

Das Epos von Silicon und die Barden des Binären

Zur Bestimmung der korrekten Reichweite des urheberrechtlichen Schutzes für Computer-Software

Anthony L. Clapes/Patrick Lynch/Mark R. Steinberg

Teil 2

III. Die Entscheidung der Kontroverse

A. Wie Programmierer sich ausdrücken

1. „Autoren-Werke“ im Unterschied zu „Ergebnissen“
2. Der Kontext, in dem Programme geschrieben werden
 - a. Was Computer „verstehen“
 - i. Prozessor-Funktion
 - ii. Speicherung
 - iii. Eingabe und Ausgabe
 - iv. Kontrolle
 - v. Kommunikation

III. Die Entscheidung der Kontroverse

Um zu entscheiden, welche Reichweite an Schutz das Urheberrecht (richtig verstanden) Computerprogrammen einräumt, ist es notwendig, zu verstehen, um welche Ausdrucksformen es sich bei Computerprogrammen handelt, und dann zu überlegen, wie das Urheberrecht auf diese Ausdrucksformen anwendbar ist. Will man dieses Verständnis wenigstens auf einem allgemeinen Niveau erarbeiten, so erfordert dies ganz einfach eine Untersuchung der Art und Weise, wie Programmierer sich in ihren Programmen ausdrücken, und eine Analyse des Vorgangs, der sich beim Schreiben dieser Programme abspielt. Dieser Untersuchung und Analyse wenden wir uns nun zu.

A. Wie Programmierer sich ausdrücken

1. „Autoren-Werke“ im Unterschied zu „Ergebnissen“

Es gibt zwei fundamentale Arten darüber nachzudenken, was Computerprogramme sind. Die eine Art und Weise repräsentiert die Wahrnehmung des typischen Kunden. Ein typischer Kunde nimmt Computerprogramme als etwas wahr, das Computer dazu veranlaßt, sich in einer bestimmten Weise zu verhalten. In einem gewissen Sinne ähnelt diese Perspektive dem Eindruck, den ein Konzertbesucher von einer Symphonie gewinnt. Für den Konzertbesucher ist die Symphonie die Tonmenge, die er hört, eine ästhetische Erfahrung. Trotzdem weiß der Konzertbesucher auch, daß die Symphonie, so wie sie von ihrem Komponisten geschaffen wurde, tatsächlich eine Abfolge von Instruktionen ist, die von einem Instrumentarium

verarbeitet werden müssen, dem Orchester nämlich, soll die „Symphonie“ entstehen, die der Zuhörer hört. Die zweite Art und Weise, über Computerprogramme zu denken (oder auch über Symphonien) besteht darin, sie als Schriftwerke zu betrachten. Programme (wie andere literarische Werke auch) bestehen aus Zeilen von Text, die aus symbolischen Buchstaben zusammengesetzt sind. Das ist die Wahrnehmung des Programmierers, denn das ist die Art und Weise, in der er ein Programm schreibt oder liest. Um auf unsere Konzert-Analogie zurückzukommen: Die Sicht des Programmierers würde der des Komponisten entsprechen. Die Ausdrucksform eines Programms kann aus beiden Blickrichtungen wahrgenommen werden, aber mehr von ihr wird wahrgenommen, wenn Programme als Schriftwerke betrachtet werden⁴⁹. In diesem Abschnitt laden wir den Leser dazu ein, eine Erfahrung mit der Sicht zu machen, die der Programmierer von seiner Kunst hat. Zuerst werden wir untersuchen, ob die Beschränkungen, die der Computer selbst der programmbezogenen Autorentätigkeit auferlegt, so sehr einschränkend sind, wie dies einige Kommentatoren annehmen. Dann werden wir einen Kurs der Elemente des Programmierstils absolvieren, im Anschluß an den wir untersuchen wollen, was Experten zur Bandbreite der Programmier-Ausdrucksmöglichkeiten gesagt haben. Wir beschließen unseren Ausflug mit einem Überblick über den Prozeß des „Programme-Schreibens“ und einer Betrachtung der Frage, was dieser Prozeß hinsichtlich der Bandbreite der Programmier-Ausdrucksmöglichkeiten anzunehmen nahelegt.

2. Der Kontext, in dem Programme geschrieben werden

Verschiedentlich wird behauptet, daß Programme sich von anderen literarischen Werken unterscheiden, weil nicht beabsichtigt ist, daß sie durch Menschen gelesen werden. Diese Behauptung ist falsch. Programme, die für die gewerbliche Verbreitung geschrieben werden, werden fast immer mit dem Gedan-

49) Es gibt natürlich Programme (wie z.B. Computerspiele und andere hochinteraktive Programme), bei denen ein großer Teil des Ausdrucks in der audio-visuellen Darbietung liegt, zu deren Präsentation das Programm den Computer instruiert. Der vorliegende Artikel behandelt nicht die Fragen, die mit der Ausdrucksqualität derartiger audio-visueller Darstellungen zusammenhängen, da die Anwendung traditioneller Urheberrechtsprinzipien auf diese Frage leichter verständlich ist. Obwohl es in letzter Zeit in der Computerindustrie eine gewisse Kontroverse über die Frage der Haftung derjenigen gegeben hat, die das „look and feel“ der audio-visuellen Anzeigen von bildschirmorientierten Computerprogrammen kopieren (vgl. z.B. Klageschrift, Lotus Development Corp. v. Mosaic Software, Inc., No. 870074k (D. Mass 1987); Klageschrift, Lotus Development Corp. v. Paperback Software Int'l, No. 87-0076K (D. Mass 1987); vgl. auch Sanger, A Divisive Lotus 'Clone War', N.Y. Times, 5. Februar 1987, bei D1, Sp. 3), ist der solchen audio-visuellen Darstellungen eingeräumte Schutzbereich präzise beschrieben. Er verändert sich auch nicht in Abhängigkeit davon, ob die Quelle derartiger Displays Befehle eines Computerprogramms, ein Film, ein Videoband oder andere körperliche Ausdrucksmedien sind. Vgl. Broderbund Software, Inc. v. Unison World, Inc., 648 F. (N.D. Cal. 1986).

ken geschrieben, daß sie von Menschen gelesen werden. Wir sprechen selbstverständlich nicht über Bettlektüre (vielleicht mit Ausnahme der hartgesotenen „Byteniks“). Aber wir sprechen über die Tatsache, daß Programme – sollen sie modifiziert, erweitert oder korrigiert werden – der Lektüre des menschlichen Lesers zugänglich sein müssen.⁵⁰

Natürlich wird das Programm schließlich auf eine Form gebracht, die von einem Computer gelesen werden kann. Aber sogar in maschinenlesbarer Form kann das Programm nur von Menschen gelesen und verstanden werden, obwohl diese Form von allen Versionen des Programms die für Menschen am schwierigsten zu lesende und zu verwendende ist.⁵¹ Damit Programme sowohl maschinenlesbar als auch für Menschen lesbar sind, muß sich der Programmierer an besondere Arten syntaktischer, semantischer und konstruktiver Konventionen halten. Diese Konventionen beeinflussen die Art des Ausdrucks, aber (wie wir sehen werden) nicht die Bandbreite der Ausdrucksmöglichkeiten für Computerprogramme.

a. Was Computer „verstehen“

Im gegenwärtigen Kontext sind „Computer“ nichts anderes als aus Stahl, Silicon, Plastik und anderen physikalischen Komponenten hergestellte Maschinen. Konzeptuell sind diese Maschinen nur Mengen von Elementen, die – ebenso wie Glühbirnen – in einen „angeschalteten“ oder „ausgeschalteten“ Zustand versetzt werden können. Sie werden zum Leben erweckt durch Schalter, entweder automatisch als Reaktion auf eine Veränderung in einem anderen Element oder auf die Anweisung eines intelligenten Operators hin. Solch eine Maschine kann nichts „verstehen“. Sie reagiert nur in Übereinstimmung mit den Prinzipien der Physik auf die Aktionen der Operatoren, die sich auf ihre verschiedenen Schaltungen beziehen. Die „Lichter“, die die Zustände „an“ und „aus“ annehmen, haben Bedeutung nur für Menschen, die einen Code entworfen haben, der in Abfolgen der Zustände „an“ und „aus“ ausgedrückt werden kann, einen Code, in dem bekannte elektrische Reaktionen in exakter Weise der großen Anzahl elementarer Instruktionen entsprechen.

Damit diese Maschine irgend etwas Nützliches tut, muß jemand die Schalter bedienen. Dies könnte, Schalter für Schalter, von Hand geschehen; in der frühesten Zeit der elektronischen Computer war das auch so.⁵² Mit der Zeit jedoch wurden Methoden entwickelt, um die Schalter durch „Kontrolle aus der Entfernung“ zu bedienen. Eine Methode, die durch den Jacquard-Webstuhl im nachrevolutionären Frankreich veranschaulicht wird, besteht darin, Lochkarten zu verwenden. In dem Jacquard-Webstuhl fielen ausgewählte Stifte in die Löcher in den Karten und schalteten dadurch das Muster des Webstuhls. Ein wesentlich ausgeklügelterer Ansatz besteht darin, elektronische Signale zu verwenden, Zustände von „an“ und „aus“. Hier instruiert man die Hardware, die Schalter nach einem vorher festgelegten Plan an- oder auszuschalten und, wie sich zeigt, dies mit elektronischer Geschwindigkeit.

Die Anwendung der zuletzt genannten Technik ist der Hardware-Unterbau der Kunst des Programmierens und die Basis für moderne Computersysteme. Die Mächtigkeit dieser Technik ist so groß, daß kein Computer mit einem allgemeinen Zweck nur aus Hardware besteht. Alle Computer bestehen aus (1) Hardware, die die Zustände des „an“ und „aus“, die für

Information oder Daten stehen, speichert und manipuliert, (2) Hardware, die die entscheidenden Schalter des Computers auf bestimmte Art und Weise kontrolliert, und zwar als Reaktion auf einen Satz festgelegter Muster von „an“ und „aus“ (man spricht vom „Anweisungssatz“ des Computers), und (3) ausgearbeiteten Instruktions-Mengen (Programmen), die den Computer zur Ausführung von Aktionen veranlassen, die Menschen in wahrnehmbare und nützliche Resultate übersetzen können.⁵³ Diese drei konstituierenden Elemente eines Computers oder, korrekter ausgedrückt, eines Computersystems, werden nach der folgenden Taxonomie funktional organisiert:

i. Prozessor-Funktion

Der Hardwareteil, der die Programme „liest“, ist der Prozessor. Die Aufgabe des Prozessors besteht darin, in Übereinstimmung mit den Programm-Instruktionen über Daten zu operieren.⁵⁴ Wichtige Bestandteile des Prozessors sind:

- a. ein Pointer, der auf die Adresse der Instruktion zeigt, die als nächste zu verarbeiten ist,
- b. eine Schaltung für die Interpretation der Instruktionen,
- c. Speicherplätze (Register) für die Aufbewahrung der Daten, über die operiert werden soll, und
- d. ein Schaltkreis für die Ausführung der arithmetischen oder logischen Operationen über die Daten, auf die aufgrund der Instruktion zugegriffen wird.⁵⁵

Von bescheidenen Ausnahmen abgesehen, kann die heutige Prozessor-Hardware nur (obwohl das sehr schnell) eine elementare Instruktion in einem Zeitpunkt ausführen, so wie beispielsweise das Addieren zweier Zahlen oder einen Vergleich zweier Zahlen.⁵⁶ Sie kann nur solche elementaren Operationen ausführen, die in ihren Schaltkreis eingebaut sind.⁵⁷ Für jeden Computer entspricht die Menge derartiger Operationen einer Serie elektronischer Signale, auf die wir uns bereits als den „Instruktionssatz“ eines Computers bezogen haben. Eine die Ausdrucksmöglichkeiten bestimmende Regel für Computer ist also die, daß Instruktionen an den Prozessor als Kombination der elementaren Operationen seines Schaltkreises adressiert werden müssen.⁵⁸ Eine andere Regel für die Ausdrucks-

50) R. Linger, H. Mills & B. Witt, *Structured Programming: Theory and Practice* 147 – 48 (1979). Vgl. auch J. Sammet (a.a.O., Fn. 41) bei 14 – 17.

51) Ironischerweise räumen die Befürworter eines engen Schutzbereichs bereitwillig ein, daß die in striktem Sinne maschinenlesbaren Sequenzen eines Programms durch das Urheberrecht geschützt sind (und zwar als das einzige, das geschützt wird), während die Elemente des Programms, die die Übersetzung in eine höhere, eher für Menschen einsehbare Ausdrucksform überleben, dieser Ansicht nach nicht schützbar sind.

52) Vgl. unten die Anmerkungen 82 bis 84 und den Text dazu.

53) C. Gear, *Introduction to Computer Science* 49 – 50 (1973); M. Mandl, *Fundamentals of Electronic Computers: Digital and Analog*, 6 – 8 (1967).

54) *The MacGraw-Hill Computer Handbook* §§ 2 – 8 (hrsg. v. H. Helms, 1983).

55) a.a.O. bei §§ 2 – 8, 6 – 16; vgl. auch A. Ralston, *Encyclopedia of Computer Science and Engineering*, 102 – 106 (2. Aufl. 1983).

56) *The MacGraw-Hill Computer Handbook* (a.a.O., Fn. 54), §§ 6 – 16. Die Ausnahme, auf die im Text Bezug genommen wird, ist der sogenannte „Parallelprozessor“, der zur Zeit noch kein wichtiger Wirtschaftsfaktor ist, der aber in Zukunft der Aufgabe des Programme-Schreibens eine ganz neue Dimension von Komplexität und Flexibilität hinzufügen wird. Vgl. *Parallel Processing Systems* (hrsg. v. D. Evans, 1982).

57) *The MacGraw-Hill Computer Handbook* (a.a.O., Fn. 54), §§ 7 – 16.

58) a.a.O.

möglichkeiten, die Programmierern zur Verfügung stehen, ergibt sich daraus, daß der Prozessor nur Zustände von „an“ und „aus“ bearbeiten kann.⁵⁹ Diese Begrenzung zwang Programmierer in den frühen Zeiten der Industrie, eine „binäre“ Notation zu verwenden, d.h. ein System, in dem alle Buchstaben durch Kombinationen der Elemente „0“ und „1“ ausgedrückt wurden.⁶⁰ Binärziffern verhalten sich ziemlich genau wie die Ziffern des Dezimalsystems, obwohl sie weniger kompakt sind. Beispielsweise würde die Dezimalzahl 37 binär geschrieben als 00100110. Die binäre Notation kann auch benutzt werden, um Buchstaben darzustellen.

In vielen Computern wird die binäre Information in Blöcke eingeteilt, wobei jeder Block einen einzelnen Buchstaben in einer anderen Sprache repräsentieren kann, wie z.B. die Buchstaben des Alphabets oder das Symbol für eine arabische Ziffer. In diesem Sinne verwenden viele Computer 8 „Bits“ (binary digits, d.h. Nullen oder Einsen) um Buchstaben darzustellen. Mit 8 Bit kann man 256 Buchstaben darstellen, die das gesamte englische Alphabet mit Groß- und Kleinbuchstaben umfassen können, sowie die Ziffer 0 bis 9 und viele andere Symbole. Dieser Block von 8 Elementen wird 1 „Byte“ von Information genannt. Es handelt sich dabei um eine allgemeine Art und Weise, mit der man binäre Information in Computern organisiert.⁶¹

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	DLE		0	@	P	p	Ç	É	ä	⋮	L	⋮	α	≡	
1	SOH	DC1	!	1	A	Q	a	q	û	œ	ï	⋮	⋮	β	±	
2	STX	DC2	"	2	B	R	b	r	é	/E	ó	⋮	⋮	Γ	≧	
3	ETX	DC3	#	3	C	S	c	s	ä	ô	û		⋮	Π	≦	
4	EOT	DC4	\$	4	D	T	d	t	ä	ó	ñ	⋮	⋮	Σ	Γ	
5	ENQ		%	5	E	U	e	u	ä	ó	Ñ	⋮	⋮	σ	J	
6	ACK	SYN	&	6	F	V	f	v	ä	û	ä	⋮	⋮	μ	+	
7	BEL	ETB	'	7	G	W	g	w	ç	ú	o	⋮	⋮	τ	≈	
8	BS	CAN	(8	H	X	h	x	ê	ÿ	¿	⋮	⋮	φ	°	
9	HT	EM)	9	I	Y	i	y	ë	Ö	⋮	⋮	⋮	θ	•	
A	LF	SUB	*	:	J	Z	j	z	è	Ü	⋮	⋮	⋮	Ω	•	
B	VT	ESC	+	;	K	[k	{	ï	ç	½	⋮	⋮	δ	√	
C	FF	FS	,	<	L	\	l		í	£	¼	⋮	⋮	∞	η	
D	CR	GS	-	=	M]	m	}	ï	¥	ï	⋮	⋮	∅	2	
E	SO	RS	.	>	N	^	n	~	Ä	R	«	⋮	⋮	ε	■	
F	SI	US	/	?	O	_	o	DEL	À	f	»	⋮	⋮	∩		

Der erweiterte ASCII-Zeichensatz.

Binäre Arithmetik hat mit der formalen Logik viel gemeinsam.⁶² Das Format der beiden Zustände 1 und 0 entspricht dem Wahr- Falsch-Format der formalen Logik. Im Ergebnis bietet sich deswegen das binäre System nicht nur für die Berechnung mathematischer Probleme an, sondern auch für die Zerlegung logischer Probleme. Beispielsweise könnte ein Computer instruiert werden, das folgende zu tun:

Wenn es wahr ist, daß ein Fall das Wort „Copyright“ erwähnt, und es wahr ist, daß er entweder „Computerprogramm“ oder „Software“ erwähnt, und es wahr ist, daß er nach 1982 entschieden wurde, dann soll der Name dieses Falles ausgedrückt werden.

Wie bereits angedeutet, gibt es elektronische Schaltkreiselemente, die in genau derselben Weise wie die binäre Arithmetik funktionieren.⁶³ Das bedeutet, daß sie nur dazu in der Lage sind, sich in einem der beiden Zustände „an“ oder „aus“ zu befinden. Ein Schaltkreis, der aus solchen Elementen gebaut werden kann, ist der „und“-Schaltkreis. Ein einfacher „und“-Schaltkreis hat zwei Eingaben und eine Ausgabe. In diesem Schaltkreis fließt Strom an der Ausgabe nur (die Bedeutung ist dann binär 1), wenn Strom an beiden Eingabestellen anliegt. Ein anderer solcher Schaltkreis ist der „oder“-Schaltkreis, aus dem Strom fließt, wenn Strom an mindestens einer der beiden Eingabestellen anliegt. Aus solchen elementaren Schaltkreisen kann ein einfacher Addierer konstruiert werden, in dem die Eingaben 1 und 0 die Ausgabe 01 ergeben, die Eingaben 1 und 1 die Ausgabe 10 und die Eingaben 0 und 0 die Ausgabe 00.

Durch die Verbindung großer Mengen dieser elementaren Schaltkreise, wurden leistungsfähige Computer geschaffen, die in der Lage sind, Millionen von Instruktionen pro Sekunde auszuführen.⁶⁴ Schaltet man die Verbindungen zwischen diesen Schaltkreisen mit Hilfe von Software, so ergibt sich, daß die Bandbreite der Probleme, für deren Lösung der Computer eingesetzt werden kann, tatsächlich grenzenlos ist.

ii. Speicherung

Genauso wie es Schaltkreise gibt, die in der Lage sind, logische Operationen auszuführen, so gibt es auch Schaltkreise, die 1 Bit an Information (eine 1 oder eine 0) speichern, über den Speicherinhalt Auskunft geben, wenn eine entsprechende Abfrage erfolgt, und auf Anweisung den Inhalt des Speichers verändern.⁶⁵ Eine große Anzahl derartiger Schaltkreise kann auf einem Halbleiterchip erzeugt werden. 1988 konnten die industriell breit verfügbaren Speicherchips 16000 Bits an Information festhalten.⁶⁶ Heute sind Chips mit einer Million Bits (Megabit) weit verfügbar und der 16-Megabit-Chip wird in der Handelspresse diskutiert.⁶⁷ Für unsere Zwecke kann die Speicherung in einem Prozessor konzeptualisiert werden als Sammlung von Briefkästen oder Zellen, die je ein Bit an Information halten können.⁶⁸ Bei dieser Information kann es sich um Daten oder um den Teil einer Instruktion handeln. Jede Zelle hat eine mit ihr verbundene Adresse, und der Prozessor sendet Information zu der Zelle oder holt Information aus der Zelle, indem er die Adresse in Bezug nimmt.

59) a.a.O. bei §§ 1 - 7.
 60) A. Gillie, Binary Arithmetic and Boolean Algebra 1 - 14 (1965).
 61) M. Weik, Standard Dictionary of Computers and Information Processing 49 (2. Aufl., 1977).
 62) S. Adelfio & C. Nolan, Principles and Applications of Boolean Algebra 51 (1964).
 63) The MacGraw-Hill Computer Handbook (a.a.O., Fn. 54), § 5.2; S. Adelfio & C. Nolan (a.a.O., Fn. 62), bei 206 - 15.
 64) A. Ralston (a.a.O., Fn. 55), bei 103, 1127 - 31.
 65) The MacGraw-Hill Computer Handbook (a.a.O., Fn. 54), § 7.3.
 66) W. H. Davidson (a.a.O., Fn. 12), bei 107.
 67) Vgl. „16 Meg - A Test Vehicle for Now, Electronic News, 2. März 1987, bei 36.
 68) M. Mandl (a.a.O., Fn. 53), bei 164. Praktisch ist gegenwärtig die Information, die in den Zellen enthalten ist, nur in einem String wiedergewinnbar, z.B. als ein Byte, und nicht in einzelnen Bits. The MacGraw-Hill Computer Handbook (a.a.O., Fn. 54), bei 2 - 4.

Die Speicherung im Prozessor ist eine kurzfristige Speicherung für Daten oder Instruktionen, die unmittelbar vom Prozessor benötigt werden.⁶⁹ Permanente Langzeitspeicherung wird im allgemeinen durch Plattenlaufwerke, Bandlaufwerke oder andere Speichermedien gewährleistet, die Information magnetisch speichern oder aber, um auf eine aufregende neue Technologie Bezug zu nehmen, dies in optischer Form mit Hilfe von Punkten tun, die auf einer Oberfläche durch Laserlicht erzeugt werden.⁷⁰ Diese Speichermedien bieten im allgemeinen eine wesentlich größere Speicherkapazität als der Prozessor an, sie können aber gespeicherte Daten nur wesentlich langsamer wiedergewinnen, als dies der Prozessor tut. Eine Aufgabe des Programmatoren besteht darin, die Probleme zu lösen, die durch solche Geschwindigkeitsunterschiede verursacht werden.⁷¹

iii. Eingabe und Ausgabe

Die Information, die von einem Computersystem verarbeitet wird, muß im Ursprung von irgendeinem Punkt her kommen und schließlich zu einem anderen Punkt weitergeschickt werden. An der Eingabeseite kann die Information in einer oder mehreren verschiedenen Formen zur Verfügung stehen. Daten, Text, Bilder, Stimmen und Graphik sind fünf allgemeine Typen von Information, die als „input“ für einen Computer verfügbar sind.⁷² Die Komponente des Computers, die die Eingabeaufgabe übernimmt, kann gleichfalls in verschiedenen Formen vorhanden sein: Kartenleser, Bandlaufwerke, Festplatten oder Diskettenlaufwerke, Lochstreifenleser, Tastaturen an Bildschirmgeräten, optische oder magnetische Buchstabenlesegeräte oder Spracherkennungsgeräte.⁷³ In zunehmendem Maße dienen Computer als Eingabebeugeräte für andere Computer.⁷⁴ Das aus der Verarbeitung resultierende Ergebnis kann auf Band oder auf Festplatte abgespeichert werden. Es kann auch auf einem Drucker ausgedruckt oder auf einem Bildschirm angezeigt werden, durch eine Sprachausgabereinheit über das Telefon gesprochen werden, als Löcher in Karten oder Papierstreifen gestanzt werden, durch einen Plotter gezeichnet werden oder direkt von einem Computer zum anderen gesendet werden.⁷⁵ Jede Art von Eingabe- und Ausgabebeugerät hat ihre spezifische Art und Weise, den Prozessor zu adressieren oder von ihm adressiert zu werden. Die Fähigkeit, mit solchen Geräten zu kommunizieren, wird für den Prozessor im allgemeinen durch Software zur Verfügung gestellt. Eine der Aufgaben, die Autoren von Betriebssystemen typischerweise zu erledigen haben, besteht darin, die Eingabe und Ausgabe von Information zwischen dem Prozessor und der Fülle von Eingabe- und Ausgabebeugeräten mit ihren je verschiedenen Operationsanforderungen zu erleichtern.⁷⁶

iv. Kontrolle

Computersysteme bestehen aus einer Vielzahl von Elementen. Die Interaktionen zwischen diesen Elementen müssen kontrolliert werden. In der Mehrzahl der Fälle ist es die Aufgabe der Autoren von Software, die Elemente des Computers zu kontrollieren.⁷⁷ Ihre kreative Energie transformiert die Hardware-Welt der elementaren, auf Einsen und Nullen bezogenen Operationen, in eine koordinierte Menge von Produkten, die riesige Bibliotheken von Information speichern und wiedergewinnen können, Nachrichten zu Tausenden von Terminals übertragen und es den Benutzern dieser Tausenden von Terminals erlauben, die Leistung desselben zentralen Prozessors zur

selben Zeit zu nutzen. „Kontrolle“ bezeichnet in diesem Sinne die imaginative Ausarbeitung von Instruktionen, die den Befehlsumfang eines gegebenen Computers verwenden und auf dieser begrenzten Basis eine komplexe, fein-artikulierte Umgebung komponieren, die wegen der Geschwindigkeit des Prozessorschaltkreises in nahezu geheimnisvoll verschiedenartiger Weise auf die unterschiedlichen Benutzereingaben reagiert.

v. Kommunikation

Computersysteme und Teile von Computersystemen sind in der Lage, miteinander über Telekommunikationsverbindungen zu kommunizieren.⁷⁸ Der PC auf dem Schreibtisch eines Rechtsanwaltes kann Rechtsdatenbanken in einem großen Computer in einer Entfernung von Hunderten von Meilen abfragen. Die Computer eines internationalen Unternehmens können rund um den Globus Informationen untereinander über Satelliten austauschen.⁷⁹ Die verschiedenen Geldautomaten in einer Bank können mit einem Prozessor über Telefonverbindungen oder andere Kabelsysteme kommunizieren, indem sie ein „lokales Netzwerk“ oder ein „Datenkommunikationsnetzwerk“ bilden.⁸⁰ Um den Erfolg solcher Netzwerke sicherzustellen, muß es Programme geben, die die Computer so instruieren, daß sie sich in ganz ähnlicher Weise wie ein Postamt oder eine zentrale Telefonvermittlung verhalten.⁸¹

69) Anders ausgedrückt: die Daten und Instruktionen im Speicher des Prozessors können sich im Verlaufe der Verarbeitung ständig verändern; vgl. The MacGraw-Hill Computer Handbook (a.a.O., Fn. 54), bei 2 - 4. Wie alle Generalisierungen, so hat auch diese ihre Schwächen. Beispielsweise enthalten viele Computer einige permanente Speicher in der Form von ROM's, in denen bestimmte Programmierschritte fest oder halb-fest gespeichert sind. A. Ralston (a.a.O., Fn. 55), bei 1264 - 65.

70) A. Ralston (a.a.O., Fn. 55), bei 965 - 67. Vgl. auch J. Hecht & D. Terisi, Laser - Supertool of the 1980's, 207 - 208 (1982).

71) Vgl. S. Kurzban, T. Heines & A. Sayers, Operating Systems Principles, 93 - 107 (1975).

72) A. Ralston (a.a.O., Fn. 55), bei 736 - 38.

73) a.a.O., bei 739 - 66.

74) a.a.O., bei 647 („front end processors“). Vgl. auch D. Charofas, Personal Computers and Data Communications, 299 (1986) („die erfolgreiche Implementation von intelligenten, multifunktionalen, interaktiven Workstations ist gleichbedeutend mit ihrer Online-Verbindung zu Großrechner-Ressourcen, hauptsächlich großen Datenbanken.“).

75) A. Ralston (a.a.O., Fn. 55), bei 739 - 66.

76) S. Kurzban, T. Heines & A. Sayers (a.a.O., Fn. 71), bei 93 - 107.

77) Vgl. M. Mandl (a.a.O., Fn. 53) bei 241 („auf diese Weise gibt ein Programm der Maschine an, was sie tun soll, wie sie es tun soll und wo sie die Daten findet, die während der Ausführung des Programms abgearbeitet werden sollen.“).

78) Computer Network Architectures and Protocols 3 (hrsg. v. P. Green, 1982).

79) Vgl. West Publishing Co. v. Mead Data Cent., Inc., 616 F. Supp. 1571 (D. Minn. 1985), cert. denied, 107 S. Ct. 962 (1987); W. Stallings, Data and Computer Communications, 299 - 319 (1985).

80) „Lokale Netzwerke“ sind Kommunikationssysteme, die Geräte miteinander verbinden, die sich in geringer Entfernung voneinander (z.B. einigen Kilometern oder weniger) befinden. Es mag sein, daß sie hauptsächlich Kabelsysteme verwenden, die sich im Privateigentum befinden. Vgl. Computer Network Architectures and Protocols (a.a.O., Fn. 78, bei 148).

„Datenkommunikationsnetzwerke“ sind Kommunikationssysteme, die Geräte miteinander verbinden, die in weiter Entfernung voneinander (d.h. bis zu tausenden von Meilen) installiert sind. Sie benutzen gewöhnlich verteilte, geschaltete Netzwerke oder Breitbandnetzwerke, hauptsächlich jedoch die öffentlichen, geschalteten Netzwerke.

81) Vgl. U. Black, Data Communications, Networks, and Distributed Processing, 121 - 35 (1983).