellungsformen gerecht werden. Die Tatsache, dass der Großteil der Prozessorleistung für die Verwaltung von Ein- und Ausgabe verwendet wurde - alleine die Ansteuerung des Grafikmonitors benötigte fast 75 % der Gesamtleistung¹⁷⁴ - lässt erkennen, dass der „Alto“ weniger als Rechenmaschine, sondern vielmehr als ein multimedialer Computer zur interaktiven Textverarbeitung, Echtzeitanimationen oder auch Musikerzeugung konstruiert wurde. Die Verbesserung der Kommunikation zwischen Benutzer und Computer erhielt oberste Priorität. Sobald die ersten „Altos“ 1974 funktionsfähig waren, begann sich die Dynamik des Bootstrapping zu entfalten. Das lokale „Alto“-Netzwerk war eine ideale Entwicklungsplattform für neue

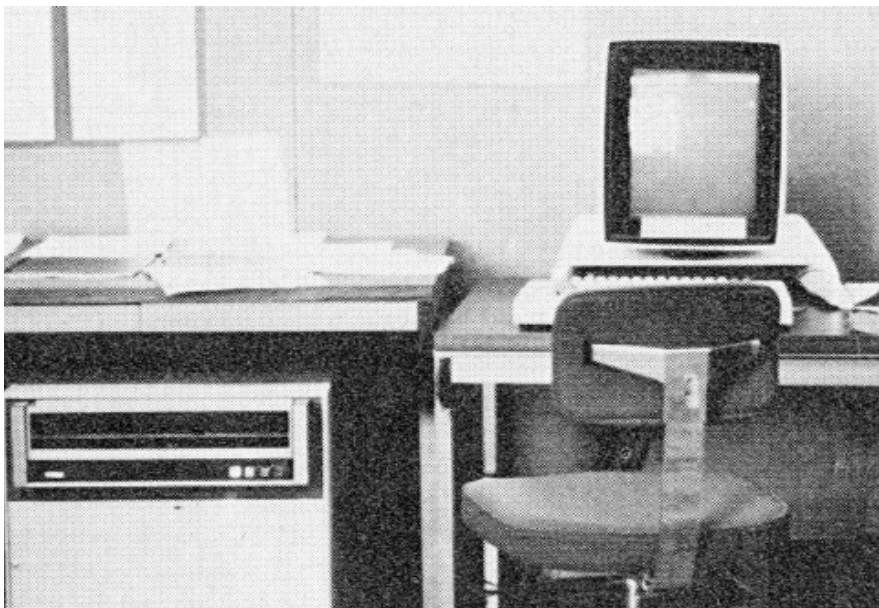


Abb.13: Der „Alto“ bzw. das „interim Dynabook“

Hard- und vor allem Software. Die wichtigsten Programme waren das E-Mail-Programm LAUREL, die Grafikprogramme MARKUP, DRAW sowie das Textverarbeitungsprogramm BRAVO. Die 72 dpi des Grafikmonitors erlaubten eine (für damalige Verhältnisse) akzeptable Qualität für die Entwicklung von verschiedenen Schriftarten. Mit BRAVO und dem ebenfalls am PARC entwickelten Laserdrucker, war es erstmals möglich ein Textdokument mit unterschiedlichen Schriftbildern und Layouts als einzelne Seite am Monitor darzustellen, die weitgehendst mit dem Ausdruck übereinstimmte. Dieses Konzept etablierte sich als das WYSIWYG-Paradigma (What you see is what you get).

¹⁷⁴Vgl.: Friedewald (1999): S. 265.

6.5.1. Das „interim Dynabook“ und „Smalltalk“

Für Kay und seine *Learning Research Group* (LRG) bedeutete der „Alto“ ein Schritt zur Verwirklichung der Vision des Dynabooks - konsequenterweise wurde er als „interim Dynabook“ bezeichnet. 1972 entwickelte Alan Kay gemeinsam mit Dan Ingalls die erste explizit objektorientierte Programmiersprache und Entwicklungsumgebung „Smalltalk“.¹⁷⁵ Um den menschlichen Problemlösungsstrategien so gut wie möglich zu entsprechen, wird in „Smalltalk“ alles als Objekt definiert: Nummern, Zeichenketten (etwa Textblöcke), Fenster und sogar gesamte Projekte. Objekte sind Instanzen übergeordneter Klassen, die das gemeinsame Verhalten der Objekte definieren. Darin werden alle zulässigen Botschaften deklariert, sowie alle Operationen, welche das Objekt ausführen kann. Objekte kommunizieren miteinander, indem sie Nachrichten senden und empfangen, wobei jedes Objekt einen eigenen Speicher besitzt, also die Fähigkeit hat, sich an seine Funktionen zu erinnern. Im Prinzip ist jedes Objekt ein kleiner Computer, der durch ein Netzwerk mit anderen Objekten kommunizieren kann. Dieser Ansatz bietet eine hohe Flexibilität und die Vermeidung starrer, unveränderbarer Strukturen. Das Prinzip der Objektorientierung hat somit eine starke Auswirkung auf die Kommunikationsmöglichkeiten mit dem Computersystem, denn „the essence of a medium is very much dependent on the way messages are embedded, changed and viewed.“¹⁷⁶ Nach eigener Schilderung erkannte Kay bereits 1966 wichtige Prinzipien der Datenverarbeitung, die er gar als Epiphanie beschreibt, zumal ihm klar wurde, dass es sich im Grunde um universale Prinzipien handelte. Als Biologiestudent erkannte er eine Parallele zur Funktion vielzelliger Organismen „with its notions of simple mechanisms controlling complex processes and one kind of building block able to differentiate into all needed building blocks.“¹⁷⁷ Außerdem erinnert das Konzept philosophisch an Leibniz Monaden- sowie an Platons Ideenlehre. Eine Klasse enthält sozusagen das abstrakte Urbild der Idee, deren konkrete Erscheinung die Objekte bilden. Für Kay war „Smalltalk“ jedoch weit mehr als nur eine Programmiersprache, nämlich ein Mittel des persönlichen Ausdrucks und nicht zuletzt sogar der megalomane Versuch „for a qualitativ shift in belief structures - a new Khunian paradigm in the same spirit as the invention of the printing press.“¹⁷⁸ Gemäß Kays Fixierung auf Kinder als die intuitivsten und unvoreingenommensten aller User wurde die Tauglichkeit von „Smalltalk“ anhand eben dieser „Referenzobjekte“ bemessen. Von 1974 bis 1976 testeten Kay und Goldberg daher „Small-

¹⁷⁵ „Smalltalk“ orientierte sich stark an Sutherlands Sketchpad und an SIMULA.

¹⁷⁶ Kay/Goldberg (1977): S. 393.

¹⁷⁷ Kay (1993a): S. 71.

¹⁷⁸ Ebda, S. 63.

talk”, nachdem es auf dem „Alto” implementiert wurde, an Schulen. „The particular aim of LRG was to find the equivalent of writing - that is learning and thinking by doing in a medium - our new `pocket universe‘.”¹⁷⁹ Angelehnt an LOGO war das Ziel der Arbeit, dass Kinder explorativ bereits entwickelte Modelle benutzen, diese durch ihre eigenen Ideen ergänzen bzw. in einen anderen Kontext einbetten können. Die Kinder erhielten z.B. ein illustriertes Büchlein namens „The Box Book”, in dem ihnen die Funktionen der Klasse Box näher gebracht wurden. Ein kurzer Ausschnitt soll dies verdeutlichen:

„Smalltalk is like a post office. The objects are like people who mail letters containing messages. The objects, or person, who receives the letter must read each message to find out what things to do. The name box is the name of a class of Smalltalk object that we have provided. If you type „box new name „joe”.!” you are telling Smalltalk to create joe, a new member of the class box. Because joe is now a member of the box class, anything a box can do, joe can do.”¹⁸⁰

Das Ausrufezeichen signalisierte in „Smalltalk” die Datenfreigabe - ”do it!”. Wurde etwa „joe grow 50.!” eingetippt, so vergrößerte sich das Quadrat am Bildschirm um 50 Einheiten. Mit „repeat (joe turn 5.)” begann sich das Quadrat gegen den Uhrzeigersinn zu drehen. Durch eine spielerische Heranführung an das Programmieren konnten die Kinder nach und nach Klassen erweitern und eigene Mal- und Animationsanwendungen entwickeln. Die LRG musste nach anfänglicher Euphorie jedoch feststellen, dass nur wenige wirklich in der Lage waren, von sich heraus kreative Fähigkeiten eines Programmierwurfs zu entwickeln. Nicht die Sprache selber schien das Problem zu sein, sondern eben doch die mangelnde Computerkompetenz bzw. die Fähigkeit, die repräsentierten Konzepte zu verstehen und sie umzusetzen. Die Hoffnung auf eine allgemeine „computer literacy” durch die Bereitstellung einfacher Programmierstrukturen schien noch in weiter Ferne - und wird es wohl auch noch länger bleiben.¹⁸¹ Doch die Experimente gewährten den Wissenschaftlern nichts desto trotz tiefgründige Einblicke in die Probleme der Mensch-Computer-Interaktion. Neben den Schulexperimenten der LRG wurde „Smalltalk” auch zur Implementierung des ersten modusfreien Texteditors (mini-MOUSE) und eines experimentellen Dokumentverwaltungssystems (FindIt) genutzt. Parallel zu diesen, eher konventionellen Anwendungen, entwickelten sich im Geiste von Kays Vision des Dynabooks auch die ersten Programme zur Aufzeichnung, Bearbeitung und Wiedergabe von Musik sowie das erste Animationsprogramm SHAZAM. Das gewichtigste Erbe von

¹⁷⁹Ebda, S. 84.

¹⁸⁰Goldberg, Adele/Kay, Alan (1977): „Methods for Teaching Smalltalk”, <http://www.bitsavers.org/pdf/xerox/parc/techReports/> (20.11.2010), S. 4f.

¹⁸¹Für nähere Ausführungen der Vorgehensweise und der Probleme siehe Goldberg/Kay (1977) bzw. Friedewald (1999): S. 319 ff. Kay hält nach wie vor an seiner Hoffnung fest. Er entwickelte beispielsweise Squeak, eine Open-Source-Implementierung von „Smalltalk”, die u.a. im Rahmen des „100-Dollar-Laptop”-Projekts läuft, für das Kay eine wichtige Triebkraft ist.

„Smalltalk“ ist jedoch die Entwicklung der ersten grafischen Benutzerschnittstelle, welche die Weichen für grundlegende Konzepte legte, die sich heute in jeder Benutzeroberfläche finden.

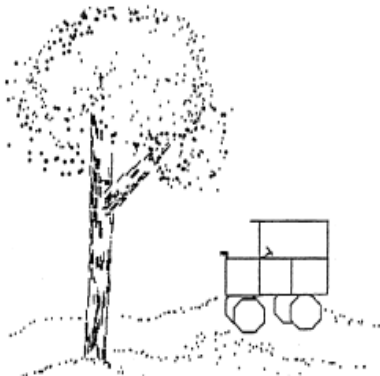
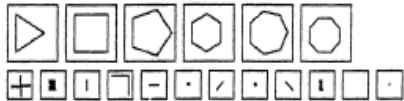


Abb.14: Zeichenprogramm der 13-jährigen Marian

Die Klasse Box wurde zu einer allgemeinen Klasse für Vielecke erweitert. Marian positionierte alle verfügbaren geometrischen Formen in Boxen am oberen Bildschirmrand. Mit dem Cursor konnten diese dann ausgewählt und am Bildschirm eingefügt werden - so als ob sie gestempelt werden.

6.5.2. Entwicklung überlappender Fenster und der Beginn der Benutzerillusion

„Smalltalk“ war am „Alto“ nur eine von vielen Anwendungsmöglichkeiten, jedoch die erste mit einer homogenen Benutzeroberfläche und sich überlappenden Fenstern. Die Einheitlichkeit aller „Smalltalk“-Programme, was somit auch ein konsistentes Interaktionsmodell festlegte, verdankte sich den Prinzipien der objekt-orientierten Programmierung und der Vererbung von Eigenschaften. „Vererbung bedeutet, dass eine Unterklasse die Eigenschaften und Methoden der übergeordneten Klasse übernimmt, zusätzlich aber weiter Eigenschaften und Methoden enthält.“¹⁸² Ein Bildschirmfenster etwa ist als Objekt der Klasse Rechteck primär ein rechteckiges Feld. Doch es ist zusätzlich durch die Bildschirmposition, Größe, Dicke der Randlinie und Füllung definiert. Außerdem können Fenster um diverse Schaltflächen oder etwa einen Rollbalken erweitert werden.¹⁸³ Auch das Kalendericon (Abb.15) gehört der Klasse Rechteck an, weist jedoch andere Charakteristika als Fenster auf. Bereits Sutherland und Engelbart verwendeten Fenster, um die räumliche Begrenzung des Bildschirms aufzuheben bzw. diesen zu strukturieren. Kay entwickelte, basierend auf der Analogie des Bildschirms mit ei-

¹⁸²Friedewald (1999), S. 322.

¹⁸³Bei „Smalltalk“ zeigte sich der Rollbalken erst, wenn man den Cursor über die linke Begrenzungslinie bewegte.

nem Schreibtisch, auf dem einzelne Dokumente abgelegt und übereinandergestapelt werden können, die Idee der „overlapping windows“. Im Gegensatz zu einem Blatt Papier kann der Inhalt eines Dokumentes jedoch weitaus größer sein als er im jeweiligen Fensterausschnitt

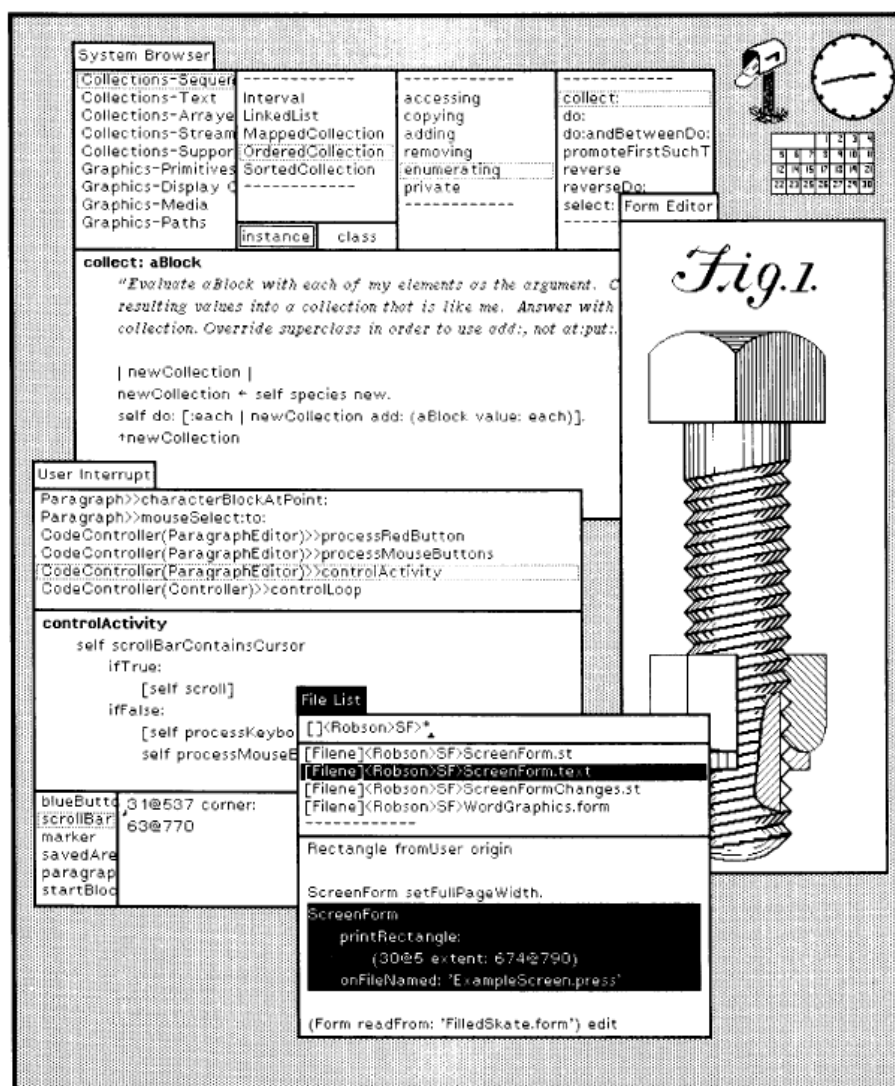


Abb.15: Benutzeroberfläche von „Smalltalk-76“ am Rasterbildschirm des „Alto“

erscheint. Vertikale und horizontale Rollbalken gewähren dementsprechend die Navigation innerhalb des Dokuments. Mit Referenz auf Bruner bemerkt Kay: „The flitting about nature of iconic mentality suggested that having as many resources showing on the screen as possible would be a good way to encourage creativity and problem solving and prevent blockage.”¹⁸⁴ Will der Benutzer zwischen Fenstern respektive Programmen wechseln, klickt er einfach auf das, in dem er arbeiten möchte. Dieses schiebt sich dann an die oberste Stelle des virtuellen Papierstapels. Die Benutzeroberfläche von „Smalltalk“ bot somit zum ersten Mal die

¹⁸⁴Kay (1990): S. 197.

Möglichkeit einer, scheinbaren, modusfreien Interaktion mit dem System. Ohne erst eine Anwendung beenden zu müssen, bevor die Arbeit mit einer anderen erfolgen kann, erlaubt diese Interaktionsstrategie schnell und einfach zwischen Projekten zu wechseln. Kay bemerkt dazu:

„The active window constituted a mode to be sure - one window might hold a painting kit, an other might hold text - but one could get to the next window to do something *without any special termination*. This is what *modeless* came to mean for me - the user could always get to the next thing desired without any backing out.”¹⁸⁵

Die Analogie von überlappenden Fenstern mit einem Papier- bzw. Dokumentstapel und die damit angedeutete Tiefe des virtuellen Handlungsraums markierte somit zum ersten Mal eine selbsterklärende Umgebung, in welcher der Benutzer Dokumente, wie an einem physischen Arbeitsplatz organisieren und intuitiv mit dem System arbeiten konnte. Kays elektronischer Papierstapel war der Anfang der sogenannten Desktop-Metapher. Allerdings ist es weniger eine exakte Nachahmung der realen Welt in einer virtuellen Umgebung, die das Erfolgsrezept der grafischen Benutzeroberfläche ausmacht, sondern die Tatsache, dass mit dem Computer Aufgaben bewältigt werden können, die eben keine Entsprechung an einem physischen Arbeitsplatz finden. Insofern sprachen die Entwickler auch nicht von einer Metapher, denn dies erschien zu einengend. Statt dessen war die Rede von einer Benutzerillusion:

„[W]hat is presented to one's senses *is* one's computer. The ‚user illusion‘ as my colleagues and I called it at Xerox Palo Alto Research Center, is the simplified myth everyone builds to explain [...] the system's action and what should be done next.”¹⁸⁶

Es ist eine Illusion insofern, als dass die Magie der Bilder die eigentliche Komplexität des Systems verdeckt, damit der Benutzer sich eine Vorstellung vom System machen kann. „There are clear connotations to the stage, theatric, and magic [...] it is the magic- the understandable magic - that counts.”¹⁸⁷ Ebenso wie im Theater sieht man nicht den mühsamen Probeprozess, den Schnürboden oder das ausgeklügelte Lichtsystem - muss man auch nicht - sondern lässt sich vom Spektakel verzaubern. Die Dichotomie zwischen dem codierten Hintergrund und der phänomenologischen Oberfläche, der Benutzerillusion also, ist paradigmatisch für die grafische Benutzeroberfläche. Was bereits als Charakteristikum des Computerbildes herausgestellt wurde, gerinnt nun zu einer Prämisse des Interface-Designs:

„Perhaps the most important principle is WYSISWG: the image on the screen is always a faithful representation of the user's illusion. Manipulating the image in a certain way immediately does something predictable to the state of the machine (as the user imagines the state).”¹⁸⁸

¹⁸⁵Ebda.

¹⁸⁶Kay (1984): S. 42.

¹⁸⁷Kay (1990): S. 199.

¹⁸⁸Kay (1984): S. 42.

Illusion und die Magie der Oberfläche bestimmen also seit Xerox PARC die Mensch-Computer-Interaktion. Nicht mehr der Benutzer habe sich an die Logik des Computers anzupassen, sondern die Erscheinung des Computers an den Benutzer. Und mit jedem Feedback auf seine Aktionen kann sich dessen mentales Modell von der Funktionsweise der Maschine verdichten bzw. durch Selbsterkundung erweitern. Dieses Paradigma wurde ab 1975, als mit der Konstruktion eines vermarktbaren „Office Automation System“, bekannt als Xerox Star, begonnen wurde, zur omnipräsenten Desktop-Metapher weiterentwickelt. Die grafische Benutzeroberfläche am „Alto“ basierte noch wesentlich auf Eklektizismus und auf keiner einheitlich fundierten Theorie. Erst beim Xerox Star erfolgte eine systematische und theoretische Fundierung der grafischen Benutzeroberfläche.

6.6. Xerox „Star“ - der Anfang der Automation

Die Optimierungsarbeit Engelbarts an den physiologischen Komponenten des Interfaces wurde beim Xerox „Star“ auf wahrnehmungspsychologischer Ebene weitergeführt. Doch ging es eben nicht mehr um die von Engelbart eingeforderte „Augmentation“ sondern eben um eine „Automation“, die der Xerox „Star“ ja schon programmatisch im offiziellen Namen trägt. Eine entscheidende Wende, die den theoretischen Beginn der Mensch-Computer-Interaktion als automatisierten Gestus markiert. Unter dem kognitionspsychologischen Einfluss von Allen Newell¹⁸⁹erfuhr v.a. die Psychologie des Computerbenutzers in den 70ern am PARC verstärkte Zuwendung, wobei sowohl theoretische Konzepte als auch empirische Testreihen entwickelt wurden, welche die Entwicklungsmethodik des „Star“ stark prägten. Bereits Kay erkannte die Wichtigkeit der Hinwendung zum Benutzer beim Systementwurf. War das Benutzermodell bei Kay noch sehr allgemein gehalten, wurde bei der Entwicklung des „Star“ User-Interfaces wurde von Anfang an genau deklariert „who the users are, what their goals are in performing the task, what information they use in performing it, what information they generate, and what methods they employ.“¹⁹⁰ Als prototypisch wurden gelegentliche Nutzer ohne Programmier- und Computerkenntnisse definiert, die vor allem daran interessiert sind, ihre Büroarbeit zu erledigen. Der „Star“ sollte also leicht erlernbar und die Funktionsweise leicht zu erinnern sein. Die Designprämisse lautete also „to make the ‚computer‘ as invisible to users as possible.“¹⁹¹

¹⁸⁹Der KI-Pionier Allen Newell war seit Anfang der 70er als Berater von Xerox PARC tätig; ab 1974 arbeitete er gemeinsam mit seinen ehemaligen Schülern Stuart Card und Tom Moran an einer systematischen Konzeption kognitionspsychologischer Grundlagen der Mensch-Computer-Interaktion. Vgl.: Card, Stuart et.al. (1983): *The Psychology of Human-Computer-Interaction*.

¹⁹⁰Smith, David Canfield et.al. (1982): „Designing the Star User Interface“, S. 264.

¹⁹¹Johnson et.al. (1989): S. 53.

Verschwinden sollten vor allem der intellektuelle und zeitliche Mehrwert, der größte Feind der Benutzerfreundlichkeit. Kays Konzeption „doing with images makes symbols“ legte bereits die theoretischen Grundlagen für die Gestaltung einer intuitiven Interaktion. Doch die Ziele waren andere: Kreativer Ausdruck und Lernen durch neugieriges Erkunden und Spielen mit dem System wurden im „Star“ eingetauscht gegen die Logik eines arbeitswissenschaftlichen Pragmatismus. Der Benutzer sollte so wenig wie möglich darüber nachdenken, wie er erreichen kann, was er will. Um diesem Ziel gerecht zu werden, wurde derselbe genauestens studiert und analysiert, welche Interaktionskonzepte als einfach bzw. schwer empfunden wurden: „Concrete vs. abstract, visible vs. invisible, copying vs. creating, choosing vs. filling in, recognizing vs. generating, editing vs. programming.“¹⁹² Alles, was einen intellektuellen und zeitlichen Mehrwert bedeutete, wurde systematisch aus dem Interfaceentwurf gestrichen. Direkte Manipulation konkret repräsentierter Daten fordert unmittelbare Reaktion des Systems und die Sichtbarkeit aller Effekte: „[T]he display becomes reality. Objects and actions can be understood purely in terms of their effects upon the display. This vastly simplifies understanding and reduces learning time.“¹⁹³

Zumal die Realität des Benutzers das Display war, erhielt die Gestaltung des Bildschirmlayouts höchste Priorität, was langjährige Testreihen mit klarerweise computerunkundigen Benutzern wie Sekretärinnen oder administrativem Xerox-Personal nach sich zog. Im Xerox „Star“ wurden fundamentale Gestaltungsprinzipien des grafischen User-Interfaces ausgearbeitet und zum ersten Mal standardisiert - z.B. die „illusion of manipulable objects“, i.e. der User sollte immer wissen mit welchen Objekten er wie interagieren kann und ob diese Objekte selektiert und bereit für weitere Aktionen sind. Ein Icon wurde etwa, sobald angeklickt, invertiert, um dem Benutzer ein sicheres Feedback auf seine Eingabe zu gewährleisten.¹⁹⁴ Auch wurden unter Berücksichtigung des Figur/Hintergrund Verhältnisses alle Fenster weiß ausgefüllt (im Vergleich zum grauen Desktop-Hintergrund), was außerdem die Analogie zum Papier unterstreichen sollte. Bei der Gestaltung der Benutzerillusion bezogen sich die Designer des Xerox „Star“ auf eine herkömmliche amerikanischen Büroumgebung. Die Metaphorisierung einer bekannten Arbeitsumgebung hat den Vorteil, dass die Benutzer auf ihr Alltagswissen zurückgreifen, und sich somit besser im System zurecht finden können. Die elektronischen Analogien zu Dokumenten, Ordnern und Druckern wurden dementsprechend als Reprä-

¹⁹²Smith et.al. (1982): S. 265.

¹⁹³David Canfield Smith zit. in. Johnson et. alt. (1989): S. 57.

¹⁹⁴Bei Mac OS X erscheint z.B. ein transparentes, gerahmtes Rechteck um anzuzeigen, dass das Objekt ausgewählt wurde. Außerdem indizieren bei Icons, welche Funktionen auslösen, Farben bzw. Grautöne, ob der Befehl ausgeführt werden kann oder nicht. Ist der Befehl nicht verfügbar, erscheint das Icon grau, ansonsten farbig.

sentationen konkreter Objekte - als Icons - realisiert, die mit der Maus ausgewählt und manipuliert werden konnten. „Seeing something and pointing to it is easier for people than remembering a name and typing it. This principle is often expressed in psychological literature as ‚recognition is generally easier than recall‘.”¹⁹⁵

Namensvater der Icons war David Canfield Smith. Noch bevor er Teil des Xerox „Star” Projekts wurde, verfasste er seine Doktorarbeit bei Alan Kay über ein „Smalltalk”-Programm namens Pygmalion - dem Vorläufer grafischer Entwicklungsumgebungen. In Pygmalion erfolgte die Kommunikation mit dem System ausschließlich „by means of visual entities called ‘icons’, subsuming the notions of variable, reference, data structure, function and picture.”¹⁹⁶ Icons sind nach Smith analoge Repräsentationsformen, d.h. sie besitzen, im Gegensatz zur Schrift, eine strukturelle Ähnlichkeit mit dem, was sie bedeuten. Daraus folgt, dass die Strukturen und Operationen mit diesen Analogien (also im metaphorischen Kontext) eine funktionelle Ähnlichkeit zu ihrem repräsentierten Kontext aufweisen.

„Analogical representations suggest operations to try, and it is likely that operations applied to analogical representations would be legal in the other context and vice versa. This is the philosophical basis for the design of Xerox Star, and ultimately, Apple Macintosh ‚desktop’ user interfaces [...] Being able to put documents in folders in a physical office suggests that one ought to be able to put document icons in folder icons in the computer ‚desktop’, and in fact one can.”¹⁹⁷

Grafische Symbole sind außerdem gegeneinander leichter abgrenzbar als Wörter, und somit einfacher anzusteuern und auszuwählen. Auch die Tatsache, dass Icons mehr Informationen auf einem geringeren Platz repräsentieren können, spricht angesichts des begrenzten Monitors für deren Einsatz. Vor allem aber stellt die semantische Nähe zu den durch die Icons repräsentierten Daten einen gravierenden Vorteil gegenüber textbasierten Interfaces dar. Bei der Entwicklung des „Star” wurden deshalb erstmals Grafikdesigner herangezogen um Vorschläge für Icons vorzulegen, die dann hinsichtlich ihrer Selbsterklärungsfähigkeit und Unterscheidbarkeit getestet wurden (Abb.16). Claus Pias bemerkt dazu treffend:

„Was wir heute als Papierkörbe, Ordner- und Festplatten-Icons auf Bildschirmen sehen, hat sich also erst einmal experimentalpsychologisch daran bemessen, was für SekretärInnen [...] die geringste Abweichung zur Wahrnehmung der Büro-Lebenswelt hat und daher am wenigsten Ap-
perzeptionsarbeit und Lohnkosten bedeutet.”¹⁹⁸

¹⁹⁵Bewley, William et.al. (1983): „Human Factors Testing in the Design of Xerox’s 8010 ‚Star’ Office Workstation”, <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.125.4373> (13.12.2010), S. 72.

¹⁹⁶Smith, David Canfield (1977): *Pygmalion. A Computer Program to model und stimulate creative Thought*, S. iv.

¹⁹⁷Smith, David Canfield (1993): „Pygmalion: An Executable Electronic Blackboard”, S. 35.

¹⁹⁸Pias, Claus (2000): „Computer Spiel Welten”, S. 78. (Das Papierkorb-Icon wurde erst bei Apples Lisa eingeführt. Das Löschen von Dokumenten erfolgte beim Xerox Star noch über die Funktionstasten am Keyboard.)

Der Xerox „Star“ wurde 1981 offiziell vorgestellt, war allerdings ein wirtschaftlicher Misserfolg, was u.a. am anfänglichen Wucherpreis von knapp 17 000 \$ lag. Zu dieser Zeit dominierten nämlich bereits der Apple II wie auch der IBM PC den Markt der (stets noch textbasierten) Einbenutzermaschinen, die mit 1000- 5000 \$ zwar weniger leistungsfähig dafür eben erschwinglicher waren. Xerox schaffte nie den einst erhofften Sprung in die kommerzielle Landschaft der Datenverarbeitung. Nichts desto trotz waren die Benutzeroberfläche von „Smalltalk“ und des Xerox „Star“ eine der wesentlichsten Inspirationsquellen für Firmen wie IBM und Apple, welche diese „Laborinnovationen“ schließlich in kommerziell erfolgreiche Produkte überführten.

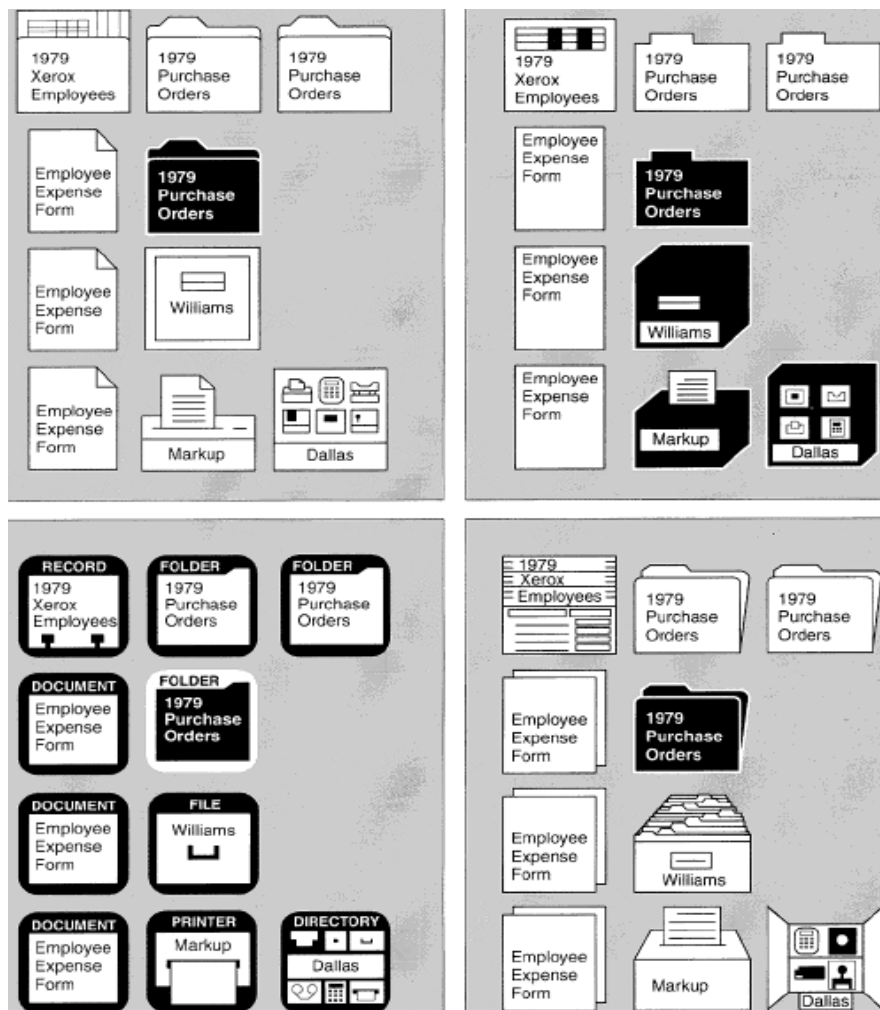


Abb.16: Icons Testset

links oben das Endergebnis; das schwarze Icon wurde selektiert und ist daher invertiert dargestellt; aufgrund des hohen Kontrastes nach der Selektion wurden alle Icons für den Repräsentationsmodus weiß konzipiert; der Test bestand aus 3 Teilen: Bewertung der Ähnlichkeit mit realen Objekten (da z.B. der Drucker öfters mit einer Wahlurne verglichen wurde, beschloss man, die Icons zu benennen); zwei Zeit-Testreihen, in denen hinsichtlich der Erkennung und Unterscheidbarkeit der Icons die Reaktionszeiten beim Auswählen gemessen wurde; zum Schluss wurden die Versuchspersonen nach ihrer subjektiven Meinung zur Gestaltung befragt.



Abb.17: Xerox „Star“ Workstation (1981)

6.7. Apple Lisa und Macintosh

Bereits 1976 entwickelten die zwei Computerenthusiasten Steve Jobs und Steve Wozniak den legendären Apple II, der die Weichen für Apples Aufstieg im Computergeschäft und zugleich auch für den Computer als Konsumgut legte.¹⁹⁹ 1978 begann die Entwicklung des Apple Lisa²⁰⁰, der die Schwächen des Apple II, v.a. dessen Benutzerunfreundlichkeit ausbügeln sollte.²⁰¹ Auch der LISA PC war für Büroanwendungen und somit unerfahrene Benutzer gedacht. Oberstes Gebot bei der Entwicklung war, dass die Arbeit mit dem Computer Spaß machen sollte:

„Lisa must be fun to use. It will not be a system that is used by someone ‚because it is part of the job‘ or ‚because the boss told them to.‘ For this reason, special attention must be paid to the friendliness of the user interaction and the subtleties that make using Lisa rewarding and the job enriching”²⁰²

Bei einem Besuch im Xerox PARC 1979 kam das Entwicklungsteam von Apple zum ersten Mal in Kontakt mit dem Konzept der grafischen Benutzeroberfläche. Insbesondere die modusfreie und mausfokussierte Schnittstelle von „Smalltalk“ begeisterte Jobs - zu Recht, denn

¹⁹⁹Vgl. Friedewald (1999): S. 371 f.

²⁰⁰Der Name stand offiziell für *Large Integrated Software Architecture*, inoffiziell wurde der Computer nach Jobs Tochter Lisa benannt.

²⁰¹In Apples *Training Department* wurde nämlich festgestellt, dass ca. 20-30 Stunden notwendig waren, um mit dem Apple II arbeiten zu können. Vgl. Friedewald (1999): S. 378.

²⁰²Perkins, Roderick et.al. (1997): „Inventing the Lisa User-Interface“, <http://portal.acm.org/citation.cfm?id=242405> (20.11.2010), S. 42.

sie war einzigartig in einer Zeit, als sich die Mensch-Computer-Interaktion noch großteils auf das Tippen von Kommandos in Form weißer oder grüner Buchstaben auf dunklem Hintergrund beschränkte. Inspiriert von „Smalltalk“ und geleitet „by the desire for consistency, ease of use, and efficiency“²⁰³, wurde ab 1980 der *Desktop Manager* - Lisas Benutzeroberfläche mit überlappenden Fenstern und Icons für Dokumente und Ordner - entwickelt. Lisa führte auch zum ersten Mal das Papierkorb-Icon ein. Durch das, ebenfalls neue Konzept, des »drag & drop« (Ziehen und Loslassen) konnten Dokumente bzw. Folder in den Papierkorb verschoben und somit gelöscht werden. Außerdem war es möglich, die Aktionen wieder rückgängig zu machen, was das Vertrauen des Benutzers ins System förderte. Anstelle der Pop-Up-Menüs von „Smalltalk“ wurde eine Befehlsleiste am oberen Bildschirmrand eingeführt.²⁰⁴ Durch Anklicken einer der Befehlskategorien, expandierten diese zu einem Pull-Down-Menü. Nachdem mit der Maus ein Bildschirmobjekt markiert wurde, konnte der Benutzer durch Öffnen des Menüs den entsprechenden Befehl auswählen. Das Objekt konnte solange manipuliert werden, wie die Maustaste gedrückt war. Beim Loslassen wurde der Befehl angewendet. Alternativ wurden auch Tastenkombinationen (keyboard shortcuts) angezeigt, die - besonders bei oft verwendeten Befehlen - ein zügigeres Arbeit erlaubten. Die Verwendung einer gemeinsamen Befehlsleiste bei allen Anwendungen hatte eine normierende Wirkung auf das Arbeiten mit dem System und erleichterte somit dessen Erlernen. Der Benutzer konnte sicher sein, dass sich unter FILE/PRINT, anwendungsunabhängig, stets die Befehle zum Laden, Speichern oder Drucken eines Dokuments befanden.²⁰⁵ „The interface was moving toward a standard that was called the Lisa ‚look and feel‘. All the applications would be similar in their appearance and use commands that would be common to each of them.“²⁰⁶ Um die Konsistenz der Anwendung zu gewährleisten wurde sogar ein 35-seitiger Leitfaden namens „Lisa User Interfaces Standard“ herausgegeben, welcher detaillierte Vorschriften zur Entwicklung von Anwenderprogrammen enthielt. Diese waren natürlich nur für computerliterate Entwickler gedacht, der Benutzer selbst konnte im Betriebssystem selbst nicht mal Programme öffnen, zumal es zugunsten der Benutzerfreundlichkeit, dokumentenzentriert gestaltet wurde. Der Benutzer sollte und durfte sich keine Gedanken darüber machen, welches Programm für welche Aufgabe

²⁰³Perkins et.al. (1997): S. 47.

²⁰⁴Diese räumliche Positionierung ist ein wunderbares Beispiel für ein benutzerfreundliches Interface-Design nach Fitt's Gesetz, zumal der Benutzer kein vertikales Feingefühl - damit auch weniger Zeit - braucht, um den Cursor über den gewünschten Menüpunkt zu positionieren.

²⁰⁵Die Konsistenz eines Systementwurfs ist ein wichtiges Gestaltungsprinzip bei Interfaces jeder Art. Steuerungsbefehle und allgemeine Funktionalitätsmodi des Systems müssen stetig sein, d.h. immer denselben Effekt hervorrufen. Ein einfaches Beispiel dafür sind die drei Buttons in der oberen Leiste von Fenstern, die unabhängig von der Anwendung und dem Betriebssystem die Befehle schließen, minimieren und maximieren bedeuten.

²⁰⁶Perkins et.al. (1997), S. 45.

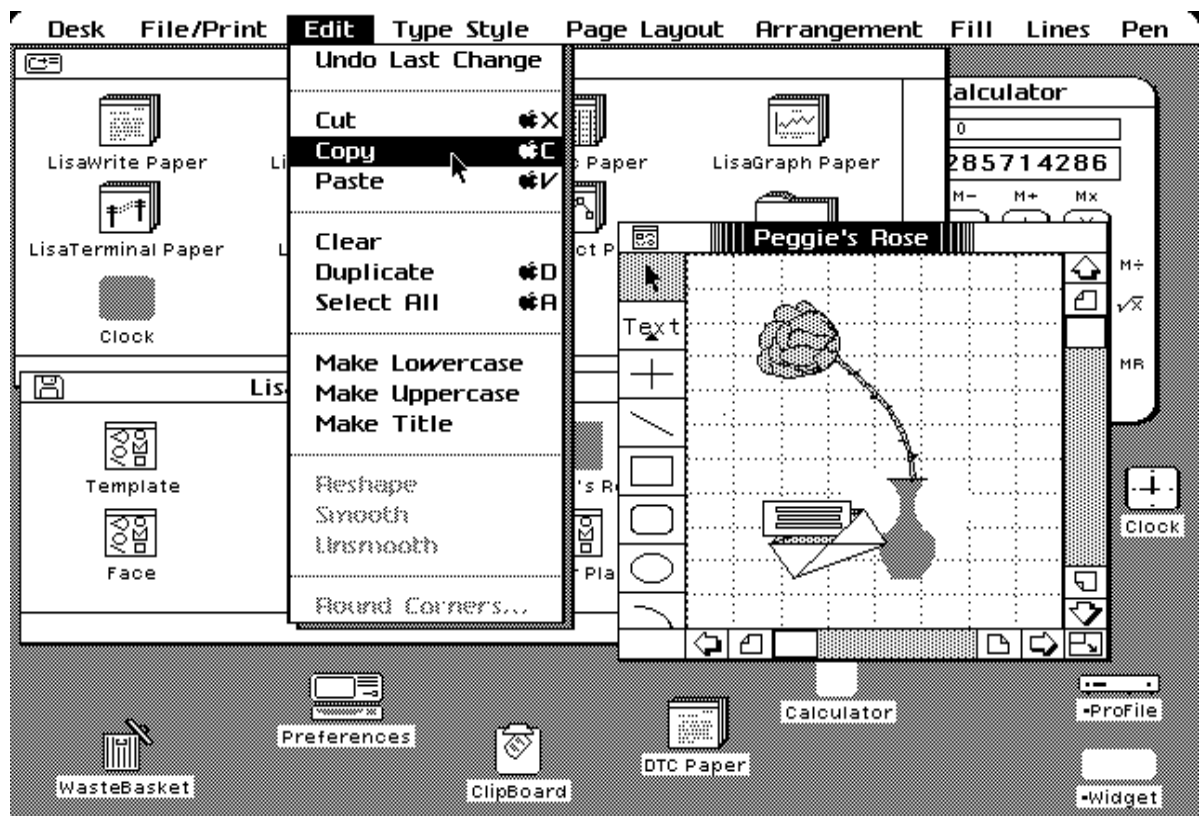


Abb.18: Desktop der Apple Lisa - Befehlsleiste mit Drop-Down-Menüs, Shortcuts, Papierkorb-Icon und überlappenden Fenstern

verwendet wurde. Wurde ein Dokument über einen Doppelklick auf das entsprechende Icon geöffnet, so erledigte der *Desktop Manager* alle Aufgaben hinter dem Rücken des Benutzers. Jedes Dokument war also mit einem bestimmten Programm verknüpft. In der *Desktop Library* waren sämtliche Routinen zur Verwaltung der Benutzeroberfläche gespeichert, die dafür zuständig waren, dass sich automatisch das richtige Programmfenster öffnete. Konsequenterweise hatten Anwendungen auch kein »Beenden« oder »Neues Dokument«. Ein neues Dokument wurde aus Vorlagen erstellt. Mit dem »Lisa Write Paper« (Abb.18. links oben) etwa konnte ein neues Textdokument erstellt und mit »LisaWrite« bearbeitet werden. Die Gestaltung des Interface war jedoch nicht nur auf die Bildschirmoberfläche begrenzt. Die Designer machten sich beispielsweise auch Gedanken über die äußere Gestaltung des Diskettenlaufwerks (sollte es offen sein oder eine Klappe haben) oder wie der Computer an- und ausgeschaltet werden kann. Bei Lisa war es z.B. möglich, den Computer jederzeit auszuschalten. Dieser speicherte den aktuellen Systemzustand und stellte ihn beim nächsten Hochfahren wieder automatisch her.²⁰⁷ Auch an der Maus wurden wesentliche Veränderungen vorgenommen.

²⁰⁷Die heutigen PCs erlauben das nur im beim Schlaf-Modus.

Die fehleranfällige mechanische Abtastung (Reibrad und Potentiometer) wurde durch eine optoelektronische Abtastung ersetzt (Bewegung einer Kugel wird von Sensor in elektrische Signale umgewandelt). Außerdem wurde die Anzahl der Tasten von drei auf eine einzige reduziert.²⁰⁸ Der Lisa Computer wurde so gestaltet, dass die gesamte Elektronik samt einem 12 Zoll Monitor in einem kompakten Kunststoffgehäuse untergebracht werden konnte.²⁰⁹ Mit sechs Anwenderprogrammen²¹⁰ und einer interaktiven Lernsoftware, die es beim Erwerb des



Abb.19: Apple Lisa

10 000 \$ teuren Gerätes (ohne Festplatte) gratis dazu gab, wurde Lisa 1983 der Öffentlichkeit präsentiert. Ergebnis: Flopp - zu teuer, zu langsam (die grafische Benutzeroberfläche verschlang zu viel Rechenleistung) und für Laien immer noch etwas zu kompliziert. Erst der 1984 auf den Markt gebrachte Macintosh verhalf Apple, dem PC sowie der grafischen Benutzeroberfläche zum massenweiten Durchbruch.

Jef Raskins war ab 1979 Leiter eines Forschungs- und Entwicklungsprojektes, das die Tradition von Kays Dynabook unter dem Einfluss der Volkscomputerbewegung weiterführen wollte. Der Computer, den Raskins nach seiner Lieblingsapfelsorte Macintosh taufte, sollte preiswert sein und weniger als 10 Kilo wiegen. Raskins wurde von Steve Jobs als Projektleiter ver-

²⁰⁸Tests ergaben, dass Benutzer, die mit einer Maus mit drei Tasten (wie beim „Alto“) arbeiteten, zu oft den Blick vom Bildschirm abwenden mussten, um sich zu vergewissern, welche Taste sie betätigten. Außerdem stellen mehrere Buttons einen erhöhten kognitiven Aufwand dar, zumal sich der Benutzer an die Funktion der einzelnen Tasten erinnern muss. Die Reduktion sollte vor allem den Lernaufwand für ungeübte Computerbenutzer schmälern. Vgl.: Perkins et.al. (1997): S. 46 f.

²⁰⁹Ein hochauflösender Ganzseitenmonitor wie beim Xerox Alto bzw. Star kam erstens aus Kostengründen und zweitens aus Platzgründen nicht in Frage.

²¹⁰Textverarbeitung, Zeichenprogramm, Tabellenkalkulation, Programm zur Erzeugung von Geschäftsgrafiken, Datenbank und Projektplanungsprogramm.

drängt, was zur Folge hatte, dass sich der Macintosh technologisch immer mehr der Lisa an-
glich. Das Problem war allerdings, dass die Produktionskosten des Macintosh so gering wie
möglich gehalten werden mussten. Auch der gewünschte Rasterbildschirm, dessen Auflösung
um dem WYSIWYG-Prinzip bei einer Größe von 9 Zoll zu entsprechen, auf 512 mal 342 er-
höht wurde, fraß viel an zur Verfügung stehender Speicher- und Rechenkapazität. Da den Ent-
wicklern bewusst war, dass der Erfolg des Computers v.a. von den verfügbaren End-User An-
wendungsprogrammen abhing, wurden diese in kompaktem Maschinencode geschrieben. Neu
waren kleine Schreibtischprogramme wie Rechner oder Kalender. Lisas Interface-Konzepte
wurden großteils übernommen, der *Desktop-Manager* wurde durch den *Finder* ersetzt. Icons
wurden verwendet, um Dokumente, Ordner, Programme und Laufwerke zu repräsentieren.
Um das OS so schlank wie möglich zu halten, wurden im ersten Design außerdem die Multi-
tasking-Fähigkeiten über Bord geworfen. Der Benutzer konnte also nicht mehr mit mehreren
Applikationen gleichzeitig arbeiten.²¹¹ Außerdem wurde zugunsten einer Platz- und Kostener-
sparsnis auf sonst übliche Erweiterungssteckplätze verzichtet. Der Macintosh war schließlich,
auch wenn er kein im Sinne des Dynabook portabler Computer war, ein sehr kompaktes Gerät
mit einer Stellfläche von 25 mal 25 cm und einem Gewicht von 10 kg. Entgegen der geringen
Produktionskosten von 500 \$ wurde er um 2495 \$ auf den Markt gebracht. Die Kommerziali-
sierung ging natürlich auch einher mit der geschickten Erschaffung eines Absatzmarktes, der
nicht zuletzt an die öffentliche Wahrnehmung des Status des Macintosh geknüpft war. Die
Markteinführung wurde von einer 15 Millionen \$ teurer Marketing-Kampagne, deren Höhe-
punkt der legendäre „1984“ Spot während des Super Bowl Endspiels war, begleitet.²¹² Apple
stärkte damit sein Image als Verfechter der kalifornischen Gegenkultur, der mit dem „compu-
ter for the rest of us“²¹³ die Unabhängigkeit der seelenlosen „Big Brother“-Konzerne à la
IBM, propagierte. Nach anfänglich guten Verkaufszahlen stagnierte v.a. aufgrund des unzurei-
chenden Speicherplatzes und fehlender Erweiterungsmöglichkeiten der Absatz. Die hardware-
technischen Kinderkrankheiten wurden erst 1986 mit dem Macintosh Plus beseitigt. Der lang-
fristige Erfolg verdankte sich jedoch vor allem den Softwareinnovationen, die Apple u.a. von
Microsoft und anderen Softwareunternehmen programmieren ließ.²¹⁴ Die „Killerapplikation“,

²¹¹Wollte man etwa Grafik in ein Textdokument übernehmen, so kopierte man die Grafik in eine Zwischenablage, beendete das Programm, öffnete den Texteditor und fügte die Grafik dann ein.

²¹²Für den 60-Sekunden-Spot, verfolgt von rund 96 Millionen Zuschauern, wurde sogar Ridley Scott engagiert. Der Produktionsetat war 900 000 \$, die Sendezeit kostete weitere 800 000 \$. Vgl.: Friedewald (1999): S. 400.

²¹³Ankündigung des Macintosh in der 16-seitigen, auffaltbaren Werbebroschüre im Newsweek Magazin 1984. Vgl.: Apple (1984): Newsweek Advertising Insert, online-Abdruck, <http://toastbucket.com/apple1984ad/> (20.11.2010)

²¹⁴Bereits 1982 erfolgte eine Übereinkunft mit Bill Gates, dass Apple Microsoft den ersten Prototypen des Ma-
cintosh zur Verfügung stellt. Im Gegenzug dazu verpflichtete sich Microsoft Anwenderprogramme für den Ma-



Abb.20: Der erste Apple Macintosh (1984)

also jene Anwendung wegen der sich ein potentieller Kunde für das Produkt entscheidet, kam jedoch aus Apples Labor.«PageMaker» war das erste Programm, mit dem sich einfach Texte und Grafiken kombinieren ließen. Gemeinsam mit dem Macintosh Office, welches die Vernetzung mehrerer Computer und die gemeinsame Nutzung eines Laserdruckers mit Hilfe eines Dateiservers ermöglichte, bildete sich das Konzept des Desktop-Publishing heraus. Das „Publizieren vom Schreibtisch aus“ bezeichnet das rechnergestützte Erstellen von anspruchsvollen Dokumenten, die für Printmedien bestimmt sind. Dieses Konzept prägte auch stark Apples Image, von dem sich die Firma noch immer nicht ganz befreien kann. Im Gegensatz zur „Killerapplikation“ des IBM PCs, der Tabellenkalkulation, avancierte der Mac somit zum Werkzeug der Kreativen, der Layouter und Grafiker.

Apple war Anfang der 80er Vorreiter in der Einführung benutzerfreundlicher PCs mit grafischer Benutzeroberfläche. Auch Microsoft brachte 1985 die erste Version von Windows heraus, welche jedoch erst 1990 auf den leistungstärkeren Intel 80386 Prozessoren ihre volle Wirkungskraft entfalten konnte.²¹⁵ Damit erhielt die grafische Benutzeroberfläche und die Desktop-Metapher neuen Aufschwung und ihren omnipräsenten Status. Doch es war eben auch zu jener Zeit, als die ersten Internet-Browser entwickelt wurden und der Siegeszug des World Wide Web begann. Ebenso wie eine, der Wahrnehmung des Benutzers angepasste Be-

cintosh zu entwickeln. Vgl.: Friedewald (1999): S. 403.

²¹⁵Vgl.: Friedewald (1999): S. 408 f.

nutzeroberfläche des Betriebssystems, sollen die auf verschiedenen Servern gespeicherten Informationen dem Benutzer über Hyperlink-Informationen zugänglich gemacht werden. Um die Komplexität der Informationen auf visueller Ebene zu verwalten, wurde das HTTP-Protokoll entwickelt. Jeder Internetbrowser erscheint seitdem als Fenster am Desktop, der somit nicht mehr nur virtueller Schreibtisch, sondern auch Portal zum größten Informationsnetzwerk unserer Kultur ist.

7. Interface-Kultur

Die kulturelle Relevanz des Interface gründet eben im offensichtlichen Paradoxon, dass virtuelle, hyperreale Räume einen zunehmenden Einfluss auf den heutigen Alltag ausüben, gleichwohl sich ihre Fundamente unserer Wahrnehmung entziehen. Die auf den Serverfarmen gelagerten Informationen befinden sich in einem abstrakten submedialen Raum, der erst durch eine visuelle Aufarbeitung an Form und Bedeutung gewinnt. Die Architektur der Information, wie sie sich einst schon Xerox auf die Fahnen haftete, nimmt heutzutage in ihrer Komplexität und Dringlichkeit ungeahnte Ausmaße an. Die Monumente unserer Kultur scheinen sich jedoch mehr und mehr in Bereichen der Unsichtbarkeit zu entwickeln. Welt- und Wertvorstellungen, soziale Strukturen und Machtkonfiguration gerinnen in technische Strukturen, die ihrer spezifischen Logik gehorchen - der Logik der digitalen Technologien, dem Verbund aus mathematischer Theorie und elektromagnetischen Steuerungsmechanismen. Es ist die Aufgabe der Programmierer, der System- und Interfacedesigner diese unsichtbaren Ordnungen zu konfigurieren, zu strukturieren und sie dem Benutzer über Interfaces zugänglich zu machen. Insofern verweist jede Gestaltung der Interfaces auf die Intentionen und Wertvorstellungen der Designer, die durch ihre „architektonischen“ Konzepte über die Art und Weise bestimmen, welche Informationen wie zugänglich gemacht werden. Das Interface ist also nicht nur Ausdruck einer Maschine-Mensch-Schnittstelle, welche die Logik der Informationsverarbeitung des Computers mit jener des Menschen kompatibel macht, sondern ein kulturelles Paradigma schlechthin. Denn als „Kulturtechnik verstanden bedeutet Interface die Organisationsfähigkeit innerhalb einer dynamisierten Mediensituation.“²¹⁶ Das Phänomen des Interface bringt somit die Notwendigkeit und Dringlichkeit, geeignete Strukturen von Informationsfiltern, -strukturierung und -organisation zu gestalten wortwörtlich an die (Benutzer)oberfläche. Gerade im Bereich des Interface-Design zeigt sich, dass einst noch getrennt gedachte Kategorie Technik und Kultur bzw. Technik und Kunst nahtlos ineinander fließen. Je mehr Kultur in digitale Formate der virtuellen Infosphären übersetzt und in ihnen verwaltet wird, umso mehr bahnt sich ein geeignetes Design, welches die Orientierung in diesen Räumen ermöglicht, als gestalterische Herausforderung des 21. Jahrhunderts an. „Es gibt im modernen Leben nur wenig schöpferische Taten von gleicher Bedeutung wie diese und nur wenige mit so umfassenden gesellschaftlichen Folgen.“²¹⁷ So Steve Johnson.

²¹⁶Hartmann (2006): S. 202.

²¹⁷Johnson (1999): S. 54.

Wirft man einen Blick zurück zum Anfang der Verbreitung des Computers als Massenmedium, so lässt sich diese Aussage bestätigen. Die Entwicklung der grafischen Benutzeroberfläche als Interfacekonvention war eine der folgenreichsten und durchaus auch kreativsten Leistungen des letzten Jahrhunderts. Erst die ikonografische Repräsentation von Befehlen und Speicherinhalten, die Metapher vom Computersystem als Schreibtisch, überlappende Fenster und das Konzept der direkten Manipulation bedingten die allgemeine Verfügbarkeit und leichte Erlernbarkeit komplexer, technischer Funktionen. Damit wurde das Terrain für den Computer als multimediales Medium für die Masse und als Kulturtechnik ausgelegt. Insofern drängt sich durchaus die Frage auf, ob nicht etwa die grafischen Schnittstellen die heimlichen Medien der digitalen Alltagskultur sind. Denn wird die Simulation unterschiedlicher Medienanwendungen nicht erst durch geregelte Übersetzungs- und Übertragungsaktivitäten zwischen technischer und phänomenaler Ebene gewährleistet? Und wurde letztere nicht erst massentauglich bzw. gar interessant durch die sogenannte „Desktop-WIMP“ Schnittstelle? Es ist eben jene mit Mausclicks auf grafische Elemente neu konfigurierbare Re-Präsentation und Kontrolle immaterieller Daten, welche die Magie und den Erfolg des Personal Computers ausmacht und den Computermonitor mit seinem ikonischen Antlitz zur epistemischen Instanz des postmodernen Lebens werden lässt.

8. Resumée

Den Nährboden dieser Arbeit bildete die Feststellung, dass heutige Ordnungen des Wissens, der Erfahrung sowie der Kommunikation entscheidend durch die Logik und Performanz digitaler Technologien bestimmt werden und geprägt sind. Vor allem seit der großflächigen Etablierung des World Wide Web avancierte die Mensch-Computer-Interaktion zu einem soziokulturell höchst relevanten und omnipräsenten Phänomen. Interfaces, die Übersetzungsinstanzen zwischen den abstrakten submedialen Vorgängen des digitalen Codes von Computern und der menschlichen Perzeption, ermöglichen die Interaktion mit dem System sowohl in technischer wie auch in kognitiver Hinsicht und bilden somit eine Schlüsselkategorie der heutigen Medienlandschaft. Die Entwicklung der grafischen Benutzeroberfläche vom Format „Desktop-WIMP“ - die dominante Ausprägung der Mensch-Computer-Interaktion - erweist sich retrospektiv gesehen als ein entscheidender Triebfaktor, wenn nicht gar, so die Behauptung, als das zentrale Moment der Entwicklung der heutigen, auf den Fundamenten der digitalen Technologie basierenden, Medienkultur, indem sie eine allgemeine Verfügbarkeit komplexer technischer Funktionen gewährleistete. Sie zeichnet sich durch die Ikonografisierung von virtuellen Speicherinhalten und Befehlsstrukturen sowie durch eine Konsistenz auf visueller Ebene und jener der Benutzeroperationen aus. Überlappende Fenster, Icons, kontextsensitive Pop-Up Menüs und Menüleisten, die über die Maus bzw. ein alternatives grafisches Eingabegerät angesteuert und manipuliert werden können, bilden das visuelle Arrangement der Bildschirmfläche. Diese verdichtet sich zum Zweck der Benutzerillusion zu einem virtuellen Schreibtisch, dem Desktop, der dazu dient, dem Benutzer eine Vorstellung von der Funktionsweise des Systems zu geben und sich im aufgespannten Handlungsraum zu orientieren. Das Prinzip der direkten Manipulation von grafisch visualisierten Daten erzeugt das Gefühl bzw. die Illusion einer direkten Verfügbarkeit über dieselben und verbindet die Eigenbewegung der Hand mit der visuellen Ebene des Bildschirms. Dies bedingt im Gegensatz zu kommandozeilenbasierten Interfaces eine leichtere, intuitivere Erlernbarkeit des Systems und reduziert den kognitiven wie zeitlichen Mehrwert des Benutzers. Die unmittelbare Sichtbarkeit des Effekts der Aktion des Benutzers ist unabdingbar für die Mensch-Computer-Interaktion, denn sie gibt Aufschluss darüber, was als nächstes gemacht werden kann. Jede Interaktion gründet also auf einem instantanen Feedback des Systems, das sich in der Veränderung der Oberfläche äußert. Die Variabilität digital gespeicherter Information, welche die materielle Bindung von Informationen hinter sich lässt, ist somit eine wesentliche Voraussetzung des Mensch-Computer-

Dialogs. Das Bild des Monitors ist aus einzelnen Pixeln aufgebaut, die im Speicher abgelegt sind und somit direkt adressiert werden können. Die Verbindung zwischen Speicherinhalt und Bildschirmfläche bewirkt die doppelte Existenz des Computermonitors als Repräsentations- wie auch als Kontrollfläche. Die Bildfläche ist zugleich Benutzeroberfläche, der „haptische Blick“ die technikfunktionale Geste unseres Zeitalters.

Ausgehend von diesen Feststellungen und Überlegungen, wandte sich diese Arbeit der Untersuchung der Entwicklung der grafischen Benutzeroberfläche zu. Diese stand in engem Zusammenhang mit einem Wechsel von Ansichten zur Funktion und Nutzung des Computers. An diesen Umwertungsprozessen waren eine Vielzahl unterschiedlicher Akteure, z.B. ideengebende Persönlichkeiten, finanz- und einflussstarke Institutionen bzw. Unternehmen, politische und kulturelle Strömungen, technologische Entwicklungen und jeweils vorherrschende Ideologien beteiligt, was daher eine Perspektivierung der Geschichte der Mensch-Computer-Interaktion erforderlich machte. Es wäre natürlich noch ausführlicher zu den historischen Diskontinuitäten und Brüchen zu forschen, was jedoch den Rahmen dieser Arbeit sprengen würde. Nichts desto trotz wurde versucht zu zeigen, dass ein gemeinsames Fundament der Entwicklung der Mensch-Computer-Interaktion die Idee war, den Computer nicht primär als eine automatische Rechenmaschine, die getrennt vom Benutzer arbeitet, zu betrachten, sondern an seiner Möglichkeit anzusetzen, den Menschen in seinem Denken zu unterstützen und seine intellektuellen und kreativen Fähigkeiten zu fördern. Programmatisch dafür stehen Vannevar Bushs Memex-Vision, die von Joseph Licklider geforderte „Man-Computer-Symbiosis“, Douglas Engelbarts Streben nach der Augmentierung des menschlichen Intellekts durch digitale Technologien sowie Alan Kays Dynabook-Vision als persönliches (Meta)Medium zum kreativen Ausdruck. Die grafische Benutzeroberfläche als vorläufiges Ende dieser Entwicklung kann theoretisch als eine Materialisierung bzw. Versinnlichung eines Arbeitsplatzes des Denkens begriffen werden und das Konglomerat virtueller Werkzeuge als „übertragene Zeigefinger des Geistes.“²¹⁸ Ob die heute dominante Form der Mensch-Computer-Interaktion auch in der Praxis den Ansprüchen bzw. Visionen der in dieser Arbeit erwähnten Pioniere der Mensch-Computer-Interaktion gerecht wird, sei dahingestellt.

Vannevar Bushs 1945 fiktiv formuliertes Wissensarchivierungs und -organisationsgerät, der Memex, bildete den theoretischen Ursprung des Konzept einer Mensch-Computer-Interaktion

²¹⁸Reifenrath (2000): S. 13.

zur Unterstützung intellektueller Fähigkeiten und den Ausgangspunkt der Notation eines persönlichen und intuitiv zu bedienenden Arbeitsplatzes.

Die Wurzeln der ersten tatsächlichen Ausprägung einer Mensch-Computer-Interaktion reichen zurück in militärisches Terrain, zu SAGE, wo sich bereits wesentliche Elemente der Hardware-Interfaces zur Benützung des Computers manifestierten und insbesondere das interaktive Arbeiten mittel grafischer Eingabegeräte am Bildschirm vorweggenommen wurde.

Abseits des militärische Einsatzes erfolgte die Benützung von Computern als „Batch-Processing“, was keine direkte, interaktive Zusammenarbeit mit den Rechnergiganten erlaubte. Vor diesem Hintergrund entstanden Ende der 50er Jahre vermehrt Forderungen nach der Entwicklung interaktiver Systeme, mit denen zugunsten eines Problemlösungsprozesses in Echtzeit, wie es SAGE demonstrierte, gearbeitet werden konnte.

Joseph Licklider skizzierte in seinem wegweisenden Aufsatz „Man-Computer-Symbiosis“ (1960) die intellektuelle Fruchtbarkeit des Zusammenspiels zwischen Mensch und Computer sowie die Dringlichkeit eines dialogischen Austausches, was systematische Forschungen nach neuen Interaktionsformen nach sich zog. Das Time-Sharing als vorübergehende Lösung interaktiven Arbeitens mit dem Computer durch Aufteilen der teuren Rechenressourcen auf mehrere Benutzer erlaubte erstmals eine persönliche Echtzeit-Kommunikation mit dem Rechner über Terminals und bot ab den 60er Jahren bis zum Advent des PCs die Plattform der Entwicklung innovativer Nutzungskonzepte.

Ein fundamentaler Beitrag war Ivan Sutherlands Sketchpad-System, welches bereits die direkte Manipulation von grafischen Bildschirmobjekten mittels eines Lichtgriffels, dem Vorgänger der Maus, sowie das Konzept der Fenster als Öffnung und Erweiterung des begrenzten Bildraums vorwegnahm. Schriftliche Kommandos wurden durch ikonische Interaktion ersetzt, vorrangiger Kommunikationsmodus waren Zeichnungen und Skizzen, „Sketches“, die als konkrete Entitäten abstrakte Programmstrukturen repräsentierten und über grafische Eingabe und Kommandoknöpfe verändert werden konnten.

Douglas Engelbart knüpfte an die Visionen Vannevar Bushs an und setzte sich zum Ziel, mit Hilfe des Computers die menschlichen Fähigkeiten zum Umgang mit immer komplexer werdender Problemen zu verbessern. Vor allem in dessen Fähigkeiten einer dynamischen Symbolrepräsentation, welche über die Statik analoger Medien hinausreicht, sah er die Möglichkeit der Emergenz neuer Denkräume. Konsequenterweise nahm die Ausgestaltung einer interaktiven Kommunikation über den Bildschirm oberste Priorität ein. Als Radartechniker wusste er, dass ein Computermonitor Text anzeigen kann und auch soll, obwohl die Verwendung von

Computermonitoren Anfang der 60er Jahre bis auf einige Ausnahmen noch nicht üblich war. Seine Forschungen brachten neben den ersten Hyperlinks, einer interaktiven Textverbreitung, „tiled windows“ und Videokonferenzen als bekanntestes Erbe die Maus hervor, die sich gegenüber bereits bestehenden Möglichkeiten einer grafischen Eingabe als effizientestes Eingabegerät erwies.

Ende der 60er ließen technologische Fortschritte die tatsächliche Realisierung des PCs immer mehr in Sichtweite rücken. Im Xerox PARC vermischten sich zwei Forschungstendenzen, entlang derer das grafische User-Interface seine Gestalt annahm. Einerseits die Entwicklung eines papierlosen Büros für die Zukunft, andererseits der Realisierungsversuch des von Alan Kay formulierten Konzepts des Dynabooks als persönliches, dynamisches Meta-Medium. Beide Systementwürfe forderten eine intuitive und leicht erlernbare Bedienung, was mit der Hinwendung zu Wahrnehmungsmechanismen des Benutzers einherging, um diese in die Gestaltung des Interfaces auszulagern. Alan Kay verschrieb seine Forschungen der Entwicklung eines Systems, was kinderleicht und von Kinder selbst zu bedienen sein sollte. Nach der Bekanntschaft mit Seymour Papert und LOGO, eine Programmiersprache und Entwicklungsumgebung für Kinder, wurde Kay klar, dass es sich beim Interface eines persönlichen Mediums stets um eine Lernumgebung handeln müsse, die dem Benutzer durch selbstständiges, intuitives Erkunden einen Zugang zu den Funktionsweisen des Systems verschaffen sollte. In dem Slogan „doing with images makes symbols“ brachte Kay seine Prämisse des Interface-Design auf den Punkt, die mit der Maus manipulierbare Icons und Menüs als grafische Repräsentation sonst abstrakter Konzepte von Algorithmen und Verzeichnissen auf den Bildschirm zauberte. Es war auch Kay, der im Rahmen von „Smalltalk“ das erste Mal das Prinzip überlappender Fenster als Analogie zu Schreibtischdokumenten vorschlug und in Folge dessen das Konzept der Benutzerillusion formulierte, was zur omnipräsenten Desktop-Metapher führte. Lag der Fokus von Kay vorwiegend auf dem Potential des Computers als kreatives Ausdrucksmedium und seinen didaktischen Möglichkeiten, die er mit dem „Alto“ - dem ersten PC mit grafischer Benutzeroberfläche - erkundete, so richtete das Management von Xerox PARC das Hauptaugenmerk auf Büroangelegenheiten. Diese Ambition verdichtete sich im Xerox „Star“, eines „Office Automation System“. Basierend auf kognitionspsychologischen Forschungen wurde das User-Interface immer mehr an die Bedürfnisse von Computerlaien angepasst und schlussendlich nicht nur Bürotätigkeiten sondern auch die Mensch-Computer-Interaktion automatisiert. Die Forschungen im Xerox PARC der 70er Jahre legten somit das Fundament des benutzerfreundlichen, multimedialen PCs.

Es war schließlich Apples Verdienst den PC mit der grafischen Benutzeroberfläche von seinem Laborstatus zu lösen und in ein kommerziell erfolgreiches Produkt zu überführen, das von jedem „kinderleicht“ bedient werden kann. Apple griff die Entwicklungen von Xerox PARC auf und erweiterte diese um Elemente wie „drag und drop“, das Papierkorb-Icon sowie die Menüleiste mit Pull-Down Menüs. Microsoft Windows und andere Nachahmer folgten schnell. Seitdem hielt der Personal Computer Einzug in die Wohnzimmer, heutzutage angesichts der Dominanz von Laptops, in die Taschen der Menschen. Er wurde zum Spiel- und Arbeitsgerät und seit den 90ern, mit dem Aufkommen von Webbrowsern, zu einem unverzichtbaren Bestandteil der digitalen Medienkultur, unserer Handlungs-, Wissens- und Kommunikationsräume.

Die Entwicklung des Interface ist jedoch noch lange nicht abgeschlossen, obwohl sich das „Desktop-WIMP“-Gefüge als Anfangs- und Endpunkt aller Praktiken rund um den PC hartnäckig zu halten scheint. Der zunehmende Einfluss virtueller Infosphären auf unseren Alltag verweist umso vehementer auf die Relevanz des Interface-Design als eine technische und gestalterische Herausforderung unseres Zeitalters und belegt somit die kulturstiftende Kraft des Interface - einst wie zukünftig.

„Unsere Interfaces sind Geschichten, die wir uns selbst erzählen, um die Sinnlosigkeit abzuwehren, Gedächtnispaläste aus Silizium und Licht. Sie werden weiterhin die Art und Weise verändern, wie wir uns Information vorstellen, und damit zwangsläufig auch uns verändern - im Guten wie im Bösen. Wie könnte es anders sein?“²¹⁹

²¹⁹Johnson (1999): S. 271.

9. Quellenverzeichnis

Apple (1984): Newsweek Advertising Insert, online-Abdruck, <http://toastbucket.com/apple1984ad/> (20.11.2010)

Bardini, Thierry (1997): „Bridging the Gulfs. From Hypertext to Cyberspace”, In: *Journal of Computer-mediated Communication*, 3 (2) September 1997, <http://jcmc.indiana.edu/vol3/issue2/bardini.html#rref36> (08.11.2010)

Bewley, William L. et.al. (1983): „Human Factors Testing in the Design of Xerox’s 8010 ‚Star‘ Office Workstation”, In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Boston, Dezember 1983, S. 72-77. CiteSeer-Scientific Literature Digital Library and Search Engine, <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.125.4373> (13.12.2010)

Bruner, Jerome (1966): *Toward a Theory of Instruction*. Cambridge, Mass.: The Belknap Pr. of Harvard University Press.

Bush, Vannevar (1945): „As we may think” , In: Wardrip-Fruin, Noah/ Montfort, Nick (Hg.): *The New Media Reader*. Cambridge, Mass.: MIT-Press 2003, S. 37- 49.

Card, Stuart/ Moran, Tom/ Newell Allan (1983): *The Psychology of Human-Computer Interaction*. Hillsdale: Erlbaum.

Carr, J.W. (1962): „Better Computers”, In: *Electronic Computers* 4 (4), S. 157-160.

Corbato, Fernando J. (1963): „1963 Timesharing: A Solution to Computer Bottlenecks”, Interview von John Fitch mit Fernando J. Corbato am MIT Computation Center am 9.05.1963, online-Video, <http://www.youtube.com/watch?v=Q07PhW5sCEk> (20.11.2010)

Coy, Wolfgang (2008): „Auf dem Weg zum ‚Finalen Interface‘”, In: Hellige, Hans-Dieter (Hg.): *Mensch-Computer-Interface: Zur Geschichte und Zukunft der Computerbedienung*, Bielefeld: transcript 2008, S. 309-321.

Dinkla, Söke (1997): *Pioniere Interaktiver Kunst von 1970 bis heute*. Ostfildern: Cantz Verlag.

Engelbart, Douglas (1961): „Program On Human Effectiveness” (Typoscript). Stanford Research Institute. Menlo Parc, Kalifornien Dezember 1961. Stanford Digital Archive, <http://sloan.stanford.edu/mousesite/Archive/Post68/PrHumanEffectiveness.html#1>. (23.11.2010)

Engelbart, Douglas (1962): „Augmenting Human Intellect: A conceptual Framework”, SRI Summary Report AFOSR-3223 (Prepared for: Director of Information Sciences, Air Force Office of Scientific Research, SRI Project No. 3578). Doug Engelbart Institute-Library, <http://www.doungelbart.org/pubs/augment-3906.html> (10.11.2010)

Engelbart, Douglas (1973a): „Design Considerations for Knowledge Workshop Terminals” , In: *Proceedings of the AFIPS National Computer Conference*, Nr. 42, S. 221-227. Doug En-

gelbart Institute-Library, <http://www.doungelbart.org/pubs/augment-14851.html>
(23.11.2010)

Engelbart, Douglas/Watson R.W./Norton J.C. (1973b): „The Augmented Knowledge Workshop”, In: *Proceedings of the AFIPS 1973 National Computer Conference*, Nr. 42, S. 9-21. Doug Engelbart Institute-Library, <http://www.doungelbart.org/pubs/augment-14724.html>
(23.11.2010)

Engelbart, Douglas (1986/87): Interview geführt von Judith Adams und Henry Loowood am 19. Dezember 1986, 14. Jänner 1987, 4. März 1987 und 1. April 1987 für das Stanford „Oral History Project”. Transcript von Kassettenaufzeichnung, Stanford University Archives, <http://www-sul.stanford.edu/depts/hasrg/histsci/ssvoral/engelbart/engfmst1-ntb.html>
(23.11.2010)

Engelbart, Douglas (1995a): „Toward Augmenting the Human Intellect and Boosting our Collective IQ”, In: Ders.: *Boosting our Collective IQ. A Selection of Readings*, S. 3-7. Doug Engelbart Institute-Library, <http://www.doungelbart.org/pubs/augment-133150.html>
(23.11.2010)

Engelbart, Douglas (1995b): „Dreaming of the Future”, In: *Byte.com*, <http://web.archive.org/web/20080111155845/http://byte.com/art/9509/sec15/art1.html>
(23.11.2010)

English, William/Engelbart, Douglas/Berman, Melvin (1967): „Display-Selection-Techniques for Text Manipulation”, In: *IEEE Transactions on Human Factors in Electronics*, 8 (1), S. 5-15. Doug Engelbart Institute-Library, <http://www.doungelbart.org/pubs/augment-133184.html> (13.11.2010)

Friedewald, Michael (1999): *Der Computer als Werkzeug und Medium. Die geistigen und technischen Wurzeln des Personal Computers*. Berlin: Diepholz.

Goldberg, Adele/Kay, Alan (1977): „Methods for Teaching Smalltalk”, Report SSL-77-2, Xerox Palo Alto Research Institute, Xerox Bitsavers (Online-Archiv), <http://www.bitsavers.org/pdf/xerox/parc/techReports/> (20.11.2010)

Hadamard, Jacques (1945): *The Psychology of Invention in the Mathematical Field*. New York: Dover Publications.

Halbach, Wulf R. (1994): *Interfaces. Medien- und kommunikationstheoretische Elemente einer Interface-Theorie*. München: Fink.

Hartmann, Frank (2006): *Globale Medienkultur: Technik, Geschichte, Theorien*. Wien: WUV-Univ-Verlag.

Heim, Michael (1993): *The Metaphysics of Virtual Reality*. New York et.al.: Oxford Univ. Press.

Hellige, Hans-Dieter (2008): „Krisen- und Innovationsphasen in der Mensch-Computer-Interaktion“, In: Ders. (Hg.): *Mensch-Computer-Interaktion. Zur Geschichte und Zukunft der Computerbedienung*. Bielefeld: transcript 2008, S. 11-92.

Johnson, Jeff et.al (1989): „The Xerox ‚Star‘: A Retrospective“, In: Baecker, Ronald M.et.al. (Hg.): *Readings in Human-Computer Interaction: Toward the year 2000*. 2nd edition. San Francisco: Kaufman 1995, S. 53- 70.

Johnson, Steven (1999): *Interface-Culture. Wie neue Technologien Kreativität und Kommunikation verändern*. Stuttgart: Klett-Cotta.

Kay, Alan (1969): „The Reactive Engine“, Dissertation. University of Utah, Salt Lake City, online-Exzerpt zur Verfügung gestellt von Matthias Müller-Prove, <http://www.mprove.de/diplom/gui/kay69.html> (23.11.2010)

Kay, Alan (1972): „A Personal Computer for Children of all Ages“, In: *Proceedings of the ACM National Computer Conference*, August 1972, online-Volltext zur Verfügung gestellt von Matthias Müller-Prove, <http://www.mprove.de/diplom/referencesKay.html> (13.11.2010)

Kay, Alan /Goldberg, Adele (1977): „Personal Dynamic Media“, In: Wardrip-Fruin, Noah/Montfort, Nick (Hg.): *The New Nedia Reader*. Cambridge: MIT-Press 2003, S. 393-404.

Kay, Alan (1984): „Computer Software“, In: *Scientific American*, 251 (3), S. 41-47.

Kay, Alan (1993a): „The Early History of Smalltalk“, In: *ACM SIGPLAN Notices*, 28 (3), S. 69-95. ACM-Computer Digital Library, <http://portal.acm.org/citation.cfm?id=155364> (07.11.2010)

Kay, Alan (1993b): Vorwort in Cypher, Allen (Hg.): *Watch What I Do: Programming by Demonstration*. Cambridge: MIT Press 1993, S. xi -xvii.

Kay, Alan (1990): „User Interface. A Personal View“, In: Laurel, Brenda (Hg.): *The Art of Human-Computer-Interface Design*. Reading, Mass.: Addison-Wesley 1990, S. 191-207.

Kay, Alan (2007): „The PC must be revamped- Now“, Interview mit Allan Alter, CIO-Insight Expert Voices, <http://www.cioinsight.com/c/a/Expert-Voices/Alan-Kay-The-PC-Must-Be-Revamped151Now/> (20.11.2010)

Kittler, Friedrich (1998): „Computergraphik. Eine halbtechnische Einführung“, Vortrag in Basel, In: Häring, Brigitte et.al. (Hg.): *Buchstaben, Bilder, Bytes. Das Projekt der Wahrnehmung*. Norderstedt: Books on Demand 2004, S. 47-63.

Lampson, Butler (1972): „Why Alto.“, Xerox Inter-Office Memorandum an CSL. Digibarn Computer Museum, <http://www.digibarn.com/friends/butler-lampson/> (10.11.2010)

Lampson, Butler (1986): „Personal Distributed Computing: The Alto and Ethernet Software“, In: *Proceedings of the ACM Conference on „The History of personal Workstations“*, S. 101-131. ACM-Computer Digital Library, <http://portal.acm.org/citation.cfm?id=12186> (10.11.2010)

- Laurel, Brenda (1991): *Computers as Theatre*. Reading, Mass. et.al.: Addison-Wesley.
- Licklider, Joseph (1960): „Man-Computer Symbiosis“, In: Wardrip-Fruin, Noah/Montfort, Nick (Hg.): *The New Media Reader*. Cambridge, Mass.: MIT-Press 2003, S. 74-82.
- Licklider, Joseph/Clark, W.E. (1962): „On-Line Man-Computer Communication“, In: *Proceedings of the AFIPS Spring Joint Computer Conference*, Mai 1962, S. 113-128. ACM-Digital Library, <http://portal.acm.org/citation.cfm?id=1460847> (20.11.2010)
- Licklider, Josphe (1965): *Libraries of the Future*. Cambridge, Mass.: MIT Press.
- Manovich, Lev (2001): *The Language of New Media*. Cambridge et.al.: MIT-Press.
- McLuhan, Marschall (1964): *Understanding Media. The Extensions of Men*. New York: McGraw-Hill.
- Müller-Prove, Matthias (2008): „Vom Persönlichen Computer zum Sozialen Medium“, In: Hellige, Hans-Dieter: *Mensch-Computer-Interaktion. Zur Geschichte und Zukunft der Computerbedienung*. Bielefeld: transcript 2008, S. 173-196.
- Nake, Frieder (2008): „Zeigen, Zeichnen und Zeichen. Der verschwundene Lichtgriffel“, In: Hellige, Hans-Dieter (Hg.): *Mensch-Computer-Interface: Zur Geschichte und Zukunft der Computerbedienung*, Bielefeld: transcript 2008, S.121-154.
- Nietzsche, Friedrich (2002): *Schreibmaschinentexte. Vollständige Edition, Faksimiles und kritischer Kommentar*. Aus dem Nachlaß hg.v. Stephan Günzel und Rüdiger Schmidt-Grépalý, Weimar: Bauhaus-Universitätsverlag.
- Perkins, Roderick/Smith-Keller, Dan/Ludolph, Frank (1997): „Inventing the Lisa User-Interface“, In: *Interactions*, 4 (1), S. 40-53. ACM-Digital Library, <http://portal.acm.org/citation.cfm?id=242405> (20.11.2010)
- Pflüger, Jörg (2008): „Interaktion im Kontext“, In: Hellige, Hans-Dieter (Hg.): *Mensch-Computer-Interaktion. Zur Geschichte und Zukunft der Computerbedienung*. Bielefeld: transcript 2008, S. 323- 389.
- Pias, Claus (2000): „Computer Spiel Welten“, Dissertation. Bauhaus Universität Weimar.
- Pias, Claus (2001): „Punkt und Linie zum Raster“, In: Brüderlin, Markus (Hg.): *Ornament und Abstraktion. Kunst der Kulturen, Moderne und Gegenwart im Dialog*. Katalog zur gleichnamigen Ausstellung, Köln: DuMont, S. 64-69.
- Reifenrath, André (2000): „Die Geschichte der Simulation“, Dissertation. Humboldt Universität Berlin.
- Roch, Axel (1998): „Die Geschichte der Computermaus. Von der elektrischen zur taktischen Feuerleitung“, In: *Telepolis*, 28.08.1998, [http:// www.heise.de/tp/r4/artikel/2/2440/1.html](http://www.heise.de/tp/r4/artikel/2/2440/1.html) (19.11.2010)

- Rosenberg, Nathan/Steinmüller,Edward W. (1982): „The economic implications of the VLSI revolution”, In: Rosenberg, Nathan (Hg.): *Inside the black box: Technology and Economics*. Cambridge, Mass.: Cambridge University Press 1982, S. 178-192.
- Ross, Douglas (1956): „Gestalt Programming: A new Concept in Automatic Programming”, In: *Proceedings of the AFIPS Joint Computer Conferences*, Februar 1956, S. 5-11. ACM-Digital Library, <http://portal.acm.org/citation.cfm?id=1455410.1455414> (20.11.2010)
- Schirra, Jörg (2005): „Computervisualistik”, In: Sachs-Hornbach, Klaus (Hg.): *Bildwissenschaft. Disziplinen, Themen, Methoden*. Frankfurt a.M.: Surhkamp 2005, S. 268-280.
- Schwartz, Jules I./ Goffman E.G./ Weissman C. (1964): „A General-Purpose Time-Sharing System”, In: *Proceedings of the AFIPS Joint Computer Conferences*, April 1964, S. 397-411. ACM- Digital Library, <http://portal.acm.org/citation.cfm?id=1464122.1464163> (12.11.2010)
- Shneiderman, Ben (1983): „Direct Manipulation: A Step Beyond Programming Languages", In: *IEEE Computer Society Press*, 16 (8), S. 57-69. CS Digital Library, <http://www.computer.org/portal/web/csdl/doi/10.1109/MC.1983.1654471> (20.11.2010)
- Smith, David Canfield et.alt.(1982): „Designing the Star User Interface”, In: Pylyshyn, Zenon W./Bannon, Liam (Hg.): *Perspectives on the Computer Revolution 2nd Edition*. Norwood: Ablex Publishing Corporation 1989, S. 261-283.
- Smith, David Canfield (1977): *Pygmalion. A Computer Program to model and stimulate creative Thought*. Stuttgart: Birkhäuser.
- Smith, David Canfield (1993): „Pygmalion: An Executable Electronic Blackboard”, In: Cypher, Allen (Hg.): *Watch What I Do: Programming by Demonstration*. Cambridge: MIT Press, S. 19-48.
- Stillich, Sven (2008): „Weltherrschaft verschlafen”, In: *Einestages. Zeitgeschichten auf Spiegel Online*, 29.10. 2008, http://einestages.spiegel.de/static/topicalbumbackground/3046/weltherrschaft_verschlafen.html (18.11.2010)
- Sutherland, Ivan (1963): „Sketchpad: A Man-Machine Graphical Communication System”, Dissertation, MIT. CiteSeer-Scientific Literature Digital Library and Search Engine, elektronische Neuauflage mit einem Vorwort von Alan Blackwell und Kerry Rodden. Technical Report 257 Cambridge, Mass. 2003, <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.10.4290> (18.11.2010)
- Taylor, Norman /Hurst, Jan/ Mahoney, Michael S./ Ross, Douglas T./Fano, Robert M. (1989): „Retrospectives I: The Early Years in Computer Graphics at MIT; Lincoln Lab and Harvard”, In: *ACM SIGGRAPH Computer Graphics Panel Proceedings*, 23 (5), S.19-38. ACM-Digital Library, <http://portal.acm.org/citation.cfm?id=77277.77279> (12.11.2010)
- van Dam, Andries (1997): „Post-WIMP User Interfaces”, In: *Communications of the ACM*, 40 (2), S. 63-67. ACM-Digital Library, <http://portal.acm.org/citation.cfm?id=253708> (23.11.2010)

Weibel, Peter (1995): „Die Welt der virtuellen Bilder. Zur Konstruktion kontextgesteuerter Ereigniswelten“, In: Dencker, Klaus Peter (Hg.): *Weltbilder- Bildwelten. Computergestützte Visionen. Interface 2*. Hamburg: Verl. Hans-Bredow Institut, S. 34-47.

Abbildungsverzeichnis

Ich habe mich bemüht, sämtliche Inhaber der Bildrechte ausfindig zu machen und diese entsprechend anzuführen. Sollte dennoch eine Urheberrechtsverletzung bekannt werden, ersuche ich um Meldung bei mir.

Abb.1: Friedewald, Michael (1999): *Der Computer als Werkzeug und Medium. Die geistigen und technischen Wurzeln des Personal Computers*, S. 270.

Abb.2: IBM-Archiv, http://www-03.ibm.com/ibm/history/exhibits/vintage/vintage_4506V-V2216.html (23.11.2010)

Abb.3: Screenshot aus Corbato, Fernando J. (1963): „1963 Timesharing: A Solution to Computer Bottlenecks“, 5'24''.

Abb.4: veröffentlicht von Sun Microsystems, design.osu.edu/.../pages/ivan-sutherland_jpg.htm (20.11.2010)

Abb.5: Engelbart, Douglas (1962): „Augmenting Human Intellect- A conceptual Framework“, 2c3a1.

Abb.6-9: English, William/Engelbart, Douglas/Berman, Melvin (1967): „Display-Selection-Techniques for Text Manipulation“

Abb.10: Screenshot aus „Douglas Engelbart: The Mother of All Demos (2/9), online-Video, <http://www.youtube.com/watch?v=a11JDLBxtPQ&feature=related> (15.11.2010), 0'39''.

Abb.11: Kay, Alan/Goldberg, Adele (1977): „Personal Dynamic Media“, S. 394.

Abb.12: Kay, Alan (1990): „User Interface: A Personal View“, S. 197.

Abb.13: Kay, Alan/Goldberg, Adele (1977): „Personal Dynamic Media“, S. 395.

Abb.14: Goldberg, Adele/Kay, Alan (1977): „Methods for Teaching Smalltalk“, S.12.

Abb.15: Lampson, Butler (1986): „Personal Distributed Computing: The Alto and Ethernet Software“, S.123.

Abb.16: Johnson, Jeff et.al (1989): „The Xerox ‚Star‘: A Retrospective“, S. 60.

Abb.17: Homepage von Eric S. Raymond: Online-Artikel über „A brief History of GUIs“, <http://catb.org/~esr/writings/taouu/html/ch02s05.html> (14.11.2010)

Abb.18-20: Homepage von Christoph Dernbach: Mac History. Fakten und Geschichten rund um Apple und den Macintosh, <http://www.mac-history.de/allgemein/2008-05-25/apple-lisa>, <http://www.mac-history.de/die-geschichte-des-apple-macintosh> (26.11.2010)

10. Danksagung

Ich danke all den Menschen, die mich im Prozess der Ideenfindung und des Schreibens mit ihrer Geduld und Hilfsbereitschaft begleitet haben. Besonderer Dank gilt Matthias Müller-Prove, der mir mit seinem fachlichen, informatischen Wissen zur Seite stand sowie meinem Diplomarbeitbetreuer Univ.-Prof. Dr. Klemens Gruber, dessen konstruktive Kritik und Ratschläge mir die Wichtigkeit lehrten, meine Gedanken zu fokussieren ohne dabei den größeren Zusammenhang aus den Augen zu verlieren.

Weiters bin ich all jenen Situationen und Begegnungen dankbar, die mich stets daran erinnern, dass das Leben aus weit mehr besteht als nur aus Bits und Bytes.

11. Abstract

Ausgangspunkt dieser Arbeit ist die Feststellung, dass sich heutige Ordnungen des Wissens, der Erfahrung und der Kommunikation zunehmend entlang der Performanz digitaler Technologien, besonders des PCs, aufspannen. Die Mensch-Computer-Interaktion ist ein omnipräsentes und soziokulturell höchst relevantes Phänomen. Untersuchungsgegenstand ist daher jene Instanz, welche den Computer erst zum Medium für die Masse avancieren ließ indem sie dessen technische Funktionen allgemein zugänglich machte - die grafische Benutzeroberfläche. Seit der Kommerzialisierung des PCs in den 80er Jahren und besonders durch die Verbreitung des World Wide Web ab den 90er Jahren bedingt und prägt sie wesentliche Praktiken der digitalen Alltagskultur. Neben der Betrachtung ihrer konstitutiven Elemente und den dadurch etablierten Konventionen der Mensch-Computer-Interaktion beschäftigt sich die Arbeit mit der historischen Entwicklung der grafischen Benutzeroberfläche. Der Fokus liegt dabei auf einer ideengeschichtlichen Darstellung des Verlaufs der Mensch-Computer-Interaktion seit ihren Anfängen, den Kontinuitäten und Brüchen sowie der Erörterung markanter hard- und softwaretechnischer Innovationen der Benutzerschnittstelle.

This paper's starting point is the observation that today's orders of knowledge, experience and communication increasingly unfold around the performance of digital technologies, especially the PC. Human-Computer-Interaction is an omnipresent and sociocultural extremely relevant phenomenon. The object of investigation is therefore the very instance which caused the computer to become a massmedium through making its technical functions accessible to the public - the graphical User-Interface. Since the PC's commercialisation in the eighties and especially since the rise of the World Wide Web in the nineties it determines and coins essential practices of everyday digital culture. Besides considering its constitutive features and the thereby established conventions of Human-Computer-Interaction, the paper deals with the historical development of the graphical User-Interface. It focuses on the progress of Human-Computer-Interaction's history of ideas since its beginning, the continuities and discontinuities and considers distinctive hard- and softwaretechnological innovations of the User-Interface.

Lebenslauf

Steixner Ines, geboren am 6.11.1985 in Rum in Tirol.

Ausbildung

ab WS 2005: Wechsel auf Diplomstudium der Theater-, Film- und Medienwissenschaften

SS 2005: Bakkalaureatstudium Dolmetschen Französisch/Englisch

1996- 2004: Matura, Humanistisches Gymnasium der Franziskaner Hall in Tirol

Berufliche Erfahrung/ Projekte (Auswahl)

08/ 2010: Videodokumentation vom sowie Mitarbeit beim interkulturellen Ubuntu-Theaterprojekt „Wir bauen eine Regenbogenbrücke von mir zu dir“ im SOS Kinderferiendorf in Caldonazzo (Italien)

09-10/2006: organisatorische Mitarbeiterin beim Reykjavik International Filmfestival (Island)

06-10/2005: Lichttechnikassistentz sowie organisatorische Mitarbeiterin am Theater Nestroyhof (Wien)

06-12/2004: Regieassistentz „Monsieur Ibrahim und die Blumen des Koran“, Fabian Kametz am Innsbrucker Kellertheater (Tirol)