

"Survival Skript"

Zur Vorlesung:

Einführung in die Wirtschaftsinformatik WS00/01 (bei Prof. Dr. J. Ruhland)

-Teil 1-

Hinweis: Dieses Skript wurde privat erstellt, um für die Wirtschaftsinformatik-Klausur eine Basis zu haben, und wird anderen Studierenden zur *privaten* Vorbereitung auf die Klausur ebenfalls zur Verfügung gestellt. Eine - wie auch immer geartete - kommerzielle Nutzung des "Survival-Skriptes" stellt einen Verstoß gegen die Urheberrechte der div. benützten (und nicht immer explizit ausgewiesenen) Quellen dar. Als "Privat-Papier" kann auch keine Garantie auf Vollständigkeit und/oder Fehlerfreiheit gegeben werden. Man sollte aber ruhig davon ausgehen, dass dieses Skript hier als Lerngrundlage einigermaßen zuverlässig ist. Fragen und / oder Anregungen bitte an: bast4u@gmx.net

Inhaltsverzeichnis

1. Ressourcensicht	2
1.0 Vorbemerkung.....	2
1.1 Komponenten der Hardware im Überblick	2
1.1.1 Komponenten der Zentraleinheit: Der Hauptspeicher	3
1.1.1.1 Arbeitsspeicher	3
1.1.1.2 Festwertspeicher	4
1.1.1.3 Exkurs: Bits und Bytes: Was ist das?	4
1.1.2 Komponenten der Zentraleinheit: Der (Zentral)Prozessor (CPU).....	5
1.1.3 Peripheriegeräte (Speichermedien, Ein- und Ausgabegeräte).....	6
1.1.3.1 Magnetische Speicher.....	6
1.1.3.2 Optische Speicher.....	7
1.1.3.3 Weitere Speichertypen	7
1.2 Software	8
1.2.1 Das Betriebssystem.....	8
1.2.1.1 Der Boot-Vorgang und BIOS.....	9
1.2.1.2 Dateisysteme.....	9
1.2.1.3 Das Betriebssystem an sich	10
1.2.1.4 Hinweis zur Frage 14 aus der Winfo-Übung	13
1.2.1.5. Ein paar Erläuterungen zu den Stichworten im "Ruhland Skript" Folie 27/45:	14
1.2.3 Ein paar Stichworte / Übersichten zum "Querlesen".....	14
1.2.3.1 Betriebssysteme - Marktübersicht und Trends	14
1.2.3.2. Ergänzende Systemsoftware-Komponenten	16
1.3 Vernetzung, Datenmengen, etc.	17
1.3.1 Vernetzung	17
1.3.2 Datenmengen, Konvention zur Binär-Codierung	19

1. Ressourcensicht

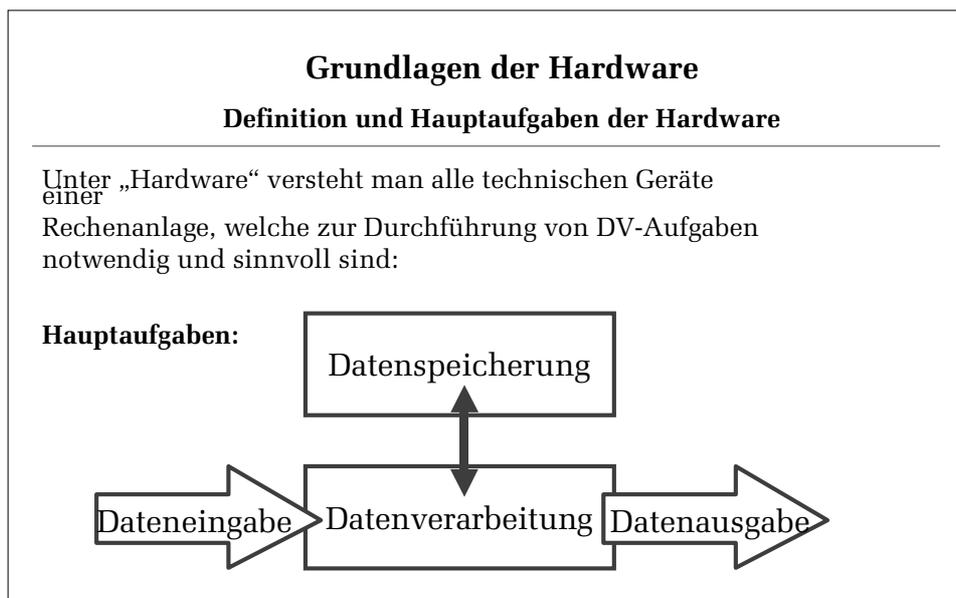
1.0 Vorbemerkung

Bei den Bestandteilen oder Elementen eines Computers wird zwischen **Hardware** und **Software** unterschieden. Man kann sich folgende Faustregel merken: **Hardware** sind alle **mechanischen und / oder elektronischen** Bauteile des Rechners. (Alles was „hart“ ist: Wenn mir ein CD-ROM Laufwerk auf den Kopf fällt, tut das weh) **Software** sind alle **Programme**, also z. B. MS Word, das Betriebssystem Windows 98 oder das Computerspiel DOOM.

Hardware (also das was oft gemeint wird, ist von einem Computer die Rede) determiniert die technisch bedingten theoretischen Möglichkeiten eines Rechners (Hatte er ein CD-ROM-Laufwerk? Wie groß ist die Festplatte? Gibt es einen Drucker?), während erst die Software den Rechner „zum Leben erweckt“, und damit die praktischen Möglichkeiten des Rechners determiniert (Der tollste Scanner nützt nix ohne ein gutes Scannprogramm). Beide Komponenten, **Hard- und Software bedingen sich gegenseitig**: Habe ich ein langsam arbeitendes Programm, hilft es mir wenig einen schnelleren Prozessor einzubauen; umgekehrt gilt aber auch, daß das tollste und neueste Programm mir nichts hilft, wenn mein Rechner hardwaremäßig eine „alte Kiste“ ist.

1.1 Komponenten der Hardware im Überblick

[Zitat] „Folie 1: Definition und Hauptaufgaben der Hardware:



Ein typischer PC-Arbeitsplatz besteht aus den folgenden Hardwarekomponenten:

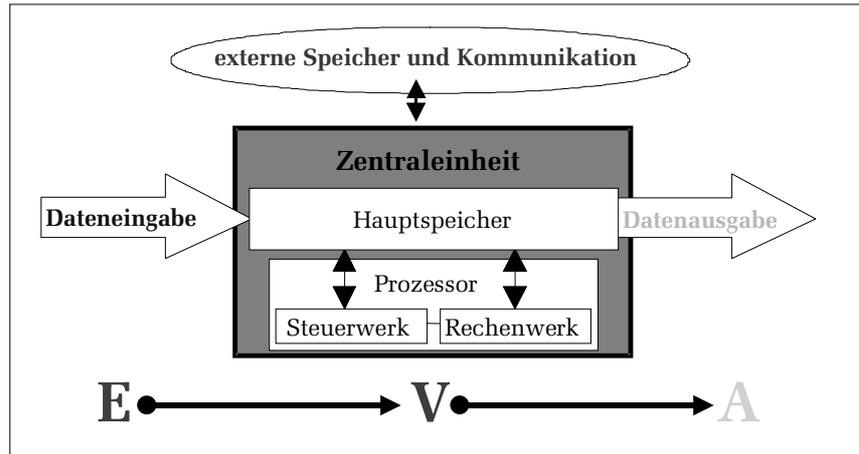
- Zentraleinheit, die sich aus **Prozessor und Hauptspeicher** zusammensetzt
- externe **Speichermedien** (z.B. Magnetplatten, Disketten)
- **Dateneingabegeräte** (z.B. Tastatur, Maus, Scanner)
- **Datenausgabegeräte** (z.B. Bildschirm, Drucker)

Darüber hinaus läßt sich ein solcher Arbeitsplatz um eine **Datenübertragungseinheit** (z.B. Netzkarte, Modem) ergänzen, wodurch ein Anschluß an ein Kommunikationsnetz ermöglicht wird.

Neben diesen stationären Arbeitsplätzen gewinnen auch tragbare PCs an Bedeutung. Verbreitet sind vor allem *Notebooks*, die in geschlossenem Zustand ungefähr dem DIN-A4-Format entsprechen.

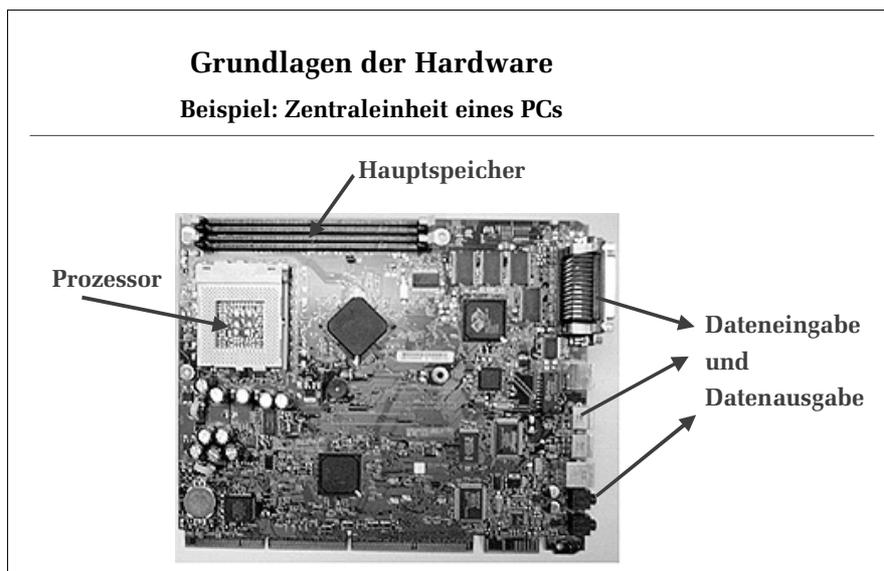
Man kann sich die Arbeitsweise eines Rechners so vorstellen, daß man zunächst Daten *eingibt*, z.B. über Tastatur oder optische Lesegeräte, oder der Rechner Daten über externe Speicher einliest. Diese Daten werden dann *verarbeitet* und anschlie-

ßend *ausgegeben*, z.B. auf dem Bildschirm, dem Drucker oder wiederum auf den externen Speichern. Diese Arbeitsweise wird als *Eingabe-Verarbeitung-Ausgabe Prinzip* (E-V-A-Prinzip) bezeichnet.



Zentraleinheit:

Unter einer *Zentraleinheit* auch CPU (*Central Processing Unit*) genannt, versteht man in der Grundform die Zusammenfassung aus *einem Prozessor* (der sich wiederum aus *einem Rechenwerk* und *einem Steuerwerk* zusammensetzt) **und einem Hauptspeicher**. Gelegentlich wird der Begriff CPU auch für den **Prozessor** allein verwendet [Anmerkung: Herr Ruhland meint mit CPU immer NUR den Prozessor!!].



Externe Speicher:

Ein externer Speicher ist speziell dazu geeignet, größere Datenmengen langfristig aufzubewahren bzw. transportabel zu machen. Die wichtigsten externen Speichermedien sind: Magnetplatte / Diskette / Magnetband bzw. Streamer / optischer Speicher (CD-ROM)“ [Zitat Ende]

1.1.1 Komponenten der Zentraleinheit: Der Hauptspeicher

Der **Hauptspeicher** ist so etwas wie der „aktive Teil des Gehirns“ des Rechners, wobei zwischen **Arbeitsspeicher** und **Festwertspeicher** unterschieden wird:

1.1.1.1 Arbeitsspeicher

In dem **Arbeitsspeicher (RAM = Random Access Memory)** werden all die Daten bzw. Programme(-teile) geladen, die momentan benötigt werden. Der Arbeitsspeicher ist quasi das „Kurzzeitgedächtnis“ des Rechners. (Auf „Zaubertricks“ wie virtueller Arbeitsspeicher braucht hier nicht eingegangen zu werden) Der mit RAM bezeichnete Arbeitsspeicher ist aus „**flüchtigen**“

Speicherchips aufgebaut, das heißt der Inhalt geht bei einer Unterbrechung der Stromversorgung (z.B. abschalten des Rechners) verloren.

Beispiel: Während ich diesen Text tippe habe ich in meinem „Arbeitsspeicher“ in meinem Kopf den letzten Satz sowie vielleicht die nächsten drei Sätze, sowie mein Wissen „Wie tippe ich richtig“ und einige einfache Grammatik und Rechtschreibregeln usw. „geladen“; also all das, was ich **jetzt im Moment** wirklich brauche. Entsprechend hat auch mein PC die Teile des Word-Programms (oder evtl. sogar alle Teile) und die von mir eingetippten aber nicht gespeicherten Daten in seinem Arbeitsspeicher; also all das, was er **jetzt gerade braucht**, um die von mir eingegebenen Daten verarbeiten und seine **momentane Aufgabe erfüllen** zu können.

[Zitat] „Zugriffszeit

Zeitdauer, die für den Zugriff auf Daten benötigt wird. Damit ist die Zeitspanne gemeint, die der Schreib-/Lesekopf einer Festplatte benötigt, um sich an die Position zu bewegen, an der sich die gewünschten Daten befinden; die die Netzwerk- oder Internet-Protokolle benötigen, den Zugriff zum Server aufzubauen; mit der ein RAM-Baustein reagiert:

Für einen PC mit maximal 60 MHz Systemtakt (Pentium 90, 120 und 150) reicht eine Zugriffszeit von 70 ns. In PCs mit 66 MHz oder noch mehr externem Takt (siehe auch Prozessor) sollten 60-ns-Module oder die schnellen SDRAMs eingesetzt werden. Findet man im Motherboard-Handbuch keine Angabe zur Zugriffszeit, hilft ein Blick auf die Chips: meist geben die letzten ein oder zwei Ziffern Aufschluß über die Zugriffszeit. So steht 60. 06 oder 6 für 60-ns-Chips. Bei SDRAMs ist statt der Zugriffszeit die Zykluszeit aufgedruckt, also der Zeitraum zwischen zwei Lesezugriffen (10, 12, oder 15 ns beziehungsweise 0, 2 oder 5). Neuen SDRAMs müssen die gleiche oder eine bessere Zykluszeit haben als die installierten. ACHTUNG: Keinesfalls dürfen neuen Module eine höhere Zugriffszeit haben als die installierten. Sie wurden mit der Systemgeschwindigkeit nicht Schritt halten und womöglich Systemabstürze verursachen. Schnellere Chips können eingesetzt werden - auch wenn sie selten einen Tempogewinn bringen. Denn im BIOS eingestellte Wartezyklen gelten für alle Bänke. Nur einige Hauptplatinen wie die AB-ITSV von Abit sprechen die Bänke getrennt an. Auf der sicheren Seite ist man, wenn pro Bank nur RAM-Bausteine mit gleicher Zugriffszeit verwendet werden.“ [Zitat Ende] [Quelle: <http://www.glossar.de/glossar/>]

1.1.1.2 Festwertspeicher

Das „Langzeitgedächtnis“, also Der **Festwertspeicher** kann i.d.R. nur gelesen, nicht aber verändert / beschrieben werden, und dient der Aufbewahrung wichtiger grundlegender Teile des Betriebssystems, auf die beim Einschalten des Rechners automatisch zugegriffen wird (z.B. hardwarenahe Programme zur Ansteuerung des Bildschirms oder zur Kommunikation mit der Tastatur). Wie gesagt, i.d.R. ist ein Festwertspeichern (vom Hersteller) einmal beschrieben worden, und kann nur gelesen werden; die englische Abkürzung für solche „nur-von-lesen-können“-Speichermedien ist **ROM (Read Only Memory)** [Vgl. auch: CD-ROM: Ein CD-Rohling kann einmal beschrieben werden, danach nur noch gelesen, löschen oder überschreiben der Daten ist nicht möglich. (Û beliebig oft beschreibbare Diskette)] Inzwischen sind jedoch auch Festwertspeicher gängig deren Inhalt geändert werden kann (PROM, EPROM); trotzdem gilt auch hier: Der Festwertspeicher, „das Langzeitgedächtnis“ hat wichtige Grundfunktionen gespeichert, soll diese log.weise auch lange behalten und wird daher nicht ständig neu überschrieben, im Gegensatz zum Arbeitsspeicher (RAM).

Um beim Beispiel von oben zu bleiben: In meinem Langzeitgedächtnis (meinem „Festwertspeicher“) sind meine grundlegendes Wissen von Sprache und (noch „more Basic“) Grundregeln meiner Grob- und Feinmotorik etc. „abgespeichert“. Entsprechend sind im Festwertspeicher meines PCs ganz wichtige grundlegende Daten des Betriebssystems aufbewahrt.

1.1.1.3 Exkurs: Bits und Bytes: Was ist das?

Wenn die ganze Zeit von „Speicher“ die Rede ist, stellt sich die Frage **wie** werden Daten denn überhaupt **gespeichert**? Bzw. in welchem **Code** werden Daten abgelegt? Die „Daten“ dieses Textes sind im „Buchstabencode“ = Schrift abgelegt, aber ein Rechner kann „nur“ rechnen aber nicht lesen (i.S.v. Schrift lesen). Und wie werden Daten **gemessen**, in welcher **Einheit**?

Die ganze Computertechnologie beruht auf dem Prinzip des **Binär-codes**, d.h. alle Daten werden durch Kombination der Werte 0 und 1 „geschrieben“. Der Grund dafür ist, daß sich relativ einfach logische elektronische Schaltelemente bauen lassen die mit „Strom fließt“ und „Strom fließt nicht“ arbeiten. (Die ganze Logik mit der der Rechner „denkt“ basiert auf der Boole'schen Algebra, also der Aussagenlogik (vgl. Mathe I). Mit den Grundfunktionen $x = y$ (Identität) und $x = \bar{y}$ (Negation), sowie den Verknüpfungen AND (Konjunktion) und OR (Disjunktion) lassen sich alle logischen Funktionen darstellen und damit „Logik technisch Bauen“! {mit 1 = wahr = „Strom“ & 0 = falsch = „kein Strom“}) Das heißt aus der Sicht des Computers sind Zahlen und Zeichen nur indirekt verwendbar, weil die Sprache des Computers nur zwei Kodierungszeichen kennt, und zwar 0 und 1 (**Bit** = Binary Digit). Anders ausgedrückt: Jedes Bit kann den Wert 0 oder 1 annehmen. Daten werden im Rechnern zeichenweise durch die Kombination mehrerer **Binärziffern** (0 & 1) dargestellt. Der Buchstabe A hat beispielsweise folgende Binärfolge: 1 1 0 0 0 0 1, der Buchstabe V: 1 1 1 0 0 1 0 1, usw. Hat man acht „Stellen“, die mit 0 oder 1 belegt werden können, also 8 Bit, kann man insgesamt $2^8 = 256$ verschieden Zeichen „definieren“ bzw. im Binär-code schreiben (2^8 ist die Summe aller möglichen Kombinationen von achte „Stellen“ die jew. Mit 0 oder 1 belegt sind). Da dies i.d.R. ausreicht um alle benötigten Zeichen darzustellen, ist definiert worden: **8 Bits = 1 Byte**.

Ein **Hauptspeicher** besteht aus mehreren Speicherzellen, wobei **jede Speicherzelle gleich ein Byte groß** ist. Nun wird auch die „Formel“ aus dem „Ruhland-Skript“ logisch (zupal wenn man sie noch ein wenig abwandelt):

→ **1 Speicherzelle = 1 Adresse = 1 Byte = 8 Bit = 2 8 Möglichkeiten = 256 Werte (0, 1, 2, ... ,255)**

→ **Pro Zeiteinheit 1 Wert zuweisbar / abfragbar** (heißt, Ich kann halt nicht gleichzeitig auf mehrere Werte zugreifen, nur nacheinander)

→ **Zugriffszeiten müssen daher recht hoch sein, damit der Rechner vernünftig rechnen kann (z.B. 7 Nanosekunden)**

Umrechnung der Bit und Byte Angaben:

8 Bit = 1 Byte. Ein Kilo-Byte: 1 KB = 2 10 Bytes ~ 1000 Bytes. Ein Mega-Byte: 1 MB = 2 20 Bytes ~ 1 Mio. Bytes.

„Typische“ Größe eines Hauptspeichers eines PCs: 128 MB ≙ 128 Mio. Byte ≙ 128 Mio Speicherzellen mit je 8 Bit

[Anmerkung: Das komische Zeug was jetzt im Skript folgt; also der ganze Bereich: „Speicherung für Nutzer bedeutsamer Information SETZT ZUSÄTZLICHE(...)“ bis „(...) Verdichtung durch Delta-Speicherung (z.B. MPEG)“; habe ich hier nicht mit übernommen. Meiner Meinung nach passt das hier nicht richtig rein (es hat gliederungsmäßig eher einen „ach da fällt mir ein“ - Charakter), ich habe es daher unter den Punkt „1.3 Vernetzung, Datenmengen, etc.“ gestellt]

1.1.2 Komponenten der Zentraleinheit: Der (Zentral)Prozessor (CPU)

Der Prozessor besteht aus einem Rechenwerk und einem Steuerwerk. Das **Steuerwerk** ist dafür zuständig den Ablauf des Befehls und Datenflusses zu regeln – ist quasi der „logistische“ Teil des CPU, die Koordinationseinheit – während das **Rechenwerk** die arithmetischen und logischen Operationen durchführt. Erst wenn die Verarbeitungsgeschwindigkeit des Steuerwerkes, der Recheneinheit aufeinander abgestimmt sind, wenn die Zugriffszeit auf den Hauptspeicher dazu passt und die Verbindungen zwischen den ganzen Elementen (Bus) auch genügend Datenmengen durchlassen, ist so eine Zentraleinheit optimal eingestellt. Ein „schneller“ Prozessor (also mit hoher Taktgeschwindigkeit) allein bringt's noch nicht.

[Zitat] „Eine geeignete Kennziffer zur Bewertung der Leistungsfähigkeit von Prozessoren ist die Maßzahl **MIPS** (Million Instructions Per Second). Sie beschreibt, wie viele Befehle einer gegebenen Anweisungsfolge pro Sekunde vom Prozessor verarbeitet werden können. Hersteller drücken die Leistungsfähigkeit ihrer Prozessoren i.d.R. durch die Maßzahl Megahertz (**MHz**) aus, dadurch wird die **Taktfrequenz** des Prozessors angegeben. Die Angabe MHz erlaubt **jedoch keinen direkten Rückschluß auf die Verarbeitungsgeschwindigkeit** eines Prozessors. Vielmehr hängt die Verarbeitungsgeschwindigkeit auch von anderen Faktoren, etwa der **Architektur** des Prozessors, der Integrationsdichte der **Schaltkreise** und dem Mix der zum Einsatz kommenden **Programme**, ab.

Die **interne** Verarbeitungsgeschwindigkeit eines Prozessors hängt davon ab, wie schnell z.B. die einzelnen Einheiten Rechenwerk, Steuerwerk und Hauptspeicher operieren. Darüber hinaus kommt es darauf an, wie rasch zwischen Rechenwerk und Steuerwerk, den beiden Komponenten des Prozessors, sowie zwischen Prozessor und Hauptspeicher kommuniziert werden kann. Diese Einheiten sind mit sogenannten *Bussen* verbunden, die man sich als Kabel mit mehreren Adern vorstellen kann.“ [Zitat Ende]

1.1.3 Peripheriegeräte (Speichermedien, Ein- und Ausgabegeräte)

Neben den **Eingabegeräten** (Tastatur, Maus, Scanner, etc...) und den **Ausgabegeräten** (Monitor, Drucker etc...) zählen hierzu insbesondere die **Speichermedien**. Die bekanntesten Speichermedien dürften, neben der **Festplatte** oder Hard-Disk, die **Disketten**, **Magnetbänder**, die **CD-ROMs** und **DVDs** sein.

Man kann nun die Speichermedien nach unterschiedlichsten Kriterien einteilen. Herr Prof. Dr. Ruhland hat hier zwischen **magnetischen** und **optischen** Speichermedien unterschieden. Der prinzipielle Unterschied uns jedem aus dem **Alltag** bekannt: Vergleiche **Audiokassetten** und **Musik-CDs**! Jeder weiß, dass man Audiokassetten "überspielen" kann, und das geht solange, bis die Kassette zu sehr "gebraucht" ist, also abgenutzt ("rauschen"). Audiokassetten sind nichts anderes als **Magnetbänder** => Ich kann die Dinger zwar immer wieder neu bespielen, die Qualität nimmt jedoch mit der Zeit ab. Demgegenüber haben normale Musik-CDs den Vorteil, dass die Qualität der Aufnahme auch durch 10.000faches abspielen nicht leidet. (Weil Laser lediglich "abtastet" aber nicht "berührt, wie z.B. Nadel eines Plattenspielers.) Der Grund ist hier, dass es sich um einen **optischen** "Musikspeicher" handelt. Auch kann man CDs nicht dadurch "kaputt" kriegen, dass man z.B. Magneten dran hält. Wer demgegenüber jedoch sein Audiokassetten-Sammlung immer auf den Lautsprechern (= > Magnetfeld!) liegen hat, braucht sich nicht wundern, wenn die "tollen alten Aufnahmen" auf einmal so scheiße klingen...

1.1.3.1 Magnetische Speicher

Wie gesagt, das Prinzip der magnetischen Speicher ist zunächst dasselbe, wie bei einer Audiokassette. Tatsächlich gibt es auch im Bereich der Datenspeicherung **Magnetbänder**:

Ein **Magnetband** besteht aus einer Kunststoffolie mit einer aufgedampften magnetisierbaren Schicht, auf der Daten gespeichert werden. Für Mikrocomputer existieren Magnetbänder in der Form von Kassetten, die als Streamer bezeichnet werden. Diese Magnetbänder haben eine sehr hohe **Speicherkapazität** von bis zu mehreren **Gigabyte**. Magnetbänder bzw. Kassetten besitzen jedoch den Nachteil, daß lediglich ein sequentieller Zugriff auf die gespeicherten Daten möglich ist. (Vgl.: Audiokassette: "Moment, Moment, ich muss noch vorspulen bis das tolle Lied kommt...") Um einen bestimmten Datensatz zu lesen müssen zunächst alle gespeicherten Daten nacheinander gelesen werden, wodurch lange Zugriffszeiten entstehen. Daher wird dieses Speichermedium hauptsächlich zur Datensicherung verwendet, z.B. für ein Backup (Sicherung) um einer ungewollten Datenvernichtung vorzubeugen. *Merksatz: Magnetbänder sind wie Audiokassetten, nur mit Daten und nicht mit Musik.*

Eine **Festplatte** ist ein Massenspeicher, welcher der Aufnahme großer Datenmengen dient. Es handelt sich um eine Kunststoff- oder Aluminiumscheibe, die mit einer **magnetisierbaren Schicht** überzogen ist. Daten werden in Form von Bitketten in "konzentrischen Spuren" durch Magnetisierung dargestellt. Eine Spur besteht aus mehreren Sektoren. In einem Magnetplattenspeicher sind i.d.R. mehrere übereinander liegende Platten angeordnet - man spricht auch von einem Plattenstapel. Die Platten drehen sich mit konstanter Geschwindigkeit. Auf die Daten greifen Schreib-Lese-Köpfe zu, die auf einem Zugriffs-kamm befestigt sind. Diese Köpfe können auf die gewünschte Spur positioniert werden und "warten" darauf dass der Sektor mit den zu verarbeitenden Daten "vorbeikommt". Diese Zugriffsform wird wegen der Wartezeit je Umdrehung auch als halbdirekter Zugriff bezeichnet. *Merksatz: Festplatte ist sowas wie eine "magnetische CD". Das heißt sie ist rund (= Zugriff ähnlich wie bei CD möglich, also kein langes Sulen nötig) aber immer wieder neu beschreibbar (weil magnetisch)*

Die **Diskette** stellt ein z.Z. noch weit verbreitetes Speichermedium für Mikrocomputer dar. Es handelt sich dabei um eine flexible **magnetisierbare** Scheibe in einer Plastikummhüllung. Sie funktioniert ähnlich wie die zuvor beschriebene Magnetplat-

te (halbdirekter Zugriff), verfügt jedoch über eine geringere Speicherkapazität und hat eine längere Zugriffszeit. Disketten gibt es im Standardformat von 3,5 Zoll. Sie bieten unter dem Standard Betriebssystem für PCs (DOS = Disk Operating System) eine Speicherkapazität von **1,44 MB**. (HD = high density).

Die technische Entwicklungen hat in der letzten Zeit eine Reihe **weiterer** transportabler Speichermedien erfunden, die in ihrer Gestalt und Funktionsweise den **Disketten** ähneln. Diese neuen Medien unterscheiden sich jedoch insbesondere durch höhere Umdrehungsgeschwindigkeiten, d.h. geringere Zugriffszeiten, und größere Speicherkapazitäten von **100 MB bis 1 GB** (z.B. Zip-, EZ- oder auch Jaz-Laufwerke).

1.1.3.2 Optische Speicher

Bei **optischen Speichern** werden die Daten mit einem Laserstrahl in die unterhalb der transparenten Schutzschicht liegenden Speicherschicht **aufgezeichnet**. Hierbei wird die Oberflächenstruktur verändert. Diese Strukturen können wiederum mit einem Laserstrahl **abgetastet** werden. Laserlicht hat eine sehr kurze Wellenlänge und kann somit genau positioniert werden. Die Möglichkeit einen Laserstrahl genau zu positionieren führt bei optische Medien zu einer **hohen Speicherkapazität**. Es werden verschiedene Techniken unterscheiden. (u.a.):

1. **CD-ROM** werden vom Hersteller beschrieben und vom Anwender gelesen. Die Speicherkapazität eines solchen Mediums liegt bei 640 - 750 MB. Die Lesegeräte als auch die Medien sind mittlerweile sehr preisgünstig, so daß Standardsoftware zunehmend über diese Medien vertrieben wird. Ein weiteres Anwendungsgebiet für diese Speichermedien liegt in der Verwaltung großer und weitgehend unveränderlicher Datenbestände, z.B. von Patenten und Büchern. Dabei gewinnt die Integration von Text, Grafik, Video und Audio zunehmend an Gewicht. Die Entwicklung in der CD-Technologie führte bislang - für den Anwender "zu Hause" über **einmalig** zu beschreibende, sog. **CD-R** (CD recordable), bis zu **mehrfach** zu beschreibenden **CD-RW** (CD rewriteable) CD's. (= > CD-Brenner nötig)
2. **DVD** als Abkürzung für Digital Versatile (vielseitig) Disk auch häufig als Digital Video Disk bezeichnet ist die jüngste Entwicklung im Bereich optischer Speichermedien. Durch **beidseitiges** Beschreiben des Datenträgers und die Verwendung von Laserlicht werden Speicherkapazitäten von **4,7 GB bis zu 17 GB** erreicht.

1.1.3.3 Weitere Speichertypen

Neben den oben beschriebenen Speichern gibt es noch eine Menge anderer. Zu erwähnen würden hier vielleicht als Beispiel einer "**Mischform**" die sog. MO. MO ist die Abkürzung für Magneto-Optical Disk. Diese lassen sich beliebig oft beschreiben und lesen. Auch **Spezialmedien** spielen eine Rolle: Genauso wie unsere Zentraleinheit können auch spezielle **Chips** als Speichermedium dienen. Diese kommen insbesondere im Bereich der "Plastikkarten" (vgl. "Geldkarte" der Sparkasse) aber auch in div. "normalen" technischen Geräten vom Auto über die Waschmaschine bis zum Videorecorder zum Einsatz.

"Rechnung" aus dem Ruhland-Skript:

Vergleich der Speichermedien nach

- Kapazität: Die Speicherkapazität einer typ. Festplatte *entspricht* 4 DVDs *entspricht* 40 CDs *entspricht* 35.000 Disketten
- Zugriffszeit: Der Zugriff auf Daten einer Festplatte dauert 100.000 bis 1 Mio. mal länger als auf Daten im Hauptspeicher

Hinweis: Den Punkt "**1.1. A betriebswirtschaftliche Konsequenzen**" habe ich hier weggelassen. Das geht es eigentlich nur um die "Branchenstruktur" auf dem Prozessor-Markt (z.B. Intel vs. AMD), und besagt im Kern: Weil die Technologie so schnell voranschreitet und jeder derjenige sein will, der den noch schnelleren Prozessor als erster auf den Markt wirft & das eine Hightech-Industrie ist, brauche ich viel Geld (ständig neue Maschinen Technologien, Fabriken). => Oligopol Markt mit "Preiskriegen" und "competition in time". Und Geld wird eigentlich nicht in den Phasen verdient, wenn ich die ganz neuen Produkte Vermarkte & Verkaufe, sondern dann wenn's "ruhiger" ist bzw. wenn ich "ältere" Prozessoren verkaufe.

1.2 Software

Unter dem Begriff **Software** versteht man alle Arten von Programmen und auch Daten, die ein Rechner zum sinnvollen Einsatz **braucht**. Software ist damit das **Gegenstück** zur Hardware, die die materiellen Bestandteile eines Rechnersystems umfaßt. Ohne Software ist ein Rechner sozusagen "tot". Software gibt es in den unterschiedlichsten Formen. Ob Betriebssystem, Gerätetreiber oder Anwendungsprogramm, alles fällt unter den Begriff Software. Die Speicherung von Software erfolgt auf Datenträgern wie Disketten, CD-ROMs oder Wechselpplatten. **Häufig benutzte** Software wird auf der **Festplatte** des Rechners gespeichert, um einen schnellen Zugriff darauf zu haben (Bsp.: Das Betriebssystem). **Hardware** determiniert sozusagen **physikalisch** die Möglichkeiten meines Rechners, die **Software** ist für die "**Intelligenz**" oder die "**logischen Fähigkeiten**" zuständig. **Beide** Komponenten bestimmen also, was ich mit meinem PC alles tun kann. **Ohne** Software ist ein Rechner im Prinzip nichts anderes, als ein Haufen elektronischer Bauteile.

Bei der Software unterscheidet man insbesondere zwischen **Betriebssystem** und **Anwendungen** (also Programmen wie Word oder AcrobatReader). Diese Unterteilung macht Sinn, weil beide Softwarearten **unterschiedliche** Aufgaben haben: Während z.B. MS EXEL ein Programm zur Tabellenkalkulation ist, also zur Lösung **ganz spezieller** Anwender"probleme" entwickelt wurde, ist die Aufgabe eines Betriebssystems (**BS**) viel **grundlegender**.

1.2.1 Das Betriebssystem

Damit ich auf meinem Rechner die diversen **Anwenderprogramme** (z.B. Microsoft Word) laufen lassen kann, brauche ich - quasi als "Basis" ein **Betriebssystem**. Das Betriebssystem definiert sich als Gesamtheit aller Programme, die - unabhängig von einer bestimmten Anwendung - den **Betrieb** des Computersystems **ermöglichen**. Sie gewährleisten die unterschiedlichen Betriebsarten und **steuern** die Abarbeitung der Anwendungsprogramme in der Zentraleinheit und das Zusammenspiel mit den peripheren Geräten. Ein Rechner **ohne** BS ist also sozusagen eine "**tote**" Ansammlung von Hightechteilen. Ähnlich wie bei einem Team von Spezialisten braucht es eine Instanz, die koordiniert, wer wann was zu tun hat. Genau diese Aufgabe fällt einem BS auf einem Rechner zu: Es **koordiniert** das Zusammenspiel der Einzelteile des Rechners. Es ist jedoch gleichzeitig auch "**Vermittler**" und "**Aufpasser**" gegenüber dem Anwender und Anwendungen.

"Vermittler" meint, dass das BS gegenüber dem Benutzer und/oder Anwendungen als "**Schnittstelle**" zur Hardware auftritt, die es erlaubt Befehle an den Rechner zu geben, und zwar in einer "vereinfachten" oder "standardisierten" Art und Weise. Um z.B. eine Datei auf einer CD-ROM zu öffnen muss ich **nicht** wissen, mit welchen Befehl-Codes mein CD-ROM Laufwerk arbeitet. Es reicht, wenn ich auf das CD-ROM -Symbol klicke, warte bis sich das "Fenster" geöffnet hat, und dann auf die Datei klicke. Das BS kann **meine "Befehle"** (klicken auf ein Symbol) **verstehen**, und entsprechend dem CD-ROM Laufwerk Befehl-Codes schicken. Umgekehrt "versteht" das BS was **mein CD-ROM** Laufwerk ihm "sagt" und sagt das wiederum dem Benutzer in einer Art und Weise die **er** versteht. (z.B. durch "öffnen" eines Fensters, in dem ein ganz best. Dateisymbol & der Dateiname dargestellt sind.) Ähnlich funktioniert das gegenüber den Anwendungen: Ein Programmierer von MS Word muss nur wissen, **welche Befehle** z.B. das **BS** Windows 98 **verstehen**, er muss **nicht** wissen, welche Befehle die **Hardwarekomponenten** auf dem Rechner des Endnutzers jew. verstehen. (Wäre ja auch blödsinnig, man müsste dann ja für **jeden** PC eine individuelle Word-Version programmieren)

Mit "Aufpasser" meine ich, dass das BS auch darauf **achtet**, dass weder der Benutzer noch irgendwelche Anwendungen "**blödsinnige**" oder "**gefährliche**" **Befehle** geben. So nimmt das BS den Befehl "Lösche Datei xy von der CD-ROM" garnicht erst an, sondern sagt mir - wenn es gemerkt hat, dass das garnicht geht - sowas wie "Datei kann nicht gelöscht werden".

"(Zitat) **Betriebssysteme** übernehmen das "**interne Management**" eines Computers, die Koordination und Synchronisation der rechnerinternen Abläufe, wobei folgende **Ziele** angestrebt werden: Die **Effizienz** von Computersystemen gewährleisten: Also der schnelle Durchsatz von Anwendungsprogrammen im Stapelbetrieb, sie sollen für kurze Antwortzeiten im interaktiven Betrieb sorgen, und eine hohe bzw. gleichmäßige Auslastung der Betriebsmittel garantieren (Hardware "ausreizen"). Die Computernutzung durch den Menschen **vereinfachen**, **komfortabler** und **sicherer** gestalten: Ein **BS** bietet eine komfortable

Benutzeroberfläche, Datensicherheit und umfassende Information über Systemzustände, Fehler usw.

Typische Aufgaben des Betriebssystems:

- Bereitstellen einer **logischen** Ebene für den Benutzer
- Kommunikation **mit** dem Benutzer
- Laden, Starten, **Steuern** und Beenden von Programmen
- Öffnen, Bearbeiten, Schließen und **Verwalten** von Datenbeständen
- Verwaltung der Hardware (Prozessor, Hauptspeicher, periphere Geräte) und Zuordnung zu Programmen
- Gewährleistung von Betriebsarten (z.B. im Mehrprogrammbetrieb Koordination des parallelen Ablaufs mehrerer Programme)
- Protokollierung (z.B. von Zugriffen auf Datenbestände)
- Fehlererkennung im System
- Gewährleistung von Datensicherheit und Datenschutz

Um ihre Aufgaben erfüllen zu können, verfügen Betriebssysteme über die Komponenten Jobmanagement ("externe Steuerung"; stellt einerseits die Verbindung zwischen Computer und Außenwelt her und dient andererseits der Abwicklung der zur Bearbeitung anstehenden Benutzeraufträge.), Prozeßverwaltung oder Taskmanagement ("interne Steuerung"; befaßt sich mit der tatsächlichen Auftragsausführung) und der Datenverwaltung (Datenmanagement)... (Zitat Ende)"

Hier ist jetzt erstmal das **Grundlegende** zum Thema Betriebssystem gesagt. Weil aber leider im "Ruhland Skript" die Stichworte "Boot-Prozess" und "File (Datei)-system" auftauchen, und weil ich unbedingte etwas zur Frage 14 aus der Übung sagen möchte, folgen hier jetzt noch ein paar Unterpunkte:

1.2.1.1 Der Boot-Vorgang und BIOS

Wenn ich einen Rechner einschalte, kann ich nicht sofort damit arbeiten. Der Rechner muss sich erst "hochfahren", oder **booten**. Hier läuft zunächst das BIOS ab, und dann wird das Betriebssystem gestartet.

Und was ist das "BIOS"? BIOS ist die "(Zitat) Abkürzung für "Basic Input Output System". **BIOS** ist der **hardwaregebundene** Kern eines Betriebssystems, der beim Ausschalten nicht gelöscht wird und sich fest im EPROM (=> Festwertspeicher des Hauptspeichers) befindet. Dieser Chip wird **einmalig** mit Daten beschrieben, die vom Computer nur gelesen und nachträglich lediglich mit Hilfe von Spezialprogrammen (Flash-BIOS) verändert werden können. Nach jedem **Einschalten** des Rechners führt das BIOS zunächst einen **Selbsttest** durch. Dann **benutzt** der Computer das BIOS, um das **Betriebssystem** zu starten und den **Datenstrom** zwischen der Festplatte, Grafik-Karte, Keyboard, Maus und Drucker zu kontrollieren, **bis** ihm diese Aufgabe von einem anderen System - z.B. dem Betriebssystem - **abgenommen wird**. (Zitat Ende)" Das BIOS ist also ein **hardwaregebundenes Steuerelement**. Einen Rechner ohne BIOS gibt es nicht, dann hätte der Rechner auch keinen Hauptspeicher (und wäre eigentlich kein Rechner). Und im BIOS sind ganz **grundlegende Informationen** (welche Festplatte(n) vorhanden? Welchen Prozessor? Auf welchem Datenträger das BS suchen?) definiert.

Außerdem **braucht** man die "schmalspurintelligenz" der BIOS um das **BS aufrufen** zu können. Schließlich erwacht Software ja nicht von sich aus zum Leben. Es geht also auch um das "finden" des BS: "(Zitat) Der **letzte Schritt** beim Hochfahren (Booten) des PC ist das **Laden** des **Betriebssystems**. Das Start-Programm im BIOS versucht den *Master Boot Sektor* zu finden. Dieser "Boot-Sektor" ist der erste Sektor auf der Festplatte (meist Laufwerk C:) oder der Diskette, wenn von Diskette gebootet wird. Normalerweise prüft der PC beim Starten, ob eine Diskete eingelegt ist. Der Boot-Vorgang wird unterbrochen wenn eine Diskette ohne Betriebssystem (ohne Boot-Sektor) eingelegt ist. Ist keine Diskette eingelegt, sucht das Start-Programm auf Laufwerk C: nach einem Boot-Sektor. Ist der Boot-Sektor gefunden wird von dort ein kleines Programm geladen, das *Boot Strap*. Es übernimmt dann die Kontrolle und **lädt das Betriebssystem**. Das Start-Programm ist dann beendet. DOS, Windows oder ein anderes Betriebssystem übernehmen von hier aus (..Ende Zitat)" die Kontrolle.

1.2.1.2 Dateisysteme

"(Zitat) Wie wir schon gesehen haben ist der PC ein "Datenverarbeiter". Wir haben auch gesehen, dass die Daten aus Bits und

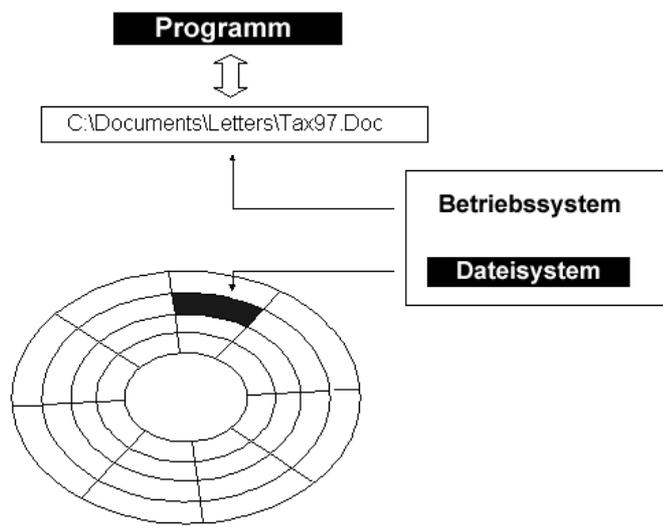
Bytes bestehen, die zu Dateien zusammengefasst sind. Eine der Hauptaufgaben des Betriebssystems ist es, diese Daten auf diverse Speicher zu schreiben. Bevor wir eine Datei auf eine Festplatte, eine Diskette etc. schreiben können, müssen diese formatiert werden. Auf diesen Seiten werden wir uns die Formatierung, Dateisysteme, etc. anschauen. Wir beginnen mit einem allgemeinen Überblick, um dann die FAT-Formatierung, die immer noch die Verbreitetste ist, genauer zu betrachten.

(...) Laufwerke sind Speichermedien, die ein Dateisystem enthalten können. Wenn ein Speichermedium formatiert wird, wird es organisiert und auf die Speicherung vorbereitet. Wenn wir ein Speichermedium formatieren, bekommt es sein *Dateisystem*. **Formatieren** kann mit der Einrichtung einer Bibliothek verglichen werden. Man muss Regale aufstellen und **ein System** zur Katalogisierung **einführen**, **bevor** man auch nur ein Buch ins Regal stellt. Wenn die Bibliothek erst eingerichtet ist; her mit den Büchern! Ähnlich ist es mit einem Speichermedium. Wenn wir formatieren, prägt man dem Datenträger ein Dateisystem auf, um ihn auf das **Aufnehmen** von Dateien **vorzubereiten**. Beim Formatieren haben wir die Wahl zwischen mehreren verschiedenen Dateisystemen:

- **FAT** (*File Allocation Table*), das alte, 16 Bit DOS-System wird wahrscheinlich in 90% aller PC benutzt. Mann nennt es auch FAT16, um es zu unterscheiden von:
- **FAT32** einer neuen FAT-Entwicklung, die Microsoft in Windows 95B - der Version vom Dezember 96 (OSR2) - eingeführt hat. Ihre Leistungsfähigkeit wurde in Windows 98 noch verbessert. (...) [Restlichen Dateisysteme weggelassen, es geht hier ja darum das Prinzip zu verstehen]

Das Dateisystem ist faktisch eine *Schnittstelle* BS und den Laufwerken. Wenn die Software, z.B. MS Word, eine Datei von der Festplatte anfordert, befiehlt das Betriebssystem (Windows 95/98) dem Dateisystem (FAT) die Datei zu öffnen. Das **Dateisystem** weiss, wo die Datei abgelegt ist. Es findet und liest die relevanten Sektoren und liefert sie dem **Betriebssystem**. (Zitat Ende)"

Wie man an diesem Beispiel recht gut sieht, ist die Logik wonach eine Datei physisch auf der Festplatte gespeichert wird, eine ganz andere, als die, die ich als Benutzer vom BS "präsentiert" kriege. Vorallem unter Betriebssystemen mit einer sog. **Graphischen Benutzeroberfläche** (Gegenstück zur "Zeichenorientierten", wo ich nur Zeilen auf dem Bildschirm habe und meine Befehle eintippen muss.) wird die genialität der "Erfindung" BS deutlich: Scheißegal wie meine Dateien "tatsächlich" auf der Festplatte verstreut rumliegen, ich lege mir - z.B. unter Windows 95 - einfach auf C:\ (= Festplatte) einen Ordner "Dokuments" an, in diesem Ordner erstelle ich noch einen Ordner "Letters" und da rein "lege" ich (= speicher ich) den Brief "Tax97" (also die Datei Tax97.doc). Das ist eine Logik, die ich mir merken kann, den die kenn ich: "In meiner Schreibtisch liegt eine Mappe, darin ein Briefumschlag und darin ist der Brief". (Ersetze in diesem Satz einfach "Schreibtisch" durch "C:\", "Mappe" durch "Ordner", "Briefumschlag" durch "Unterordner" und "Brief" durch "Tax97.doc" und man erhält eine typische Windows-Pfadangabe, vgl. auch Bild)



1.2.1.3 Das Betriebssystem an sich

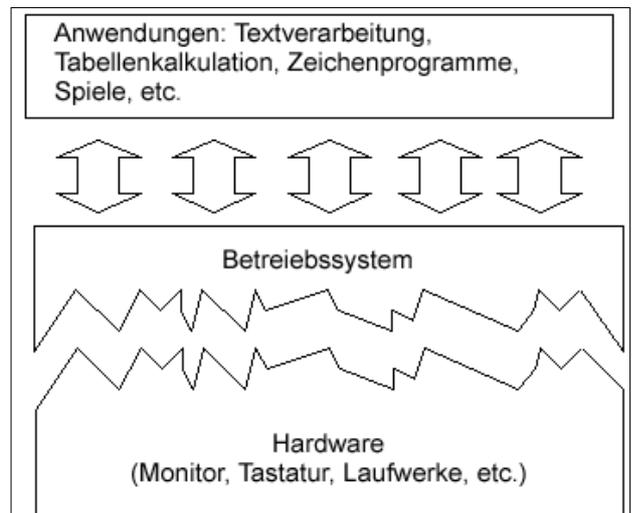
"(Zitat) Betriebssysteme sind die derzeit **wichtigsten** PC-Programme. Ohne Betriebssysteme läuft kein Computer: Sie **verarbeiten** vom Benutzer **eingeebene** Daten, **verwalten** die **gespeicherten** Dateien und **kontrollieren** angeschlossene Geräte wie Drucker und Festplatten. Gleichzeitig dienen sie als **Basis** für **Anwenderprogramme** wie Text- und Dateiverarbeitung, die ohne den **Unterbau** des Betriebssystems nicht laufen können. Mit der Entwicklung von MS-DOS und WINDOWS gelang Microsoft der Durchbruch auf dem Markt. Während DOS sich noch weitgehend auf die Eingabe von **Programmbefehlen** über die **Tastatur** beschränkte, kann der Nutzer bei Windows den Computer über eine **graphische Oberfläche** mit Hilfe der Maus steuern.

Beispiele für gängige Betriebssysteme: die Anfänge: **MS-DOS**, Novell, Novell-DOS (DR-DOS)
 die Massenware: **WINDOWS** (wichtigsten Versionen 3.11, 95, 98, Millennium)
 die Ableger: Windows NT (die wichtigsten Versionen NT4, 2000)
 ambitionierte Versuche: OS/2, BeOS
 die Alternativen: **Linux**, Mac OS von Apple, (Zitat Ende)"

"(Zitat) Das **Betriebssystem** kann man wie folgt betrachten:

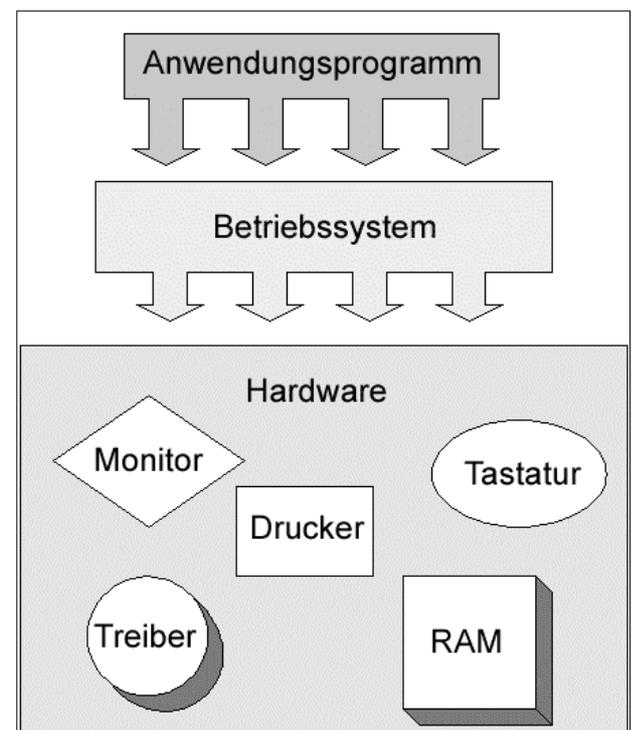
- Ein Betriebssystem ist eine Ansammlung von Dateien, die am Ende der PC- start-up-Routine von der Festplatte gelesen werden.
- Ein Betriebssystem ist eine Programmebene. Es ist mit der Hardware verbunden, um für eine optimale Ausführung der Benutzeranwendungen zu sorgen.

Die erste Definition sagt nicht besonders viel aus. Lassen Sie uns mit der **zweiten** beginnen: Das Betriebssystem verbindet Software und Hardware. Es befähigt Nutzeranwendungen wie "Works", "Office", etc., mit allen möglichen Hardware-Konfigurationen zu **kooperieren**. Die Beziehung zwischen Hardware und Benutzerprogrammen können Sie so verstehen: Es gibt **unzählige** Varianten des **PC**. Er kann verschiedene Festplattentypen haben, CPUs, Grafikkarte, etc. All diese PC-Konfigurationen verhalten sich auf **ihre eigene** Weise. Die **Benutzer-Software** ist **standardisiert**, und **erwartet** vom PC, immer auf **die gleiche Art** behandelt zu werden. Wie bringen wir die beiden Ebenen dazu, zusammenzuarbeiten? Können wir die Unterschiede in der PC-Konfiguration kompensieren, so dass ein Standardprodukt wie "Works" funktioniert? Ja, wir können. Wir lesen das Programm über ein *Betriebssystem* - ein System, das ausgleicht und die Hardware standardisiert: Sie sollten das Betriebssystem als eine notwendige Programmebene betrachten, die Schlaglöcher in ihrer PC-Hardware ausgleicht.



Das Betriebssystem erkennt die Hardware

Für das **Benutzerprogramm** stellt die **Hardware** des Rechners **Ressourcen** dar. Denken Sie nur an Ihr Textverarbeitungsprogramm: Sie wollen Ihren Text drucken. Das Programm gibt einen **Druckbefehl**, und erwartet, dass das Dokument wie von Ihnen gewünscht **gedruckt wird**. Das Textverarbeitungsprogramm verarbeitet die Daten entsprechend Ihrer Befehle. Wie diese in Signale übersetzt werden, die *Ihr* Drucker versteht - dass ist **nicht** das Problem des **Textverarbeitungsprogrammes**. Für das Programm ist der Drucker eine Ressource. Die Verbindung zu solchen Ressourcen läuft über das Betriebssystem. Dies gilt für alle Hardware-Komponenten:



Wie Sie sehen spielt das **Betriebssystem** im PC eine tragende Rolle. Es muss **jegliche** Hardware erkennen können. Es hat keinen Zweck eine neue Maus anzuschließen, wenn sie nicht funktioniert! Und was lässt sie funktionieren - das Betriebssystem. **Es muss Ihre Maus erkennen!** (Zitat Ende)" Naja, und damit das BS die Maus erkennen kann, bzw. mit der Maus kooperieren kann muss der entsprechende Treiber installiert werden. Aber das führt jetzt tatsächlich zu weit...

Zu Windows 3.x, 95 / 98:

"(Zitat) Das Betriebssystem **Windows** wurde von Microsoft erstmals **1985** vorgestellt und wurde 1990 mit der Version 3.0 marktfähig. Es brachte die bereits von Apple angewandte **grafische Benutzeroberfläche** mit Fenstertechnik und Maussteuerung sowie Multitasking in die Microsoft-Welt. Windows war **zunächst** als **Zusatz** zum alten Betriebssystem MS-DOS gedacht, konnte sich später aber zu einem "echten" Betriebssystem entwickeln. (Ganz wichtig dieser Satz!) (...) Mit Millionen-Aufwand führte Microsoft 1995 "Windows 95" ein, dessen Nachfolger "Windows 98" heute in den Regalen steht. Die "Windows"-Familie - komplettiert durch "Windows NT" respektive "Windows 2000" für Unternehmens- Netzwerke - läuft Schätzungen zufolge auf über 85 Prozent aller Personal Computer. (...)

Windows-Versionen im **historischen** Rückblick:

November 1983 Microsoft kündigt Windows auf der Computermesse Comdex als **graphische Benutzeroberfläche** und **Erweiterung** des Betriebssystems MS-DOS an. Juni 1985 Microsoft Windows 1.0 erscheint in englischer Sprache. Ähnlich wie zuvor beim Apple-Rechner Macintosh gibt es nun Mausbedienung, aufklappbare Befehlsflächen (Pull-Down-Menüs) und Symbole (Icons) für Dateien oder Programme. (Allerdings konnte niemand irgend etwas Sinnvolles mit dieser Version tun!) (...) März 1987 Windows 2.0 ist verfügbar und bringt einige Verbesserungen wie z.B. sich überlappende Fenster. Dies sind die ersten Versionen, mit denen sich leidlich arbeiten läßt, zumal auch die Software-Hersteller langsam auf den Windows-Zug aufspringen. Das erste Anwendungsprogramm für Windows ist die Tabellenkalkulation Excel. Juli 1988 (...) Die erste brauchbare Version, Windows 3.0, ist verfügbar. Sie nutzt die Fähigkeiten neuer INTEL-Prozessoren aus und kann mehrere Programme gleichzeitig laufen lassen (Multitasking). Damit schafft Windows den Durchbruch. Auf breiter Front wird Software für das neue Betriebssystem entwickelt. (...) August 1995 Windows 95 wird mit der - zumindest für die Software-Industrie - größten Werbekampagne aller Zeiten eingeführt und möchte das Ende der 16-Bit-Betriebssysteme einläuten. (...) 25. Juni 1998 Windows 98 wird als Evolutions-Stufe von Windows 95 ausgeliefert. Wesentliches Merkmal ist der integrierte INTERNET-Explorer. (usw.) (...)

Windows 95

Windows 95 war der Nachfolger der Windows 3.11-Version und wurde von Microsoft als ein **eigenständiges Betriebssystem** entwickelt. Die **Trennung zwischen Windows und DOS wurde aufgehoben**. Die Benutzeroberfläche hat sich an das objektorientierte Konzept der Datei- und Ressourcenverwaltung moderner 32-Bit-Betriebssysteme angepaßt. Nachfolger von Windows 95 sind Windows 98 und Windows ME ("ME" steht für Millenium Edition)

MS DOS

Abkürzung für "**Microsoft Disk Operating System**" • Vom US-Hersteller Microsoft entwickeltes Betriebssystem für PCs, das weltweit am weitesten verbreitet war (siehe auch DOS) und von Windows abgelöst wurde.

Anfang und Ende von MS-DOS:

August 1981: MS-DOS 1.0 basiert auf 4 000 Programmzeilen Assembler-Code, kann einen Arbeitsspeicher bis 64 Kilobyte (KB) verwalten und Daten auf Disketten mit 160 KB speichern. In drei Dateien stellt es die grundlegenden Dienste für den IBM PC zu Verfügung, dazu einige Hilfsprogramme wie Edlin oder Format. (...) März 1993: MS-DOS 6.0 brachte u.a. komprimierte Laufwerke, **verbessert** die **Zusammenarbeit** mit **Windows** und unterstützt CD-ROM-Laufwerke. Ansonsten wurde die Anwender-Gemeinde mit hinzugekauften und nur grob aufeinander abgestimmten Tools bei Laune gehalten. Die letzte ECHTE von Microsoft herausgebrachte Version war 6.22. - Erstmals werden mehr Anwendungsprogramme für Windows als für DOS verkauft. September 1995: **MS-DOS** ist nur noch ein "**Unterbau**" von **WINDOWS 95** und **kein selbständiges** Betriebssystem mehr - wird in dieser Form manchmal auch "MS-DOS 7" genannt. März 1998: Microsoft verabschiedet sich endgültig von seinem ersten Computer- Betriebssystem. Das Ende für das altgediente System verkündete Bill Gates Ende März auf der Entwickler-Konferenz WinHEC in Orlando (Florida). "**Windows 98**" wird danach die letzte Windows-Version sein, die auf "MS DOS" **aufbaut**. "Der Nachfolger von Windows 98 wird eine Art Windows NT für Verbraucher sein", sagte Gates. (1999 steht fest, daß er damit Windows2000 meinte) (Zitat Ende)"

Warum habe ich jetzt hier diese "**Geschichtsstunde**" veranstaltet? Das hat etwas mit der Beantwortung der Frage 14 aus der Übung zu tun:

1.2.1.4 Hinweis zur Frage 14 aus der Winfo-Übung

Frage 14:

Ihr Rechner bringt nach dem Einschalten die Ausschrift: *Kann das Betriebssystem nicht finden!*

Hinweis: So wie die Frage gestellt ist ("kann das **Betriebssystem** nicht finden"), sind folgende Antworten richtig:

- Was bedeutet das? Erklären Sie, was das Betriebssystem ist und was es leistet.

Das bedeutet, dass das Start-Programm im BIOS den Master Boot Sektor nicht finden kann (weil beschädigt oder gelöscht...), oder dass das "Boot Strap" die BS-Dateien nicht finden kann (also die Dateien die es braucht, um das BS zu starten), weil diese gelöscht sind, oder "kaputt" oder... Auf jeden Fall kann das Betriebssystem nicht gestartet werden. (Teil zwei der Frage "Was ist ein BS?" schenk ich mir jetzt hier)

- Können Sie dann trotzdem mit der Anwendungssoftware WinWord arbeiten?

Nein, denn WinWord benötigt, wie alle Anwenderprogramme, ein BS um laufen zu können. Weil... (siehe Skript)

- Können Sie dann den Text, den Sie eigentlich bearbeiten wollten, auf ihre Diskette kopieren und an einem anderen Computer weiterarbeiten?.

Nein! (Das ist jetzt wichtig: Die richtige Antwort auf die Frage - so wie sie formuliert wurde - lautet "Nein"). Denn: ohne ein BS kann ich garnichts machen. Mögliche Lösung des Problems (jetzt mal davon ausgehend, dass auch ein "Computerfreak" das installierte BS nicht zum laufen kriegt): Ich besitze eine Diskette, von der ich booten kann, und auf der sich eine "abgespeckte" Version von MS DOS befindet (nur Grundfunktionen). Also eine der "berühmten" Bootdisketten die jeder Nutzer haben sollte Dann kann ich diese Start-Diskette ins Laufwerk legen und den Rechner wieder anschalten. Ich drücke eine bestimmte Taste, sobald auf dem Bildschirm weiße Schrift auf schwarzem Grund erscheint (lange bevor Windows startet) und halten Sie sie gedrückt, bis das Boot-Menü erscheint. Hier wähle ich "Nur Eingabeaufforderung", und mein Rechner bootet von der Diskette (statt von der Festplatte) und lädt mein MS DOS. Jetzt habe ich ja ein BS geladen (MS-DOS), und kann mich mit dem DOS-Befehl **copy C:\Verzeichnisname\Verzeichnisname\...\Dateiname.Endung A:** die gewünschte Datei auf meine Diskette kopieren. Wer sich jetzt fragt: Hä? Ich denke ich habe die Boot-Diskette im Laufwerk? Dem sei gesagt: Da dieses "bisschen MS-DOS" so "klein" ist, wird das ganz in den Arbeitsspeicher geladen. Das heißt, ich kann dann ruhig die Boot-Diskette gegen eine leere Diskette austauschen und meine Datei darauf speichern.

Was anderes wäre es, wenn die Meldung gekommen wäre "*Kann Windows nicht starten*". Da es diese komische Verschmelzung von MS DOS und Windows (insb. Der grafischen Benutzeroberfläche) gibt, bedeutet so eine Meldung in der Regel "Der Rechner kann nur nicht das schöne bunte Windows starten. Die MS-DOS-Ebene (des Betriebssystems) funktioniert aber wahrscheinlich noch." Dann brauche ich keine Boot-Diskette sonder kann den Rechner "einfach so" im DOS-Modus starten.

Das mag jetzt vielleicht ein wenig verwirrend sein,. Aber wenn man sich das wie folgt vorstellt, wird's einigermaßen logisch: Das Betriebssystem als Ganzes nennt sich Windows 95/98. Streng genommen ist es so, das um das "alte" **Betriebssystem MS DOS** (quasi als **Kern**) herum das tolle **neue BS Windows 95/98** "gebaut" wurde. Wenn nun also die Meldung kommt "Kann kein Betriebssystem finden" heißt das, es ist **garnichts** da, und wo nix ist geht nix. Wenn aber die Meldung kommt "Kann Windows nicht starten" heißt das logischerweise, dass (hoffentlich) der Kern MS DOS noch ansprechbar ist, das also ein Betriebssystem gefunden wird (aber - wenn man jetzt von Win95/98 redet - sozusagen "nicht das Ganze", und zum Glück funktioniert der "Teil" MS DOS auch ohne den Rest).

1.2.1.5. Ein paar Erläuterungen zu den Stichworten im "Ruhland Skript" Folie 27/45:

Einprogrammbetrieb (single tasking):

Einzelne Benutzeraufträge werden von der Zentraleinheit nacheinander bearbeitet. Es befindet sich jeweils nur ein Programm im Arbeitsspeicher, das bei der Ausführung sämtliche Betriebsmittel (Prozessor, Speicher, Ein- und Ausgabeperipherie) zuge- teilt erhält.

Mehrprogrammbetrieb (multitasking):

Mehrere Benutzeraufträge werden von der Zentraleinheit gemeinsam (abwechselnd in Zeitabschnitten verzahnt) bearbeitet. Die abzuarbeitenden Programme erhalten bei der Ausführung die benötigten Betriebsmittel wechselseitig zugeteilt.

Stapelverarbeitung (batch processing):

Nutzungsform, bei der ein Auftrag vollständig beschrieben und als Ganzes erteilt sein muß, bevor mit seiner Abwicklung begonnen werden kann.

Einbenutzerbetrieb (single-user mode):

Eine Zentraleinheit unterstützt gleichzeitig nur einen Arbeitsplatz bzw. einen menschlichen Benutzer.

Mehrbenutzerbetrieb (multiuser mode):

Eine Zentraleinheit unterstützt gleichzeitig mehrere Arbeitsplätze bzw. mehrere menschliche Benutzer.

1.2.3 Ein paar Stichworte / Übersichten zum "Querlesen"

1.2.3.1 Betriebssysteme - Marktübersicht und Trends

A. PC-Betriebssysteme

- * starkes Marktwachstum
- * Dominanz von Microsoft

a) MS-DOS + Windows 3.x (nur noch historisch interessant)

- 16-Bit-Betriebssystem
- 1 MB Hauptspeicher
- hierarchische Datenverwaltung
- Dialog- und Batchverarbeitung
- Singletasking und "unechtes" Multitasking (Task-Switching)
- "Verbunddokumententechnik": OLE (Object Linking and Embedding) erlaubt es unterschiedlichen Anwendungen über DDE (Dynamic Data Exchange) zu kommunizieren und Daten aus einer Anwendung in eine andere zu holen
- Single-Using

b) Windows 95 / 98

- 32-Bit-Betriebssystem
- 16 MB Hauptspeicher
- neue grafische Benutzeroberfläche
- standardmäßig Netzwerkfunktionen
- Plug and Play (eingeschränkt)
- lauffähig: DOS-Programme, Windows 3.x-Programme, 32-Bit-Windows-Programme
- echtes Multitasking (nicht für Windows 3.x-Programme)

c) OS/2 Warp

- 32-Bit-Betriebssystem (IBM)
- 16 MB Hauptspeicher
- grafische Benutzeroberfläche (Presentation Manager)
- Netzwerkfunktionen (OS/2 Warp Connect)
- lauffähig: DOS-Programme, Windows 3.x-Programme, OS/2-Programme
- echtes Multitasking (nicht für Windows 3.x-Programme)
- lauffähig auf verschiedenen Prozessoren (Intel, PowerPC)
- SMP-fähig (SMP = Symmetric Multiprocessing: Mehrprozessorsysteme mit einem geteilten großen Arbeitsspeicher); für PCs mit bis zu 16 Prozessoren
- "Verbunddokumententechnik" mit OpenDoc
- ab 1996 als erstes Betriebssystem mit Komponenten für Spracheingabe und Sprachsteuerung
- voraussichtlich keine Weiterentwicklung

d) Windows NT

- eigenständiges 32-Bit-Betriebssystem (Microsoft)
- 32 MB Hauptspeicher
- Arbeitsplatzversion im "oberen" PC-Bereich; Haupteinsatz als Serverbetriebssystem
- lauffähig: DOS-Programme, 32-Bit-Windows-Programme
- durchgängig echtes Multitasking
- lauffähig auf verschiedenen Prozessoren (Intel, DEC Alpha-AXP)
- Dateiverwaltungssystem NTFS
- SMP-fähig
- ab Version 4.0 gleiche Benutzeroberfläche wie Windows 95
- neue Version voraussichtlich im Februar 2000 unter dem Namen Windows 2000

e) UNIX, insbes. LINUX

- eigenständiges 32 / 64 -Bit-Betriebssystem
- 4 MB Hauptspeicher (nur Textoberfläche), sonst 16 MB
- Haupteinsatz bei erfahrenen Anwendern und als Serverbetriebssystem
- lauffähig: UNIX-Programme, DOS-Programme, Windows 3.x-Programme
- durchgängig echtes Multitasking
- multiuserfähig
- lauffähig auf verschiedenen Prozessoren (Intel, DEC Alpha, PowerPC)
- Dateiverwaltungssystem (EXT2), andere (FAT 16/32, NTFS, NFS) nutzbar
- SMP-fähig

Anforderungen an moderne PC-Betriebssysteme

- leicht zu bedienende grafische Benutzeroberfläche
- weitgehend automatisierte Installation von CD-ROM
- Tutorial und kontextsensitive Hilfe
- alle Funktionen des BS-Kerns: 32 Bit Verarbeitungsbreite
- Schutz des BS-Kerns, so daß fehlerhafte Applikationen nicht zu einem Systemabsturz führen können (bei Windows 95/98 nicht voll erfüllt)
- echtes Multitasking

- Änderungen ohne Neustarts
- Intelligente Installations- und Deinstallationsfunktionen für Systemprogramme und Anwendungen
- netzwerkfähig
- aktive Wartung und Support durch Hersteller
- Add-On's: Fax-Software, Software für Internet-Zugang,
- Antivirenprogramme...
- preiswert (Einzelplatzlizenz deutlich unter 300 DM)

B. Serverbetriebssysteme

* NetWare (Novell):

- "reines" Netzbetriebssystem: steuert und überwacht die Kommunikation und Zusammenarbeit der an ein Netz angeschlossenen Computer (-> gemeinsame Verwendung von Betriebsmitteln, verteilte Datenverarbeitung im Rahmen von Client-Server-Architekturen, Netzwerkverwaltung)

- Clients unter MS-DOS, Windows (alle Varianten), OS/2, Unix und Macintosh OS

- Marktführer bis 1997 (1997: 30%)

* UNIX

* Windows NT (seit 1998 Marktführer)

* OS/2 Warp

C. Workstation-Betriebssysteme

* UNIX (dominierend)

* Windows NT

* OS/2 Warp

1.2.3.2. Ergänzende Systemsoftware-Komponenten

A. Übersetzungsprogramme

Ein Übersetzungsprogramm ist ein Programm, das Anweisungen in einer (höheren) Programmiersprache (Quellsprache) liest, analysiert und in bedeutungsgleiche Maschinenbefehle (Binärzeichen) umwandelt.

Quellprogramm -> Objektprogramm

Arten von Übersetzungsprogrammen

Assemblierer (Assembler):

übersetzt die in einer maschinenorientierten Programmiersprache (Assemblersprache) abgefaßten Quellenweisungen geschlossen in Zielanweisungen der zugehörigen Maschinensprache.

Kompilierer (Compiler):

übersetzt die in einer problemorientierten Programmiersprache abgefaßten Quellenweisungen geschlossen in Zielanweisungen einer Maschinensprache.

Interpreter:

übersetzt die in einer problemorientierten Programmiersprache abgefaßten Quellenweisungen in Zielanweisungen und läßt sie jeweils sofort ausführen.

B. Dienstprogramme

Hilfsprogramme (Utilities): dienen der Abwicklung häufig vorkommender anwendungsneutraler Aufgaben bei der Nutzung eines Computersystems.

Beispiele: Mischprogramme, Sortierprogramme, Kopierprogramme, Bildschirmschoner, Anti-Virus-Programme

Verwaltungsprogramme: dienen der Verwaltung von Systemkomponenten, z.B. Programme zur Verwaltung von Programm Bibliotheken (Libraries).

Editoren: Programme zum Erstellen, Lesen und Ändern von Dateien (Texte, Quellprogramme,...) im Dialog

C. Middleware

systemnahe Software, die in heterogenen Systemen als zusätzliche Schicht zwischen Betriebssystem(e) und Anwendungssoftware gelegt wird, um Komplexität für den Benutzer zu reduzieren bzw. zu verbergen. Der Benutzer soll über eine einheitliche Benutzeroberfläche auf unterschiedliche Anwendungen zugreifen können, ohne damit belastet zu werden, welche Hardware und/oder welches Betriebssystem sich dahinter verbirgt.

1.3 Vernetzung, Datenmengen, etc.

1.3.1 Vernetzung

Das wird jetzt etwas schwierig, weil im Skript auf der einen Seite relativ wenig dazu steht, auf der anderen Seite aber wieder so "Schlagwörter", die notwendig machen würden, tief einzusteigen. Daher habe ich mich dazu entschieden hier einfach nur das erste Kapietl aus einem - im Internet gefundenen - Skripts abzdrukken (Dann kriegt man so ein Gefühl um was geht und versteht vielleicht das "Ruhland Skript" leichter):

1. 1 Rechnernetze

In den 60er Jahren entstand der Wunsch, Peripheriegeräte entfernt vom Rechner (z.B. am Ort einer Bankfiliale) aufzustellen. Sie wurden über Datenfernübertragungseinrichtungen mit dem Rechner (z.B. in der Zentrale) verbunden und ermöglichten so den Zugang zum entfernten Rechner. Mit solchen Einrichtungen zur Datenfernverarbeitung konnte einerseits auf einen (teueren) eigenen Rechner in der Filiale verzichtet werden, weil lokale Anwendungen der Filiale den Zentralrechner mitnutzen konnten. Andererseits konnten Anwendungen, die die Gesamtorganisation betrafen, Ein / Ausgaben relativ direkt (sozusagen 'On Line') mit örtlich entfernten Endstellen austauschen.

Mit der Verbesserung des Preis / Leistungsverhältnisses der Rechner in den späten 60er Jahren und dem - durch wachsende Anwendungsfelder - steigenden Bedarf nach Rechenleistung ergab sich, daß Organisationen nicht mehr nur eine einzige zentrale Rechnerkonfiguration einsetzten, sondern daß einzelne Filialen und Abteilungen mit eigenen Rechnern ausgestattet wurden, um den Zentralrechner zu entlasten und die für die Datenfernübertragung entstehenden Kosten zu reduzieren. Anwendungen im Problembereich der Verwaltung einer Organisation insgesamt wurden nach wie vor vom Zentralrechner getragen. Lokale Anwendungen konnten isoliert vor Ort abgewickelt werden. Um den Datenaustausch zwischen Zentralrechner und Filialrechnern, später auch zwischen den Filialrechnern zu beschleunigen, war es naheliegend, an die Fernübertragungseinrichtungen auf beiden Seiten einer Verbindung Rechner und nicht Peripheriegeräte anzuschließen. Zunächst ergab sich so eine Sternstruktur mit dem Zentralrechner als Mittelpunkt. Filialrechner waren aus der Sicht des Zentralrechners untergeordnete Peripheriegeräte.

Es wurden schließlich auch (z.B. bei sehr großen Organisationen oder bei Verbänden aus befreundeten Organisationen) Zentralrechner miteinander verbunden. Man kam damit zwangsläufig vom ersten unsymmetrischen Konzept ab. Es entstanden Netze aus gleichrangigen Rechnerknoten und die Koppelung zwischen den

Rechnern konnte nicht mehr dem Rechner-Peripherie-Schema folgen. Mit Beginn der 70er Jahre wuchs ferner auch der Bedarf für Filialrechner-Filialrechner-Datenaustausch. Die Filialrechner wurden ebenfalls zu gleichberechtigten Netzknoten. Es entstanden die ersten großen Rechnernetze.

Hauptziel der Netzbildung war dabei in erster Linie die Datenaustauschmöglichkeit. Dateien eines Rechners konnten mit entsprechender Netzsoftware per Kopie an andere Rechner übertragen werden (Dateitransfer). Es folgten jedoch sehr schnell weitere Netzdienstleistungen, die an den Datenfernverarbeitungsbetrieb angelehnt waren, man konnte z.B. von einem Rechner aus Prozesse an einem anderen erzeugen (Remote Job Entry), Konsolnachrichten übermitteln und entfernte Rechner urladen (Downline Booting).

Die Ziele bei der Realisierung von Rechnernetzen wurden unter den drei folgenden Schlagwörtern zusammengefaßt:

Funktionsverbund; die Vernetzung ermöglicht es Funktionen entfernter Rechner zu nutzen (z.B. komfortabler Zugang zu Rechenleistungen eines Spezialrechners).

-Datenverbund; Daten, die örtlich im Zugriff von Station A sind, können über das Netz auch von B aus zugegriffen werden,

-Lastverbund; zwischen relativ gleichartigen Rechnern können Aufgaben verlagert werden um lokale Engpässe auszugleichen.

Hinzu kam bald der Aspekt des gnädigen Leistungsabfalls (Graceful Degradation) eines Netzes bei Ausfällen einzelner Stationen oder Verbindungen. Der Gesamtbetrieb wird nur teilweise gestört. Nach Ausgliederung der gestörten Teile aus dem Rechnernetz verbleibt ein funktionsfähiges Restsystem (**Zuverlässigkeitsverbund**).

Umgekehrt kann ein Rechnernetz in sehr flexibler Weise erweitert werden. Wenn die Gesamtrechenleistung erhöht werden soll, können einzelne Knoten in verhältnismäßig einfacher Weise neu hinzugefügt oder durch leistungsfähigere ersetzt werden (**flexible Erweiterbarkeit**).

Wenn es dabei weniger auf den Bedarf zukünftiger Anforderungen als vielmehr darauf ankommt, eine aktuell geforderte hohe Rechenleistung zu erbringen, spricht man vom **Leistungsverbund**. Man erhofft sich, daß ein Netz aus preiswerten Standardstationen sehr teure Groß- oder Spezialrechner ersetzen kann.

Die Einsatzgebiete sind unterschiedlich: Verteiltes Hochleistungsrechnen will 'Supercomputer', verteilte Transaktionsbearbeitungssysteme wollen zentrale Datenbasisverwaltende Großrechner ersetzen. Da lange Nachrichtenlaufzeiten zwischen Stationen die Leistung herabsetzen, handelt es sich bei näherem Hinsehen allerdings sehr oft nicht um Rechnernetze sondern um räumlich eng gekoppelte Mehrrechnersysteme.

Im echten (räumlich weit verteilten) Rechnernetz wird, um bzgl. einer bestimmten Anwendung eine hohe Gesamtleistung zu erzielen, die Anwendung selbst dezentralisiert und als verteilte Anwendung entworfen (**Dezentralisierung**). Verteilte Anwendungsprogramme werden dadurch ausgeführt, daß verschiedene Ausführungsprozesse an verschiedenen Orten zueinander nebenläufig ablaufen und gelegentlich miteinander kommunizieren. Durch geeignete Algorithmen können Parallelisierungsbarrieren vermindert werden. Durch günstige Ortszuordnung von Datenbasis-Teilen kann der Kommunikationsbedarf vermindert werden (**Lokalität**).

Verteilte Anwendungen können nicht nur aus der Dezentralisierung ehemals zentraler Anwendungen entstehen. Da die Anwendung nun verschiedene Orte erfaßt, können neue Probleme gelöst werden, nämlich solche, die aus einer Kombination von Datenverarbeitungs- und Datentransport-Problemen bestehen (**Ortsverbund**). Hierzu zählen interaktive Telekooperationsanwendungen und verteilte Informationssysteme.

1. 2 Rechnernetze und verteilte Systeme

Die beiden Begriffe Rechnernetz und verteiltes System werden häufig in gleicher Bedeutung verwendet. A. Tanenbaum trifft eine Unterscheidung, nach der Rechnernetz ein System aus gekoppelten aber autonomen und i.a. auch heterogenen Rechnerknoten bezeichnet, während beim Begriff verteiltes System die technische Zusammensetzung des unterliegenden Rechnernetzes wie überhaupt seine Gliederung in einzelne autonome Knoten verdeckt wird. In extremer Sicht ist ein verteiltes System -wie eine traditionelle isolierte Rechnerkonfiguration - eine einzige homogene Maschine mit der Besonderheit, daß unterschiedliche Orte vorhanden sind und Prozesse und Daten an Orten lokalisiert sind.

Eine andere Unterscheidung versteht unter Rechnernetz wie oben ebenfalls ein System aus einzelnen gekoppelten Rechnerknoten, das relativ zweckneutral als besondere Form einer Basismaschine angesehen wird. Beim Begriff verteiltes System steht dagegen ein genauer umrissener Systemzweck im Vordergrund. Ein verteiltes System ist eine Rechneranwendung, bei der Daten und Prozesse an verschiedenen Orten vorhanden sind (verteilte Anwendung). Verteilte Anwendungen werden mittels Rechnernetzen implementiert, so daß die beiden Begriffe verteilte Anwendung und Rechnernetz-Anwendung synonym sind.

Die in der ersten Unterscheidung hervorgehobene Homogenität der Basismaschine 'verteiltes System' sollte nicht überbewertet werden. Sie verspricht zwar eine Vereinfachung des Anwendungsentwurfs, weil Details des aus verschiedenen Stationen bestehenden Rechnernetzes verdeckt werden. Beim Entwurf praxisrelevanter verteilter Anwendungen kann der Rechnernetz-Charakter der Basismaschine aber kaum vernachlässigt werden. Dem stehen die Randbedingungen der technischen Realisierung, d.h. Kosten- und Leistungsaspekte entgegen. Ursachen hierfür sind:

Lose Kopplung; der Datenaustausch zwischen entfernten Rechnern verursacht nicht-vernachlässigbare Kosten; die Laufzeiten für Nachrichten sind verhältnismäßig lang; die Verbindungen können jederzeit getrennt werden; durch die lange Laufzeit können sich zufällige Störeinflüsse viel häufiger auswirken; die Wahrscheinlichkeit dafür, daß Signale auf dem Weg verfälscht werden, kann nicht vernachlässigt werden; die Zeit, die benötigt wird, um eine Nachricht unverfälscht zu übertragen, ist nicht vorhersagbar.

Nebenläufigkeit; es gibt mehrere Stationen, die eigenständig und zeitlich simultan Verarbeitungsprozesse ausführen können. Zur Ausnutzung dieser Fähigkeit sollte eine Problemlösung so entworfen werden, daß sie durch eine Menge schwach synchronisierter Prozesse erbracht wird.

Dezentrale Kontrolle; die Abläufe in den einzelnen Stationen eines verteilten Systems werden möglichst voneinander entkoppelt. Jede

Station steuert ihre eigenen Abläufe soweit als möglich selbst (durch eine lokale Betriebsorganisation). Man verzichtet darauf, daß an einem Ort eine vollständige und aktuelle Sicht des globalen Systemzustands verfügbar gemacht wird. Jede Station hat von den Zuständen an den anderen Stationen nur eine Teilsicht, insbesondere können auch an einzelnen Stationen Ausfälle auftreten, ohne daß dies sofort an den anderen Stationen erkennbar wird.

Diese Eigenschaften bedingen, daß beim Entwurf eines verteilten Systems Kommunikationskosten, -leistung und -störungen sowie die Unabhängigkeit der Abläufe an unterschiedlichen Stationen berücksichtigt werden müssen, wenn effiziente und stabile Lösungen erzielt werden sollen.

Für verteilte Systeme sind ferner folgende Eigenschaften typisch und schlagen sich im Entwurf nieder:

Systemlebensdauer; während eine typische lokale Anwendung innerhalb weniger Stunden terminiert, soll ein verteiltes System oft ein sich in der Realität sehr langfristig stellendes Problem lösen. Ein Flugbuchungssystem sollte z.B. das überhaupt nicht zeitlich terminierte Problem lösen, alle zukünftigen Flüge der angeschlossenen Flugesellschaften den angeschlossenen Reisebüros buchbar zu machen. Es sollte auf Jahre hinaus arbeiten, ohne daß - die in diesem Zeitraum mit Sicherheit auftretenden - Komponentenausfälle die Systemfunktion stören. Fehler- und Ausfalltoleranzaspekte müssen also in den Entwurf einbezogen werden.

Systemgröße; die hohe Systemgröße (Anzahl der Komponenten, der enthaltenen Funktionen, die logische Verflechtung der Funktionen) kann im Entwurf nur durch sorgfältige Modularisierung bewältigt werden. Im Verbund mit der hohen Lebensdauer muß damit gerechnet werden, daß sich Aufgabenstellung und geeignete Systemauslegung im Lauf der Betriebszeit ändern. Das verteilte System sollte also modular ausgelegt und in Modulen (zur Laufzeit) flexibel in seiner Konfiguration veränderbar sein.

1. 3 Rechnernetze und Telekommunikationssysteme

Ein Rechnernetz besteht aus einer Menge von Rechnern, die örtlich voneinander entfernt installiert sind, und aus Einrichtungen, die den Austausch von Daten zwischen den Rechnern ermöglichen.

Vernachlässigen wir, daß jeder einzelne Rechner auch lokal unabhängig vom Gesamtnetz betrieben werden kann, und vernachlässigen wir zunächst auch, daß das Rechnernetz insgesamt als Basismaschine für Rechneranwendungen - dann eben Rechnernetz-Anwendungen - betrachtet werden kann, dann ergibt sich eine Sicht, in der an verschiedenen Orten (den Standorten der einzelnen Rechner) durch dort lokale Vorgänge einerseits Daten erzeugt werden, die an andere Orte zu übertragen sind, und andererseits entfernt erzeugte Daten gelesen werden sollen. Das Rechnernetz ist ein Telekommunikationssystem. Daten werden nicht nur verarbeitet, sondern auch über (oft weite) Strecken transportiert.

Im Vergleich zu traditionellen Telekommunikationssystemen (z.B. Telephon-Dienst, Brief-Dienst) werden im Rechnernetz Informationen digital codiert und transportiert, und die systeminternen Abläufe zur Aufbereitung der Daten und Kontrolle ihres Austauschs können mit allen Vor- und Nachteilen programmgesteuert stattfinden. Die Vorteile überwiegen aus technischer Sicht. So besteht die Tendenz, auch Nicht-Rechner-Rechner-Kommunikation durch digitale Telekommunikationssysteme zu unterstützen (z.B. Telefax, BTX, digitales Telephonieren, Einführung von ISDN). Es bestehen zwei Beziehungen zwischen Rechnernetzen und (modernen digitalen) Telekommunikationssystemen:

- Ein Rechnernetz enthält und nutzt als wichtige systemtragende Bestandteile Telekommunikationssysteme, - Moderne Telekommunikationssysteme werden per Rechnernetzen realisiert.

Die (im wesentlichen auch mit internationalen Standardisierungsarbeiten einhergehende) Begriffsbildung zu und Betrachtungsweise von Telekommunikationssystemen hat deshalb großen Einfluß auf Konzeption und Architektur von Rechnernetzen.

Ein großer Teil des Vorlesungsstoffs ist also dem Begriff 'Telekommunikationssystem' und der Implementierung von Telekommunikationssystemen mittels vernetzter Rechner gewidmet. Als Zweck des Telekommunikationssystems / Rechnernetzes wird die Erbringung von (Tele-) Kommunikationsdienstleistungen in den Vordergrund gestellt. Der eigentliche, produktive Zweck eines Rechnernetzes, der darin besteht, (verteilte) Anwendungen auszuführen, tritt noch in den Hintergrund, wird aber nicht unsichtbar, weil es wiederum der Zweck der Kommunikationsdienste ist, eben die

Kommunikationsanforderungen der verteilten Anwendungen zu erfüllen.

1. 4 Rechnernetze und Datennetze

Häufig werden Netze mit den Kürzeln LAN (für Local Area Network bzw. lokales Netz), WAN (für Wide Area Network bzw. Weitverkehrsnetz oder geographisch verteiltes Netz) und MAN (für Metropolitan Area Network) bezeichnet. Auch das Kürzel DAN (für Desktop Area Network bzw. Schreibtisch-Netz) tritt schon stellenweise auf. Man erkennt eine Hierarchie vom räumlich sehr eng gefaßten DAN über LAN und MAN zum WAN.

Der Sprachgebrauch unterscheidet hier nicht zwischen Rechnernetz und reinem Kommunikationssystem, das mit Datennetz bezeichnet wird. Trotzdem sollte man sich dessen bewußt sein, daß ein Rechnernetz, das produktive Anwendungen tragen soll und bei dem es auf die integrale Funktionalität aus Datenverarbeitung und

Transport ankommt, mehr ist als ein Datennetz, dessen Aufgaben sich auf die dienstleistende Zurverfügungstellung von Kommunikationsmöglichkeiten beschränkt. Wenn z.B. das Kürzel LAN aus Kommunikationssystem-Sicht benutzt wird, hat es eine andere Bedeutung als in der Rechnernetz-Sicht.

Das Kürzel VAN (für Value-added Network bzw. Netz mit Mehrwertdiensten, manchmal auch intelligentes Netz genannt) steht übrigens dazwischen. Es ist aus der Sicht der Telekommunikationssysteme geprägt, d.h. es geht davon aus, daß "Netz" ein Datennetz bezeichnet, und hebt aus dieser Sicht hervor, daß auch datenverarbeitende Funktionalität (Informationssystem-Funktionen: Auskunftsdienste, Verzeichnisdienste, Datenhaltungsmöglichkeiten, Vorgangsverwaltung) zusätzlich zur Kommunikationsunterstützung im Netz vorhanden ist und den Netznutzern angeboten wird.

Und noch was "geklautes":

Rechnernetz

(Netz, Netzwerk; computer network)
räumlich verteiltes System von Computern und peripheren Geräten, die durch Datenübertragungseinrichtungen und Übertragungswege miteinander verbunden sind.

Klassifikation von Rechnernetzen nach der Reichweite:

Lokale Netze (LAN - Local Area Network)
interne Netzwerke einer Organisation
innerhalb desselben Standortes (Gebäude, Firmengelände)
nichtöffentliche Übertragungswege
B.: PC-Netze, Großrechner mit Terminals, Client-Server-Architekturen, digitale Nebenstellenanlagen
Fernnetze (WAN - Wide Area Network)
MAN (Metropolitan Area Network)
GAN (Global Area Network)
öffentliche Netzwerke und interne Netzwerke standortübergreifend agierender Organisationen (⇒ Corporate Networks)
standortübergreifend (Überbrückung geographischer Entfernungen)
i.a. öffentliche Übertragungswege
B.: Internet, Datex-P-Netz der Telekom, interne Netze multinationaler Unternehmen, VSAT-Netz für Satellitenübertragungsdienste, zellulare Funknetze

Ziele der Vernetzung

Kommunikationsverbund
Datenverbund (⇒ Bereitstellung und Zugriff auf Informationen)
Programmverbund (⇒ Nutzung von Anwendungssoftware in Client-Server-Umgebungen)
Geräteverbund; Sicherheitsverbund
Lastverbund; Leistungsverbund
Nutzerverbund (⇒ Computerunterstützung von Gruppenarbeit)

1.3.2 Datenmengen, Konvention zur Binär-Codierung

Worum es hier geht ist - glaube ich - folgendes: Ich kann ja mit einem Byte (einer Speicherzelle) genau 256 "Werte" darstellen. Es gibt aber allein schon bei den ganzen Zahlen mehr mögliche Zahlen als bloß 256. (z.B. 10.000) Was mache ich also? Ich nehme zwei oder drei oder... Speicherzellen "hintereinander" und kann so **immer mehr** darstellen. Das ist irgendwo logisch, weil ich ja unterscheiden kann, ob "01000001" von Speicherzelle eins oder zwei kommt. Naja, und ich muss mich eben auch drauf einigen, **welcher** "Wert" jetzt z.B. ein "ü" sein sollen. Also **Konventionen**, "Codes" entwickeltn. Und wenn jetzt jemand mit **japanischen** Schriftzeichen schreiben will, muss ich mir halt wieder was **neues** ausdenken... Und log.weise denken sich viele Menschen **viele untersch. Konventionen** aus.

Und **Bilder** (am Bildschirm) bestehen aus Punkten. Will ich, das mein Bild einigermaßen gut aussieht, brauche ich **viele sehr kleine** Punkte (Pixel). Und dann ist auch logisch, das **Bild-Dateien** immer **recht groß** sind: In der Datei muss ja **jeder** Punkt von der Farbe her definiert sein. Und dann gibt es halt verschiedene **Bild-Datei-Formate**, die Bilder unterschiedlich "groß" (Speicherplatz) und in unterschiedlicher Qualität abspeichern können. Das hängt halt davon ab, welche "Tricks" die anwenden (z.B. benachbarte Pixel einer Farbe zu einem großen zusammenfassen etc.). gängige Formate sind: .bmp (groß & schlechte Qualität), jpeg (rel. klein & rel. gute Qualität), .tiff (rel. groß & gute Qualität) ... Und **Video oder Musik-Dateien** werden halt sehr klein, wenn man die als mpeg- Datei speichert.

Lokale Netze

ermöglichen Kommunikation zwischen mehreren unabhängigen Datenstationen mit hoher Übertragungsgeschwindigkeit und niedriger Fehlerrate über private Übertragungswege im Nahverkehrsbebereich
Gewährleistung der Funktionalität durch spezielle Netzbetriebssysteme oder Netzwerkkomponenten von Betriebssystemen
Notwendigkeit einer Netzwerkadministration:
Konfigurationsmanagement
Störungsmanagement
Leistungsmanagement
Benutzermanagement
Abrechnungsmanagement
Sicherheitsmanagement

Fernnetze und Dienste der Telekommunikation Kommunikationsnetze im Weitverkehrsbereich:

Verteilnetze

alle Netzstationen erhalten die gleichen Informationen und entscheiden lokal, ob und welche Informationen sie erhalten möchten;
Beispiel: Fernsehkabelnetz
⇒ *Verteildienste* (TV-Broadcasting, Pay-TV,...)

Vermittlungsnetze

jede Netzstation erhält nur die Informationen, die sie selbst angefordert hat, oder die an sie gesendet wurden;
Beispiel: ISDN
⇒ *Vermittlungsdienste* (Telefondienst, Telefax,...)