

Kreative Software Menschen und Meilensteine



Selten zuvor hat sich in der Geschichte des Menschen ein Werkzeug so schnell durchgesetzt wie der Personal Computer. Die Ausstellung um Geschichten und Mythen dieser wundersamen Maschine und ihrer Helden überschaubar die Zeitspanne von ihren Anfängen bis zur Anerkennung als unverzichtbares Arbeitsmittel. Bei den Ausstellungsrecherchen stießen wir oft auf Erstaunen. Geschichte, die noch nicht einmal 20 Jahre alt ist, sollte ausgestellt werden. Um so erstaunlicher war, dass eben diese Geschichte schon im Dunkel selbstgestrickter Firmenmythen und journalistischer Ausschmückungen verschwindet.

Die Ausstellung beschränkt sich bewusst auf Personal Computer und ihre Software. Obwohl sich Entwicklungen bei Mainframes, Mini- und Microcomputern wiederholt haben, sind uns letztere in ihrer Alltäglichkeit und Philosophie am nächsten.

Es bleibt abzuwarten, ob die immer einfachere Bedienbarkeit der Maschinen letzten Endes zu ihrem Verschwinden im Sinne einer Absorption in den Alltag führen wird. Hier jedenfalls sehen Sie erst noch einmal Meilensteine und Maschinen.

Benjamin Heidersberger

A Star is born

Wenn Agenten in Hollywood einen Star, einen Regisseur und einen Produzenten verkuppeln wollen, brauchen sie einen Mikro-Filmplot.

Zeit: 1984. Schauplatz: Pastellfarbenes "Man-kennt-sich"-Restaurant. Tische werden nach monatelangem Warten oder dank mehrfacher Nennung in den wichtigen Gesellschaftsspalten zugeteilt; europäische Küche, Chef de Cuisine: Wolfgang Ich-hab-den-Nachnahmen-vergessen; Palmen signalisieren: Wir sind in Hollywood.

Der Agent und der Produzent treffen sich zum Power-Lunch. Wir sparen uns die verletzenden Höflichkeiten und den lebenswichtigen Tratsch.

Agent: Hacker! Was für ein Konzept!

Produzent. Hacker? Pac Man und Space Invaders? Keine Science-fiction-Filme! Schaut kein Schwein.

Agent (enthusiastisch): Nix Science-fiction: Science-Fact! Hier sind die Artikel: New York Times, Time, Newsweek. Wir leben im Zeitalter des Computers!

Produzent (gelangweilt): Wer will schon picklige junge Typen vor dem Computer sehen?

Agent: Aber wenn sie die Welt retten?

Produzent (genervt): Quatsch!

Agent: Ein Computerfreak treibt die Welt an den Rand des Atomkrieges. Unsere Computer und die der Russen stehen kurz davor, die Raketen zu feuern (fuchtelte mit den Händen in der Luft herum): Dampfende Raketensilos vor dem Abschuß!

Produzent: Blödsinn, Hacker spielen doch nur Space Invaders.

Agent. In der Zeitung steht, die Hacker gefährdeten die nationale Sicherheit.

Produzent: Spionage ist gut, aber wir brauchen einen Helden und eine Lovestory.

Agent (Chance witternd): Und einen Bösewicht. Computer sind gute Schurken — und kosten auch keine Millionengage. Wie wär's damit: College-Kid will die neusten Computerspiele via Telefonleitung stehlen, hackt sich stattdessen in das Computersystem des Pentagons hinein. Der Supercomputer wittert einen Angriff und öffnet die Raketensilos. Militärs und Politiker sind unfähig, das sprechende Elektronenhirn zurückzupfeifen — die Welt steht vor dem Armageddon. Doch fünf Sekunden vor Zwölf knackt der Hacker den Code. Dem Megabrain des Pentagon wird klar, daß alles nur ein Simulationsspiel ist, die Raketen werden deaktiviert. Abspann.

PRODUZENT: Gut, daß der jugendliche Hacker die Schwachstelle des Computers offengelegt hat und nicht der Russe oder ein Terrorist.

Agent (ruft Kellner): Können Sie mir die Rechnung bringen?



Der klassische Großrechner, eine IBM 360 Modell 85. Nach unbestätigten Berichten "360", weil alle Bedürfnisse des Kunden rundum befriedigt werden.

Foto: Boston Computer Museum



Blick in das Kabinett "Universal Rechner in Transistortechnik" des Deutschen Museums München

Foto: Deutsches Museum

Von der Lochkarte zur Bildröhre

Es gab einmal eine Zeit, da waren Computer wie Behörden: aufgeblähte Apparate, die Dienst nach Vorschrift leisteten. Jede Eingabe mußte zuerst umständlich in einem Formular umschrieben werden. Systemmanager, kafkaeske Wärter, nahmen die Formulare in Empfang, fütterten die allmächtige Maschine damit und beförderten, nach Fertigstellung der Sachbearbeitung, das Ergebnis wieder hinunter zu dem ungeduldig wartenden Bittsteller.

Die Formulare waren kleine, meist blaßgelbe Kärtchen mit vielen Löchern, die alle Eckdaten des menschlichen Lebens präzise kodieren konnten: Name, Vorname, Anschrift, Geschlecht, Gehalt, Kontonummer — alles eine Frage des Lochs an der richtigen Stelle.

So fraßen sich die Behörden-Computer durch lange Lochkarten-Stapel und machten Personalverwaltung, Volkszählung, Einwohnererfassung oder PKW-Registration möglich.

Wenn Herr oder Frau Normalbürger mal einen Computer zu Gesicht bekamen, in Film, Fernsehen, Illustrierten oder Büchern, waren wuchtige, eckige Maschinenschränke mit vielen blinkenden Lämpchen und sich fleißig drehenden Spulen zu sehen. Im Vordergrund standen die Hohepriester der Maschine: ernste Männer in weißen Laborkitteln, meist emsig damit beschäftigt, ein Magnetband auszuwechseln.

Im Science-fiction wurde die landläufige Vorstellung vom elektrischen Leviathan ins Extrem gedreht: Arthur C. Clarkes Buch "2001 — A Space Odyssey", das von Stanley Kubrik zum psychedelischen Kultfilm verarbeitet wurde, machte den Computer zum Mephisto: HAL (IBM im Alphabet immer um einen Buchstaben voraus), der Bösewicht mit der freundlichen Stimme.

Dieses diabolische Image hielt sich ziemlich lange: Von 1941 — in diesem Jahr hat der Deutsche Konrad Zuse den ersten dieser Behörden-Computer gebaut — bis 1982.

Ganz im Verborgenen begann im Jahre 1969 die technische Umwälzung: Computeringenieure der texanischen Firma Datapoint beauftragten die Firmen Intel und Texas Instruments, einen Einzellen-Computer zu entwickeln: die wesentlichen Funktionen eines Computers dichtgedrängt auf einem fingernagelgroßen Chip unterzubringen.

Intel schaffte die technische Großtat, aber dem Auftraggeber Datapoint war der neue Chip zu langsam. Die Manager bei Intel wollten ihre Entwicklung nicht wegwerfen, und so brachten sie den Baustein mit der Typenbezeichnung "4004" ein Jahr später auf den Markt. Das war die Geburtsstunde des Mikroprozessors. Die grundlegenden Lebensfunktionen dieser Computer-Einzeller waren die gleichen wie die der Dinosaurier, der Großrechner.

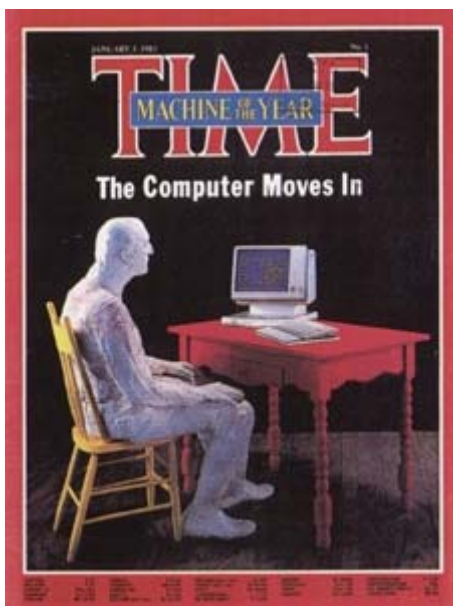


Helden der PC-Fans: Steve Wozniak & Steven Jobs
Foto: MACup Archiv

Sofort begannen Tüftler in ganz Amerika ihr Spiel mit dem neuen Bauteil. Viele Konstruktionen auf Chipbasis landeten in der evolutionären Sackgasse: Frankenstein-Computer, weder kommerziell noch technisch erfolgreich, Rechenkisten, die nur von leidenschaftlichen Bastlern gekauft wurden. Einige der Hobbycomputer-Bastler waren jedoch erfolgreich und ließen schließlich die evolutionäre Ursuppe der Hackerszene hinter sich. Stephen Wozniak entwickelte in Kalifornien den ersten Apple — klein genug für den Schreibtisch oder Küchentisch, mit Tastatur und der vertrauten Glotze als Bildschirm. 1977 führten "Steve Woz" und sein Freund Steven Jobs stolz den Apple II vor. Aus dem Team Wozniak & Jobs wurde ein Mythos; die beiden jungen Amerikaner wurden zu Helden der Success Story, die Apples PR-Firma den Journalisten und der Computerfangemeinde immer wieder vorgebetet hat: in Turnschuhen schnell mal von der Garagenklitsche zur Milliarden-Dollar-Company sprinten. Mit Cleverness, Groove und Enthusiasmus den Marktriesen IBM herausfordern.

Damit war der David-und-Goliath-Mythos der Computerbranche in Szene gesetzt, der fortan die Durchsetzung des "Personal Computers" und der neuen, aggressiv auftretenden Hersteller begleitete. Zwar konnten die Käufer der klobigen Behördencomputer in den großen Labors und Verwaltungsbüros ihre Computer mittlerweile auch mit Tastatur und Bildschirm bedienen, allerdings nur während der zuvor eingeteilten Schalterstunden — denn Rechenzeit war teuer.

Ein paar Jahre später schwenkte schließlich auch IBM, die Firma mit dem Bürokratenimage, auf den neuen Kurs ein. 1981, fünf Jahre nachdem auch Commodore in diesen Markt eingestiegen war, brachte das damals sechstgrößte Unternehmen der Welt auch so einen kleinen, feinen PC auf den Markt.



Von "Time" zum "Man of the Year 1982" gekrönt: der PC

1982 hatte der Personal Computer die Vorstellung von einem Computer schon vollständig besetzt: Das US-amerikanische Nachrichtenmagazin Time krönte den Computer zum "Mann des Jahres", auf dem Titelbild eine Gipsfigur vor einem PC, die Hände im Schoß, argwöhnisch. "HELLO/MAY I HELP YOU?" fragt der Rechner. Wer würde da Nein sagen?

Zehn Jahre später: Die Japaner zeigen erste Studien vom Body-Computer, der am Handgelenk getragen wird. Computer, die aussehen wie Schiefertafeln und nur noch mit Griffel bedient

werden, sollen das unentbehrliche Filofax ersetzen; Telefon und Computer zeugen Kommunikationsshelfer, die über Mobilfunk-Netz alle jederzeit an das globale Netz anklinken.

Exkurs: Der Hacker in seinem natürlichen Lebensraum; Zitat aus dem kleinen Handwörterbuch der Medienklischees, 29. Auflage, Hamburg, 1992:

HACKER, der [hähköhr]. — meist unsportlicher, junger Mann mit blassem Teint, Bartstoppeln, fettigen Haaren und tiefen Rändern unter den Augen. Der H. sitzt gern in kleinen, stickigen Zimmern. Sein Lebensmittelpunkt ist der Computer, vorzugsweise über das Telefonnetz verbunden mit Großrechnern des Militärs oder internationaler Konzerne. Ziel des H. ist es, das Sicherheitssystem fremder Computer zu knacken, um militärische oder kommerzielle Geheimnisse an die H.-Öffentlichkeit weiterzugeben.

Während dieser Tätigkeit schläft der H. wenig und spricht kaum. Gerne stöpselt er sich Walkman-Hörer in die Ohren. Der H. ernährt sich von Pizza, Hamburger, Coca-Cola, Bier, Kartoffel-Chips und appetitzügelnden, leistungssteigernden Psychopharmaka. In der knappen Freizeit beschäftigt sich der H. gerne mit Fragen wie "Gibt es im Simulationszeitalter noch einen Gott? Ist es sündig, Roboter und künstliche Intelligenzen zu konstruieren? Werde ich jemals eine Frau finden?" Geselliges Leben übt der H. manchmal in speziellen H.-Clubs.

Zur Etymologie: "to hack", engl. Verb [mittel-engl.: hakken]: etwas mit unregelmäßigen, groben Hieben zerhacken, aufbrechen; "hack", engl. Substantiv [abgeleitet vom mittlengl. "hackney"]: heruntergekommenes, leihbares Zuglast-Pferd, übertr.: käufli. Politiker, Lohnschreiber, Taxifahrer, Gefängniswärter; davon abgeleitetes engl. ugs. Verb "to hack": sich als billige Routine-Arbeitskraft verdingen, etwas hinpfuschen; "hacked", engl. Adjektiv zur Bezeichnung eines bestimmten Arbeitsstils: banal, abgegriffen, billig, routinemäßig, kommerziell. (Quelle: The American Heritage Dictionary of the English Language, College Edition)

Im alltäglichen Sprachgebrauch gilt der H. als Spontan-Programmierer, der versucht, Code-Probleme durch Herumspielen zu lösen — so lange, bis er es schafft oder die Maschine ihn.

Erster Auftritt des H.: 1961 wurde aus dem Modelleisenbahnclub der Ostküsten-Uni Massachusetts Institute of Technology (MIT) ein Computerclub. Dort fanden sich die ersten H. zu nächtlichen Programmiersessions ein, mit dem Ziel, das perfekte Programm zu schreiben, ein Megabrain zu schaffen.

In der Hochzeit des Computermithos, zwischen 1979 und 1989, galten die H. als Alchimisten, eingeweiht in die Geheimwissenschaft der Computer.

Verbreitung des H.ismus: Die Westküsten-H. haben den Mythos um das süßliche Aroma der Flower-Power-Bewegung angereichert. Pot-rauchende Bay-Lefties fanden sich in der People's Computer Company zusammen und prägten die moderne Robin-Hood-Losung "Computer power to the people": Befreit den Informationsfluß aus den gierigen Fängen des bürokratischen Molochs IBM! Als die Stars der H.-Szene jedoch ins Business einstiegen, nur noch bolivianisches Marschierpulver an ihre Nasen ließen und nicht mehr mit jedem Partygast reden wollten, verlegte sich der H.ismus auf preiswerte, freie Telekommunikationsnetze. Die Losung: Jeder hat ein Recht auf internationalen Partyklatsch! Ein auch in Europa beliebter Ableger des Westküsten-H.ismus ist das von Steward Brand herausgegebene "Whole Earth Review".

In Europa gibt es verschiedene nationalfolkloristische Versionen des H., in Deutschland ist das H.-Bild geprägt von der Selbstdarstellung des Chaos Computer Clubs zur Zeit der zynischen 80er Jahre.

Derzeit ist der H. eine Schattengestalt, die durch die Glossen deutscher Zeitgeist-Magazine huscht.

Die Ästhetik des Programmierens

Nicht nur Hobbyprogrammierer bewegen sich im Konflikt zwischen brillanten Entwürfen und verbissenem Herumprobieren. Damit müssen sich Profiprogrammierer genauso herumschlagen. Der Unterschied zwischen beiden besteht vor allem in der Einbindung der Profis in einen kommerziellen, arbeitsteiligen Zusammenhang. Zweitens haben Profiprogrammierer meistens mit größeren Programmen zu tun. Und ab einem gewissen Programmumfang wird die Methode, Dinge einfach auszuprobieren, immer riskanter. Es wird immer wichtiger, bei jedem Schritt genau zu wissen, welche Wirkungen computerintern ausgelöst werden.

Die Arbeit, Programme zu entwerfen und zu schreiben, hat etwas von der Schönheit der Mathematik (alles fügt sich).

Andererseits unterliegt sie einer nachvollziehbaren Funktionsprüfung (das Ding muß laufen; wenn nicht, dann ist es schlechtes Handwerk, Pfusch), und schließlich steht sie unter dem Druck gnadenloser kommerzieller Erfolgskriterien. Die Termine, die ein Programmiererteam der Geschäftsleitung oder dem Auftraggeber im Leichtsinn vorschlägt, müssen eingehalten werden. Wenn das nicht gelingt, kommt die Abteilung aufs Abstellgleis. Schlimmstenfalls kann die ganze Firma Bankrott anmelden.

Programmierer reden lieber über die spaßige Seite ihrer Arbeit, über die Abenteuer der Abstraktion, den Triumph, riesige Programme elegant auf die Reihe zu bekommen. Informatiker weisen, in aller Bescheidenheit, öfter mal auf den genialischen Charakter ihrer Arbeit hin. Tatsächlich aber geht es bei aller Erfinder-Mythologie ganz einfach nur darum, die bessere Maschine zu entwickeln. Was schwierig genug ist, mindestens so mühsam wie das Ausmisten eines Augias-Stalls. Denn es geht darum, ein möglichst einfaches, eingeschränktes, aber adäquates Modell der Wirklichkeit zu formulieren.

C. Wayne Ratliff zum Beispiel berichtet, daß er die Idee für dBase (das meistverbreitete Datenbanksystem, das 1978 auf den Markt kam) fand, als er ein mathematisches Verfahren für seine Fußballtip-Gruppe entwickeln wollte: "Ich versuchte, die Siegesmannschaft zu bestimmen, indem ich Zeitung für Zeitung durchging — ein entsetzliches Verfahren. Irgendwann habe ich entschieden, daß das ohne Computer nicht machbar war. Nach einer Woche hatte ich den Fußball völlig vergessen. Ich hatte entschieden, daß die Welt einen Datenbankmanager mit natürlicher Sprache brauchte."

Ratliff hat sich immer ganz eng an der Frage orientiert, was seine Zielgruppen brauchen könnten. An diesem Ziel gemessen, schränkt Ratliff ein, sei dBase "nicht perfekt": "Wäre ich meinem Herzen gefolgt", erklärt er, "und nicht dem Rat anderer (mach' es größer, mach' es schneller, mach' es kleiner, gehe auf 16 Bit, mach' es Multi-User fähig, nimm mehrere Sprachen) — wäre dBase fast perfekt geworden. Ich versuchte einfach, in zu viele Richtungen zu gehen, um jeden wenigstens ein kleines bißchen zufrieden zu stellen. Letztendlich war das mein eigener Fehler."

Perfektionismus ist die Berufskrankheit aller Programmierer. je größer die Programme sind, desto wichtiger ist es, daß sich die Programmierer an ihre Ideale halten: saubere Reduktion, präzise Ordnung, Vereinfachung, Strukturierung, Organisation. Der Reiz, sich einfach hinzusetzen und Seite um Seite Code herunter zu schreiben, ist, so erzählen viele Informatiker, groß. Aber auch das größte Programm muß überschaubar bleiben. Andernfalls könnte es ein zusammengestückelter, wirrer Datenhaufen werden: "Hack". Das Ideal eines Programmierers aber ist Brillanz, Eleganz, Ästhetik.

"Je besser der Techniker, desto ästhetischer und künstlerischer die Maschine, desto größer die Chancen, daß sie funktioniert", sagt Bob Frankston, der zusammen mit Dan Bricklin das Erfolgsprogramm VisiCalc (Tabellenkalkulation) geschrieben hat. "Gute Technik ist guter Kunst sehr ähnlich", meint Frankston. Den ersten Entwurf für VisiCalc hatte der Harvard-Student Dan Bricklin während seiner Studienzeit Ende der 70er Jahre geschrieben. 1979 gründete er mit Bob Frankston die Firma Software Arts, um das Programm fertigzuschreiben. Dan arbeitete tags, Bob nachts. "Wir hatten jedoch damals keinen blassen Schimmer, wie elektronische Tabellenkalkulationen aufgenommen werden würden", erzählt Bricklin.

Das Programm wurde ein Riesenerfolg. Trotzdem sind Dan Bricklin und Bob Frankston weder besonders reich noch besonders berühmt geworden. Die Firma Software Arts wurde zerrieben in einem Rechtsstreit mit ihrem Geburtshelfer, der das erste Arbeitsgerät geliehen und die Rolle des Verlegers übernommen hatte.

Was Programmieren sei, formuliert Bricklin eher entspannt: "Ein Teil ist Handwerk, und ein Teil ist Wissenschaft. Wie bei vielen Dingen führt die Praxis zur Perfektion. Es ist kein fest vorgeschriebener Ablauf, sondern eher eine handwerkliche Kunst. Programmierer mit fundierter Ausbildung haben meistens einen Vorteil gegenüber jenen, die keine haben. Manche Leute haben ein glückliches Händchen, andere nicht, aber es ist immer gut, wenn man sich mit einem Gebiet grundlegend befaßt. Wenn man Produkte für kommerzielle Anwendungen macht und die Sachen raus müssen, darf man die Arbeit nicht zu eng sehen."



Das Rennpferd der Branche Bill Gates
Foto: Microsoft

Bill Gates, der von allen Informatikern die steilste Karriere hinter sich hat, formuliert die Ansprüche an "Spitzenprogrammierer", die Rennpferde der Branche, schon sehr viel strenger.

Mitte der 70er Jahre hatte er als Harvard-Student einen Basic-Interpreter für den ersten kommerziellen Mikrocomputer, den MITS-Altair, geschrieben, zusammen mit seinem früheren High-School-Mitschüler Paul Allen. Nachdem der Compiler fertig war, gründeten die beiden Freunde die Firma Microsoft. Gates ist, wie viele gute Programmierer, die als Jugendliche ihre Nächte vor dem Rechner verbracht haben, ins Management gegangen. Heute ist der 36jährige einer der reichsten Menschen der USA.

"Die Tage, in denen jedes Programm ein Meisterstück war, sind vorbei", sagte Gates in einem Interview, das Mitte der 80er Jahre geführt wurde. Und dennoch beharrte er auf die Arbeitsprinzipien der ersten Tage: "Die beste Software entsteht, wenn ein einzelner sich genau vorstellen kann, wie das Programm funktioniert. Dazu muß man das Programm lieben und sich darauf konzentrieren, es einfach zu halten, und zwar ganz unglaublich einfach."

Gates betonte jedoch auch, daß er nichts von einem "Primadonna-Kult" hält, "wo jemand, bloß weil er gut ist seinen Code nicht kommentieren mag oder nicht mit anderen Leuten reden will oder allen anderen seine Vorstellungen aufdrängt. Gates weiter: "Wir brauchen Leute, die sich gegenseitig respektieren. Ich glaube, daß die meisten großen Programmierer gern mit anderen großen Programmierern zusammen sind. Wenn sie sich einen unglaublichen Algorithmus ausgedacht haben, sind sie gern mit Gleichgesinnten zusammen, die das, was sie sich ausgedacht haben, auch verstehen und anerkennen. Denn wenn man mit so einer Idee kommt und das Modell fest im Kopf hat ist das eine einsame Sache. Wenn man angenommen hatte, ein Prozeß würde kompliziert werden, und dann doch einen Weg findet ihn zu vereinfachen, ist das ein herrliches Gefühl. Aber dazu braucht man Feedback von anderen."

Ein guter Teil der Programmierarbeit geht jedoch für die Bekämpfung der "Bugs" drauf (abgeleitet vom amerikanischen Alltagswort "to bug": stören, belästigen, ein Verb, das von "Bug": Käfer, Bazillus, abstammt). "Debugging", die Fehlerbeseitigung, ist eine der nervenaufreißendsten Arbeiten bei der Computerentwicklung. Die Hardware-Ingenieure ("Hardy-Boys", nach einer bei kleinen Jungs beliebten Kriminalroman-Serie) haben beim Debugging mitunter auch mit echten Käfern, Spinnen, Staub und ähnlichen unberechenbaren Widrigkeiten aus dem Reich der Kleinorganismen zu kämpfen. Meistens streikt die Maschine aber wegen Konstruktionsfehlern. Software-Programmierer müssen sich beim Debugging mit Denkfehlern und unsauberen Lösungen herumschlagen, die nie — und wäre der Programmierer auch so "perfekt" wie ein Rechner — vollständig vermieden werden können.



Telefonvermittlung zu Beginn des Jahrhunderts
Foto: Hamburger Postmuseum am Stephansplatz

Alltag am Computer

Software entsteht also in einem anstrengenden, oft lustvollen, meist hektischen Prozeß, der zugleich fachliche Phantasie und tendenziell bürokratische Organisation erfordert. Die Qualität größerer Software hängt sehr stark von der Kooperation der Programmierer ab, vom Informationsaustausch zwischen allen Beteiligten, von der Qualifikation dieser Leute und den Möglichkeiten, sich in neue Gebiete einzuarbeiten. Letztlich wird die Qualität des Arbeitsprodukts, wie Bjarne Stroustrup sagt, der das Programmierwerkzeug C++ entwickelt hat, von der "Kultur" in den Software-Werkstätten bestimmt. Und um die ist es, besonders in kleineren Firmen, oft schlimm bestellt. In Silicon Valley, der Hexenküche der Computerindustrie, sind hochqualifizierte Programmierer ständig auf dem Sprung in den nächsten, besser bezahlten Job und haben daher wenig Interesse, sich lange in die Techniken einer einzelnen Firma einzuarbeiten. Was Manager wiederum dazu treibt, die Fähigkeiten der Mitarbeiter noch schneller zu nutzen.

Der Mythos, man könnte sich schnell mal zum großen Geld durchhacken, zeigt inzwischen kontraproduktive Wirkungen. Die Zeiten, als die flinke Nachtschicht noch mit steil ansteigenden Aktienanteilen bei Laune gehalten werden konnten, sind vorbei. In der Regel bekommen Programmierer mittlere Gehälter, wie alle anderen Facharbeiter auch.

Software wird also nicht immer unter idealen Bedingungen entwickelt. Um sich gut zu verkaufen, darf man ihr das aber nicht ansehen. Die Kunden wollen Software, die ganz einfach ein handliches Werkzeug ist: unkompliziert, übersichtlich, multikulturell wie Rock-Musik und so freundlich wie eine Linienflug-Stewardess, der nichts Menschliches fremd ist.

Immer noch sind die meisten Bürobedarfsprogramme reichlich kompliziert und ziehen den Benutzer ein Gestrüpp von Funktionen hinein, die nicht immer so ganz rational geordnet sind. Und da die wenigsten Büroangestellten Zeit haben, Alice in Wonderland zu spielen, gewöhnen es sich die meisten EDV-Anwender bald an, sich auf die wichtigsten Funktionen zu beschränken — auch wenn die Software mit den allerschönsten Schikanen ausgestattet ist. Noch haben Büro- und Haushaltsgeräte mit Computersteuerung ganz einfach viel zu viele Knöpfe. Sie sehen aus, als sei ihr Innenleben nach außen gestülpt.

Informatiker, die an neuen Systemen arbeiten, haben wenig Skrupel, die Mängel der gegenwärtigen Computergeneration einzugestehen. Viele Profiprogrammierer haben schon vor Jahren prophezeit, daß Computer wesentlich einfacher werden in der Bedienung, bis sie so sehr oder so wenig spektakulär sein werden wie Telefone.

Das Ziel der Programmierarbeit war von Beginn an, seit Blaise Pascals erstem Entwurf einer Rechenmaschine vor 350 Jahren: komplexe Arbeitsprozesse zu vereinfachen. Die Verwandlung von intellektuellen Tätigkeiten in Funktionen der Maschine macht natürlich nicht beim PC halt. Im Gegenteil: Nach einer schwindelerregenden Phase der schnellen Entwicklungen in den letzten zwanzig Jahren steht die Computerbranche momentan vor der nächsten großen Herausforderung. Die Nachfrage für PCs ist stark abgesunken, die Preise sind abrupt in den Keller gefallen. Die Hersteller haben integrierte Systeme, die alle Abteilungen eines Großbetriebs verbinden, als nächste technische Herausforderung entdeckt. Und da diese Technik noch nicht ausgereift ist, sollen erst einmal die vorhandenen Computer vernetzt werden. Deutscher IBM-Geschäftsführer Bernhard Dorn zur Cebit-Zeitung über diese Flickschusterei an den vorhandenen Systemen: "Eigentlich müßte man alles wegschmeißen, weil es Schund ist. Fünf Jahre würde es dauern, bis überall eine neue DV-Landschaft da ist. Diesen Stillstand kann sich aber keiner leisten."

Kleine Geschichte des PC

Erst die Software macht den Computer zu einem nützlichen Werkzeug. Software soll uns helfen, Aufgaben zu erledigen oder, als Computerspiel, die eingesparte Zeit totzuschlagen. Aber erst, wenn nichts Fachfremdes mehr zwischen uns und unseren Absichten steht, ist die Software wirklich zu gebrauchen. Die "Benutzerschnittstelle" des Computers (Ein- und Ausgabemedien) ist erst dann ideal gestaltet, wenn sie die Tätigkeiten der Menschen am Computer störungsfrei unterstützt.

Die gegenwärtige Gestalt der Benutzerschnittstelle besteht aus Eingabegeräten wie Tastatur, Maus, Joystick, Plotter, Lichtgriffel, Spracherkennungsgeräten — und Ausgabegeräten: Tonausgabe sowie optische Geräte wie Drucker und Bildschirm. Elemente der optischen Aufbereitung von Informationen sind: Bitmap-Darstellung von Worten und Grafiken, mehr oder weniger strukturierte Menue-Führung und Windows ("Fenster", um die Daten mehrerer verschiedener Programme gleichzeitig auf dem Bildschirm darzustellen). Die grafischen Elemente reichen von Linien und Kästen bis hin zu Videobildern. Viele Büroprogramme arbeiten mit Icons (Piktogrammen), die Computerfunktionen als BüROUTENSILIEN symbolisieren (eine Art Eselsbrücke zwischen Schreibtisch und Computer; Terminus technicus: desktop metaphor).

Dieser individuell bedienbare, multifunktionale Computer wurde in Grundzügen schon in den 60er Jahren entwickelt. Eigentlich wurde das ganze Konzept schon 1945 ausgearbeitet von Vannevar Bush, einem wissenschaftlichen Berater von Präsident Roosevelt. Aber Bush hat das Ding, das er "Memex" nannte, ein tastaturgesteuertes Mikrofilmgerät zur automatischen Informationssortierung, nur beschreiben, nicht bauen können. Den ersten Schritt zur interaktiven, grafischen Benutzeroberfläche machte Ivan Sutherland Anfang der Sechziger mit dem Sketchpad, einem Grafikprogramm, das mit Leuchtstift auf den Monitor gezeichnete Formen verarbeiten konnte. Etwa zur gleichen Zeit entwickelte Douglas Engelbart, ein früherer Radartechniker der US-Navy, am Stanford Research Institut das "NLS" System, einen interaktiven Computer mit Maus, Röhrenbildschirm und grafischer Darstellung von Informationen in Form von Suchbäumen.

Engelbart hatte diesen ersten "Personal Computer" im Auftrag des 1962 gegründeten Advanced Research Projects Agency (ARPA) entwickelt — dem nach dem "SputnikSchock" vom Pentagon üppig finanzierten "think tank" der USA. Im ARPA wurde unter der Leitung von J.C.R. Licklider und unter Einsatz von immensen Ressourcen die Entwicklung von direkt steuerbaren Rechnern forciert — für Informatiker ein Forschungsparadies. Der "ARPA-dream" währte bis Anfang der 70er Jahre, bis zu Beginn der Nixon-Ära die Pentagon-Forschungsgelder radikal zusammengestrichen wurden und die Ressourcenverteilung wieder direkt an die Entwicklung von Militärtechnik geknüpft wurde.

An die Stelle von ARPA trat nun die Kopierfirma Xerox. Angeregt vom rapiden Wachstum der Firma Digital Equipment, beschloß Xerox, ins Minicomputer-Geschäft einzusteigen und gründete 1970 das Forschungsel dorado Xerox Palo Alto Research Center (PARC). Hier wurde 1972 der Prototyp aller nachfolgenden PC-Generationen vorgestellt: der Alto, ein Minicomputer mit einem mobilen Speicherteil (ein Vorläufer der Floppy-Diskette), BitmapBildschirmanzeige und Maus. Neun Jahre später, 1981, stellte Xerox Star vor, ein (außerhalb von PARC entwickeltes) vielseitiges Bürosoftware-System, das schon mit allen uns heute bekannten Elementen ausgestattet war. Star war das erste Softwarepaket, dessen Design voll auf Büroautomation ausgerichtet war. Das Zielpublikum waren Leute, die Informationen verwalten, den Computer nur gelegentlich benutzen und, wie es in einer XeroxRetrospektive heißt, "daran interessiert sind, ihre Arbeit zu erledigen, aber überhaupt kein Interesse an Computern haben".

Das Star-System war kommerziell eine Pleite, vor allem weil es Anfang der Achtziger noch keinen Markt gab für die teure Allzweckmaschine (\$ 15.000). Xerox hatte aufs falsche Pferd gesetzt. Anfang 1975 war schon der erste Kleincomputer im Handel, der auf dem Intel-Chip beruhte: der Altair 8800 (Firma: MITS), ein Bausatz, mit dem man eigentlich noch nichts anfangen konnte. Aber der Altair, der, so geht die Legende, seinen Namen aus der Star-Trek-Saga erhielt, wurde zum leuchtenden Stern am PC-Himmel. Zwei Jahre später hatten auch Apple, Commodore und Radio Shack kleine Personal Computer auf den Markt gebracht, die zwar auch noch nicht sehr viel leisten konnten, aber reißenden Absatz fanden. Viele der Designprinzipien des Star gingen in Apples Lisa (1983, ein Flop) und Macintosh (1984, Apples aggressiv auf den Markt gedrangtes Erfolgsmodell) ein und wurden ebenso von Microsoft aufgegriffen.

Zwischen Xerox, Apple und Microsoft hat sich längst ein Vaterschaftsstreit über die Urheberrechte an den damals entwickelten Designprinzipien entfaltet. Tatsächlich aber kam es allein schon durch den Firmenwechsel vieler Entwickler zum Know-How-Transfer. Alan Kay zum Beispiel, der die PARC-Heiligtümer DynaBook-Konzept und Smalltalk entwarf, ging als Hans Dampf durch alle Gassen der Szene: ARPA, Xerox, Atari, Apple, MIT Media Lab. Apples LisaWrite und Microsoft Word stammen von früheren Xerox-Bravo-Mitarbeitern (Tom Malloy und Charles Simonyi).

Aus dem PARC kamen auch die Nachfolger von Sutherlands Sketchpad, Grafikprogramme wie Draw (von Patrick Beaudelaire und Bob Sproull) und Doodle (von Dan Silva), und schließlich auch die Laserdrucktechnik, der Schlüssel zum Desktop-Publishing: Einige Programmierer aus der PARC-Gruppe Interpress gründeten die Firma Adobe Systems und entwickelten das inzwischen zum Standard gewordene Postscript.

Gleichzeitig wurden in der Firma Atex Textverarbeitungssysteme für Zeitungen und Zeitschriften entwickelt. Als schließlich 1985 das erste große Desktop-Publishing-System, Pagemaker, von Paul Brainerd, Ex-Atex-Mitarbeiter und Aldus-Gründer, vorgestellt wurde, war der Büro-PC-Werkzeugkasten komplett.



(Marvel Schnipp/Cosmix)





Spiele, Spaß und Spinnereien

Spielschreiber sind von allen Programmierern am stärksten auf Kundenwünsche ausgerichtet. Computerspiele müssen schon sehr ansprechend und einfach gestaltet sein, um gegen all die anderen Spielsachen anzukommen, die sich in einem Kinderzimmer so ansammeln. Amerikanische Kinder wachsen von vornherein mit einem innigeren Verhältnis zum Bildschirm auf (it's magic!). In Deutschland, haben Studien ergeben (DJI München), verlieren die meisten Kinder schon nach kurzer Zeit das anfangs noch brennende Interesse an Spielen mit dem Schreibtischrechner. Und die allerwenigsten Kinder erfüllen die elterlichen Hoffnungen und benutzen die Kiste dann dazu, kleine Basic-Programme zu schreiben. Interessanter sind da die Videospiele, denen nicht mehr anzusehen ist, daß sie rechnergesteuert sind.

Softwaredesigner, die sich mit der Gestaltung des "Interface" (der Benutzeroberfläche des Computers) beschäftigen, greifen zunehmend Ideen von Spielschreibern auf. Ihre Fragestellung ist: Wie müssen wir die Dialogebene von Business-Computern gestalten, damit die Arbeit mit ihnen "fun" wird? Die Arbeit mit dem Computer, hat zumindest — "Ich freu mich aufs Büro" — Xerox immer behauptet, soll Spaß machen.

Videospiele wenden sich an ein Publikum, das mit Computern nichts am Hut hat. Die meisten Game-Designer kommen jedoch aus der Computerszene und thematisieren in ihren Spielen Lust und Leid der Arbeit, immer und immer wieder aufs Neue: das alte Spiel von Gesetz und Zufall; unter Zeitdruck verschiedenartige Teile auf die Reihe bekommen und wegarbeiten; Eigenschaften von Objekten erkennen, ordnen, stabile Gleichgewichte herstellen (beim Simulationsspiel SimCity zum Beispiel); gegen die allwissende Maschine ankämpfen, Territorium gewinnen; Viren vermeiden, rechtzeitig speichern, damit beim Absturz nicht alles verloren geht (ein Ritual in fast allen Sierra Adventures); die Albernheit überanstrengter Kombinationskünstler (z.B. in Lucas-Adventures); Orientierung im unbekanntem, geschlossenen Raum (z.B. Colossal Cave, das legendäre, immer wieder umgeschriebene,

allererste Adventure-Spiel für den Computer: "... it is rumored that some who enter were never seen again ...").

Viele Spieleprogrammierer sind vor allem von der Idee fasziniert, den Rechner zu völlig maschinenfremden Äußerungen zu bewegen. Auch bei vielen Musik-Programmen spielt die Auseinandersetzung mit der Maschine noch eine wichtigere Rolle als die Musik (it's so hard!). Das erklärte Ziel virtuoser Software-Designer ist die Ablösung vom Maschinellen, die völlige Integration des Werkzeugs in den kreativen Prozeß. Es gibt jedoch auch eine Menge Programmierer, die sich ausgesprochen für die Autopoiesis einer sich verselbständigenden, durchdrehenden Maschine interessieren.

Ein besonders beliebter Scherz unter Programmierern, die sich mit künstlicher Intelligenz beschäftigen, sind Psychopaten-Programme. Am MIT amüsierten sich einige Programmierer damit, Szenen aus Psychiatersprechstunden zu simulieren. Marvin Minsky und seine Freunde freuten sich diebisch, als einer der KI-Programmierer gleich ein paar Psychologen mit einem Programm hereinlegen konnte, das einen paranoiden Menschen imitierte: "Wenn man etwas eintippte", erklärt Minsky, "dann antwortete es auf eine ziemlich befremdende, irrelevante Art und sagte Sachen wie: Ich traue dir nicht. Nach einer Weile sagte es: Ich rede einfach nicht mehr mit dir. Dann sagte es gar nichts mehr."

Das Programm war, sachlich betrachtet, sehr primitiv, aber die Psychiater, berichtet Minsky, waren beeindruckt von den Leistungen der Forschungsgruppe Künstliche Intelligenz. Und das alles ganz ohne komplizierte neuronale Netze.

Das bekannteste Doktor-Programm stammt aus dem Jahre 1966: Joseph Weizenbaums Sprachanalyse-Programm Eliza, das nach der Schülerin in Shaws Pygmalion-Geschichte benannt wurde, weil es genauso begriffslos "gebildete" Gespräche führt, indem es die Stichworte eines menschlichen Dialogpartners übernimmt und variiert.

Als Weizenbaum feststellte, daß das Eliza-Programm ernst genommen wurde, schrieb er seine bekannte Abrechnung mit dem Titel "Die Macht der Computer und die Ohnmacht der Vernunft". Der mittlerweile emeritierte MIT-Professor sagt heute: "Eliza ist Public Domain. Normalerweise will ich nichts mehr damit zu tun haben, wenn ich darauf angesprochen werde, aber es gehört durchaus zur Geschichte."

Ein relativ neues Spiel mit der Ästhetik eines verselbständigten Computers sind Screensaver. Der Bildschirm, erzählen gut informierte Softwareverkäufer, geht kaputt, wenn die Leuchtpunkte der Bildröhre nicht ab und zu neu verteilt werden: Das Bild brennt ein. Eigentlich würde es ja ausreichen, bei längeren Pausen die Bildschirmhelligkeit herunterzudrehen. Aber wieso sollte man es sich so einfach machen, wenn einem acht Megabyte Speicher zur Verfügung stehen, von denen schon ein Bruchteil für die bloße Arbeit ausreichen würde? Screensaver sind sinnlos und schön.

Die "After Dark"-Screensaver (Berkely-Systems, 1989/90) für den Macintosh bewegen den Computer zu überraschenden Bildern, sobald der Computeranwender, wie das so korrekt wie bürokratisch heißt, die Computeranwendung vernachlässigt und nur nachdenklich vor der Kiste sitzt. Bei "Fish", verwandelt sich der Bildschirm freundlicherweise in ein Aquarium: Fische, Quallen, Seepferdchen schwimmen schwerelos über den Monitor. "Puzzle" zerlegt das Monitor-Bild zuerst in Würfel und schiebt die so lange hin und her, bis die grafische Oberfläche völlig sinnlos wird.

Solche die Bildschirmordnung auflösenden Automatismen beziehen ihren Reiz aus der oberflächlichen Ähnlichkeit mit Virenprogrammen, die jedoch leider, leider manchmal auch unter der grafischen Oberfläche Daten durcheinanderbringen. Viren sind Programme, die sich in vorhandene Wirtsprogramme hineinkopieren und bei Aufruf dieser Programme tätig werden, manchmal nur harmlos ein lustiges Liedchen dudeln ("Yankee-Doodle-Virus" z.B.), schlimmstenfalls den ganzen Inhalt der Festplatte zerstören. Viren sind der Horror jedes Kleineigentümers, der eigenhändig mehr als genug Scheine für seine Software hingeblättert hat und sich seine Daten nicht von irgendwelchen karriereverhindernden Hackern kaputt machen lassen will. Aber die Softwarehersteller haben schon das garantiert wirksame Antiviren-Rezept parat: Raubkopien meiden! Vorsicht vor allem bei Spielen und Kopien aus Unirechnern! Nur lizenzierte Originalsoftware auf den persönlichen Rechner laden! Um ihre Empfehlung zu bekräftigen, legen die meisten Hersteller bei Großeinkäufen auch einige (pädagogisch wertvolle) Spiele in das multifunktionale Bürosoftware-Paket.

Die ganze Hingabe der Viren-Hacker an ihre Mission, "das System" durcheinanderzubringen, zeigt sich in den Grafiken, die in Aktion treten, sobald die Festplatte infiziert ist: Es wird eine poetische Ästhetik des Verschwindens inszeniert — die klammheimliche Freude des Querulanten an der Zerstörung der Ordnung. Romantiker sprechen auch vom Biotop-Prinzip, denn sobald die Computerviren von ihrem Schöpfer in Umlauf gebracht wurden, pflanzen sie sich, wie ihre Verwandten aus der Biologie, geschlechtslos fort bis in die Ewigkeit — oder zumindest bis sie von einem der vielen ausgefeilten Antivirenprogramme entdeckt werden.

Bei fast allen Virenprogrammen bleiben die Daten, die auf dem Bildschirm sichtbar festgehalten wurden, zunächst erhalten; sie werden vor ihrem endgültigen Verschwinden erst einmal nur etwas verändert: "Zeroeat" frißt alle Nullen weg, "Drop" läßt Zeichen langsam, unspektakulär, ganz nonchalant aus ihrem Zusammenhang in irgendwelche darunterliegende Zeilen fallen. "Joker" macht alles bunt und teilt bei Dateiwechsel eine diabolische Message mit: "Have you ever danced with the devil under the weak light of the moon? Pray for your disk!" "Whirl" läßt die auf dem Monitor sichtbaren Zeichen wie ein Wirbelwind über die dunkel gewordene Bildfläche strudeln, bevor sie ins Schwarze Loch der Bits und Bytes stürzen.

Software, die Software verschwinden läßt

Software, so wie sie sich heute präsentiert, hat sich im Spannungsfeld zwischen Arbeit, Maschine und Spiel entwickelt. Computerspiele werden immer aufwendiger gestaltet, Kinofilmen immer ähnlicher, während die interaktive Oberfläche von Bürosoftware immer mehr reduziert und vereinfacht wird.

In beiden Bereichen zeigt sich die Tendenz, komplexe Steuerungsprozesse ins Innere der Maschine zu integrieren, Technik aus dem Blickfeld verschwinden zu lassen. Komplizierte Fuzzy-Logik-Bauteile zum Beispiel dienen nur dazu, die Tokioter U-Bahnbenutzer gerade die technischen Details einer Zugfahrt möglichst vergessen zu lassen.

Auch in der Wissenschaft werden die bisherigen Methoden, einen Ausschnitt der Wirklichkeit modellhaft zu rekonstruieren, ersetzt von neuen Methoden, die eben, unter anderem, neue Computersysteme zur Hilfe nehmen. Das Arbeitsmittel ist neu. Der Methodenwechsel ist nichts Neues. Er stützt sich nur auf neue Hilfsmittel.

In der Ausstellung "Kreative Software — Menschen und Meilensteine" sind unter anderem folgende Programme zum Ausprobieren, Spielen und Experimentieren bereitgestellt:

VisiCalc, Bravo, Lisa Office System, dPaint Amiga, Basic, Logo, PacMan, Star, Collosal Cave, Eliza, After Dark, (entschärfte) Viren-Programme, SimCity, Life und das Multimedia-Spiel Piazza Virtuale, das von der Kasseler documenta aus inszeniert wird.

Eva Weber, Klaus Madzia

Konzept: Benjamin Heidersberger

Ausstellung: Bernd von den Brincken

Leihgaben der Ausstellung:

Boston Computer Museum (Brian Wallace und David Greschler), Benjamin Heidersberger, Legend Entertainment Company (Bob Bates), Thomas Nitsche, Joseph Weizenbaum, Christian Wolff, Rank Xerox Deutschland (M. N. Foster), Klaus Schiwinsky, Atari Österreich, Hard & Soft Wien (Next), SD Computer Hamburg, Apple München, ProMo Hamburg

