

Einführung

Betriebssysteme WS 2011/2012



Jörg Kaiser
IVS – EOS

Otto-von-Guericke-Universität Magdeburg

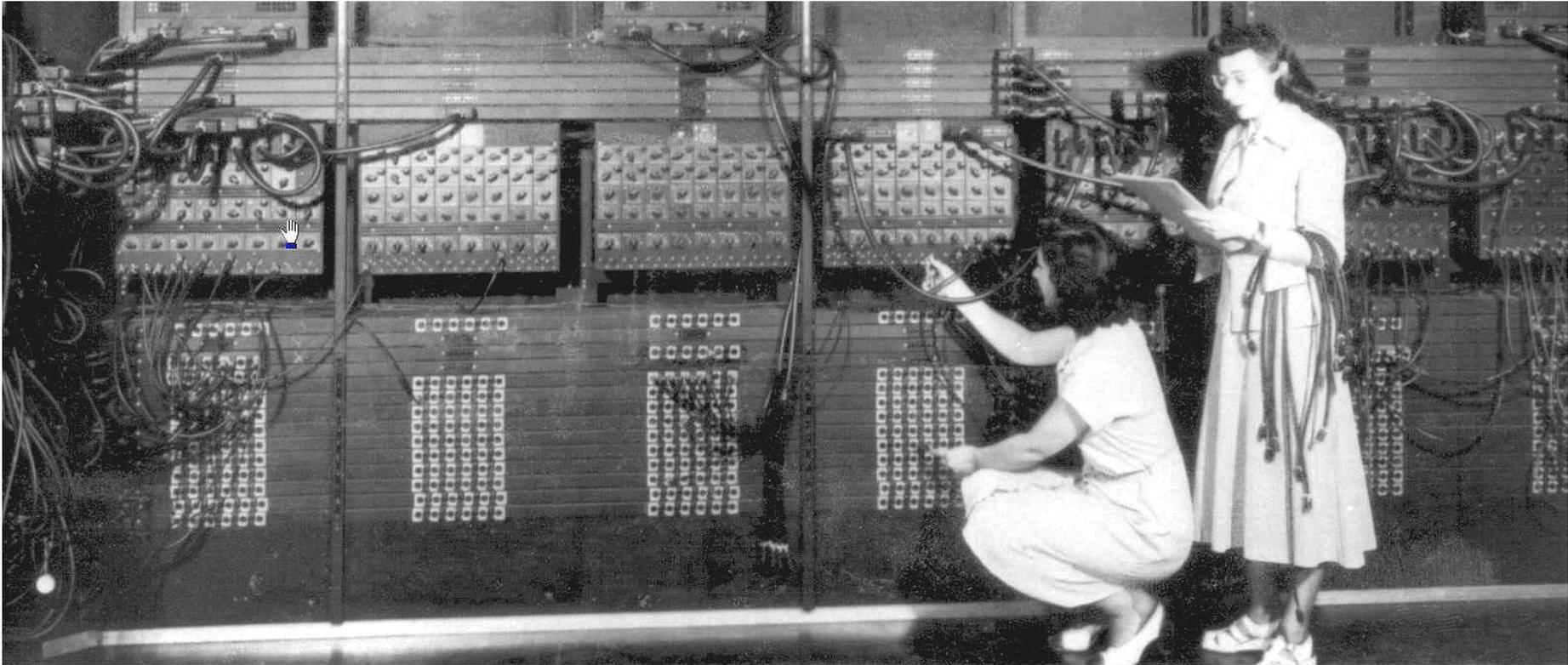
Preisfragen:

Was ist ein Betriebssystem?

Wozu braucht man ein Betriebssystem?



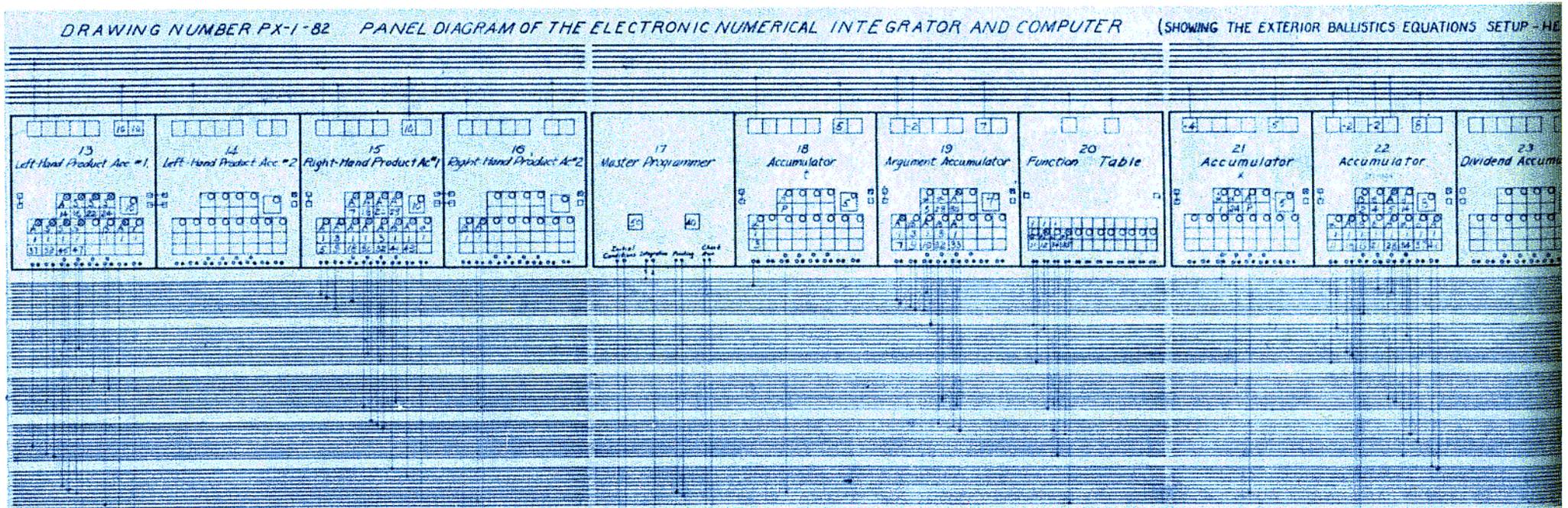
Programmierung des ENIAC (1945)



Gloria Gordon and Ester Gerston at work on the ENIAC



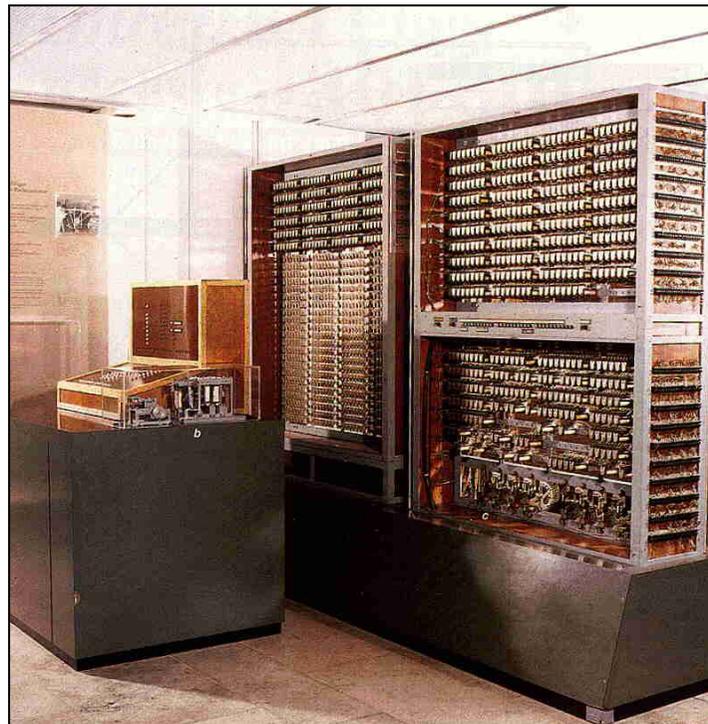
Programmierung des ENIAC



Die Anwendungen der frühen Rechner hatten keine Betriebssystemunterstützung.

Welche Konsequenz hat das Fehlen eines Betriebssystems?

Gibt es auch heute noch Anwendungen, bei denen kein BS benötigt wird?

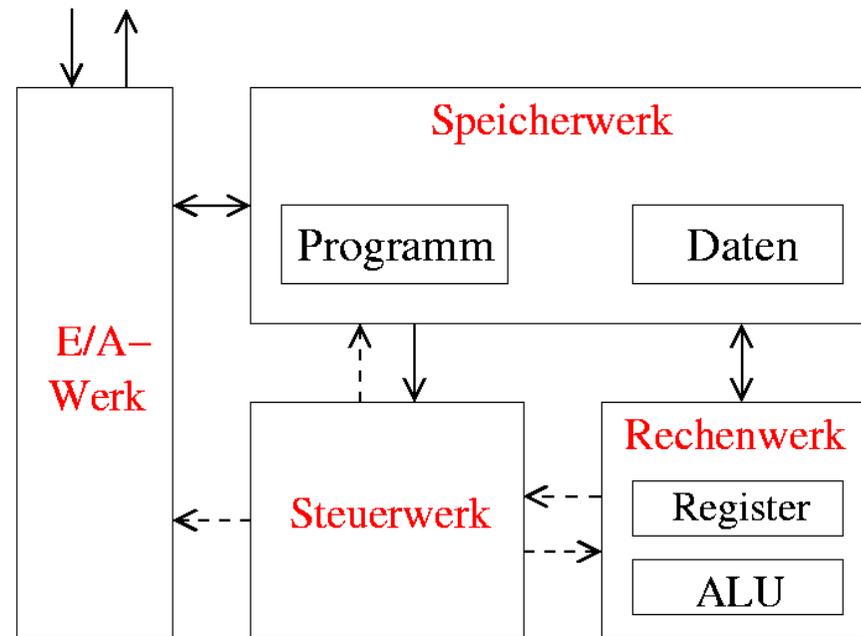


Zuse Z1: 1941



seit 1945: Der "Stored-Program-Computer" (John von Neumann)

- **Speicherwerk** enthält Programme und Daten
- **Rechenwerk** mit ALU und Registern
- zentrales **Steuerwerk** mit „*fetch-decode-execute*“ Instruktions-zyklus
- **Ein-/Ausgabe-Werk** mit Datenumwandlung



Probleme für den Programmierer

Programmieren auf der "nackten" CPU-Hardware wie bei ENIAC ist aufwändig, fehleranfällig und nicht auf andere CPUs zu übertragen.

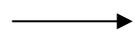
Häufig benutzte Funktionen, z.B. Eingabe und Ausgabe müssen für jede Anwendung neu programmiert werden.

Es besteht im Allgemeinen eine große Diskrepanz zwischen den Eingabe und Ausgabegeschwindigkeit und der Arbeitsgeschwindigkeit des Prozessors.

Keine Unterstützung um mehrere Anwendungen im Speicher zu halten und auszuführen.



Bessere Hardware/Software-Schnittstelle



Unterstützung einer Ausführungsumgebung für Anwendungen



Definition: Betriebssystem

BS als Schnittstellen- Manager

Hofstadter *The operating system is itself a program which the functions of shielding the bare machine from access (thus protecting the system), and also of insulating the programmer from the many extremely intricate and messy problems of reading the program, calling a translator, running translated program, directing the output to the proper channels the proper time, and passing control to the next user.*

- **Silberschatz/Galvin** Ein Programm, das als *Vermittler zwischen Rechnernutzer und Rechnerhardware* fungiert. Der Sinn des Betriebssystems ist es, eine Umgebung bereitzustellen, in der Benutzer bequem und effizient Programme ausführen können.



Definition: Betriebssystem

BS als Ausführungsumgebungs-Manager

- **Lexikon der Informatik** *Summe derjenigen Programme, die als residenter Teil einer EDV-Anlage für den Betrieb der Anlage und für die Ausführung der Anwenderprogramme erforderlich ist.*
- **DIN 44300** *Die Programme eines digitalen Rechensystems, die zusammen mit den Eigenschaften der Rechenanlage die Grundlage der möglichen Betriebsarten des digitalen Rechensystems bilden und insbesondere die Abwicklung von Programmen steuern und überwachen.*



Definition: Betriebssystem

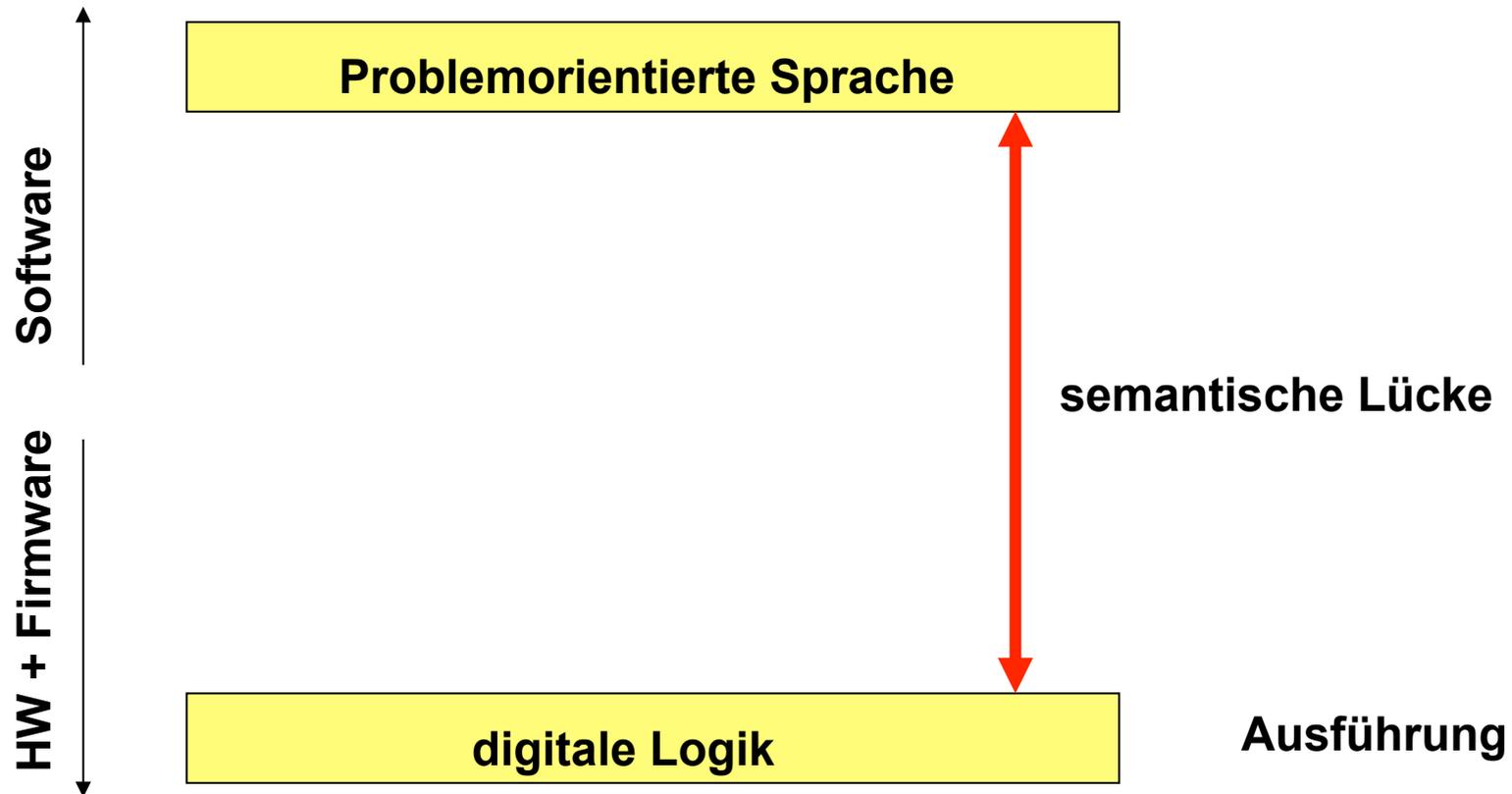
BS als (virtuelle) Maschine

Tanenbaum *Eine Softwareschicht, die alle Teile des Systems verwaltet und dem Benutzer eine Schnittstelle oder eine Maschine anbietet, die einfacher zu verstehen und zu programmieren ist [als die nackte Hardware].*

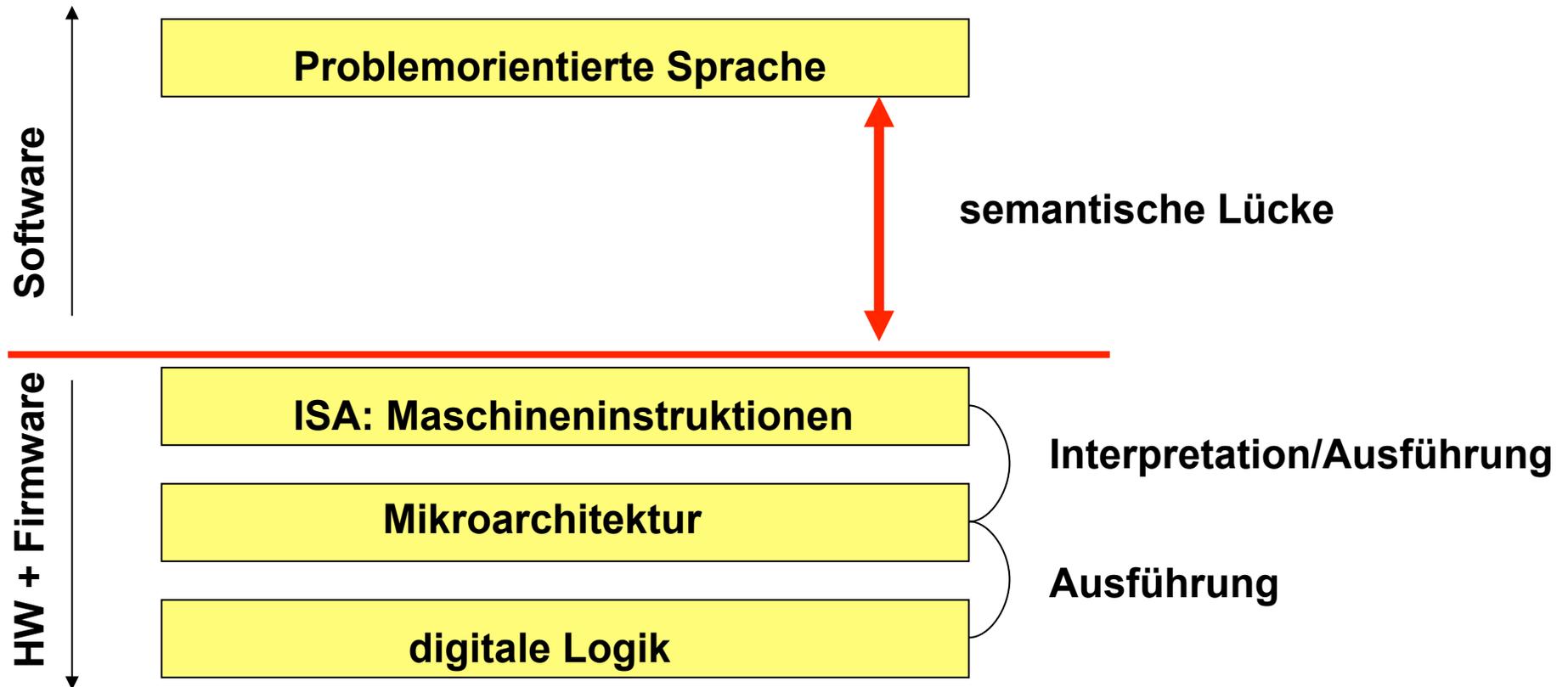
Kittler *Ein Betriebssystem kennt auf jeden Fall keinen Prozessor mehr, sondern ist neutral gegen ihn, und das war es vorher [d. h., bevor die Schnittstelle zum real existierenden Prozessor softwaremäßig formuliert wurde] noch nie. Und auf diese Weise kann man eben jeden beliebigen Prozessor auf jedem beliebigen anderen emulieren, wie das schöne Wort lautet.*



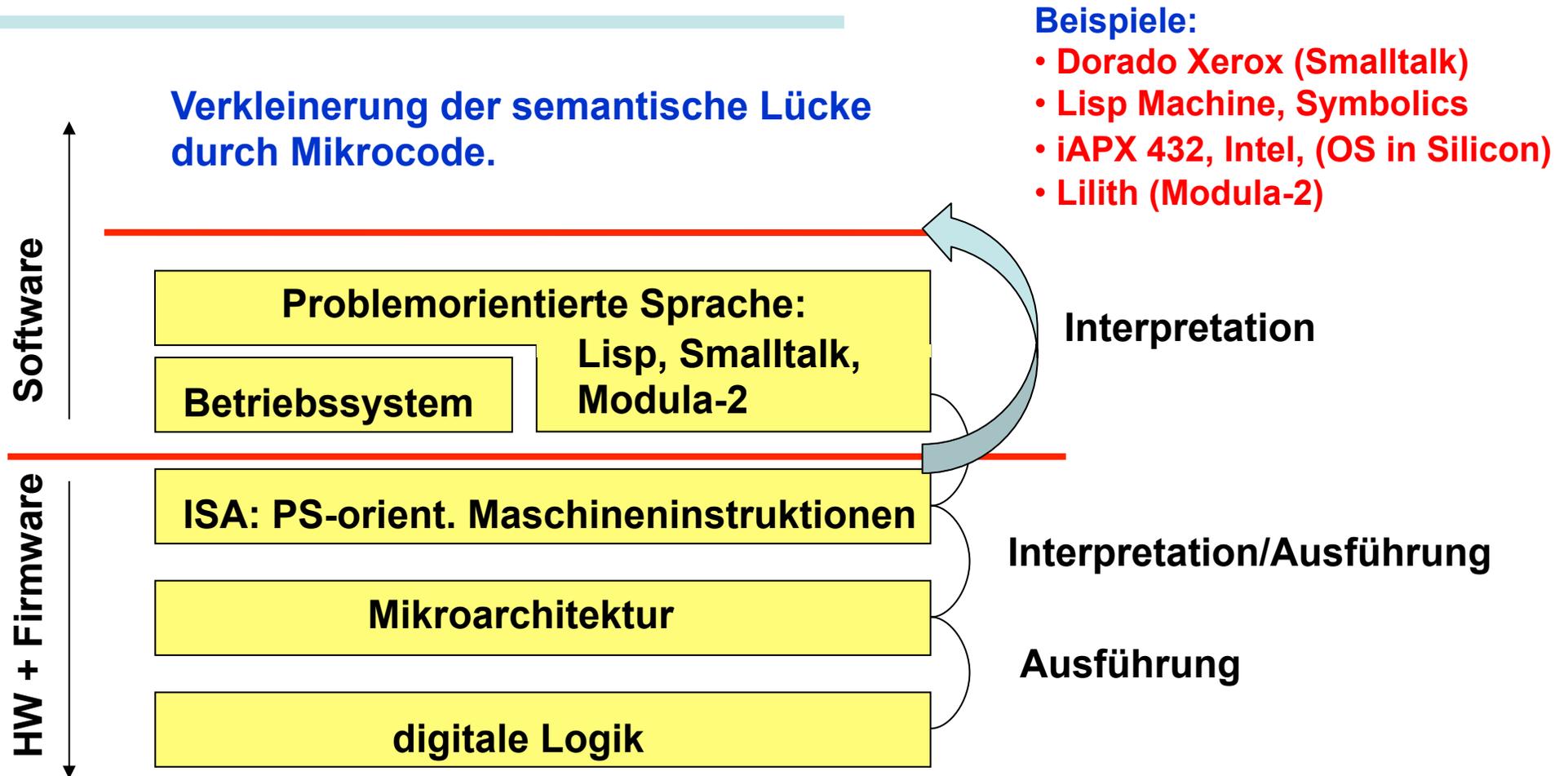
Ein Computersystem als Hierarchie virtueller Maschinen



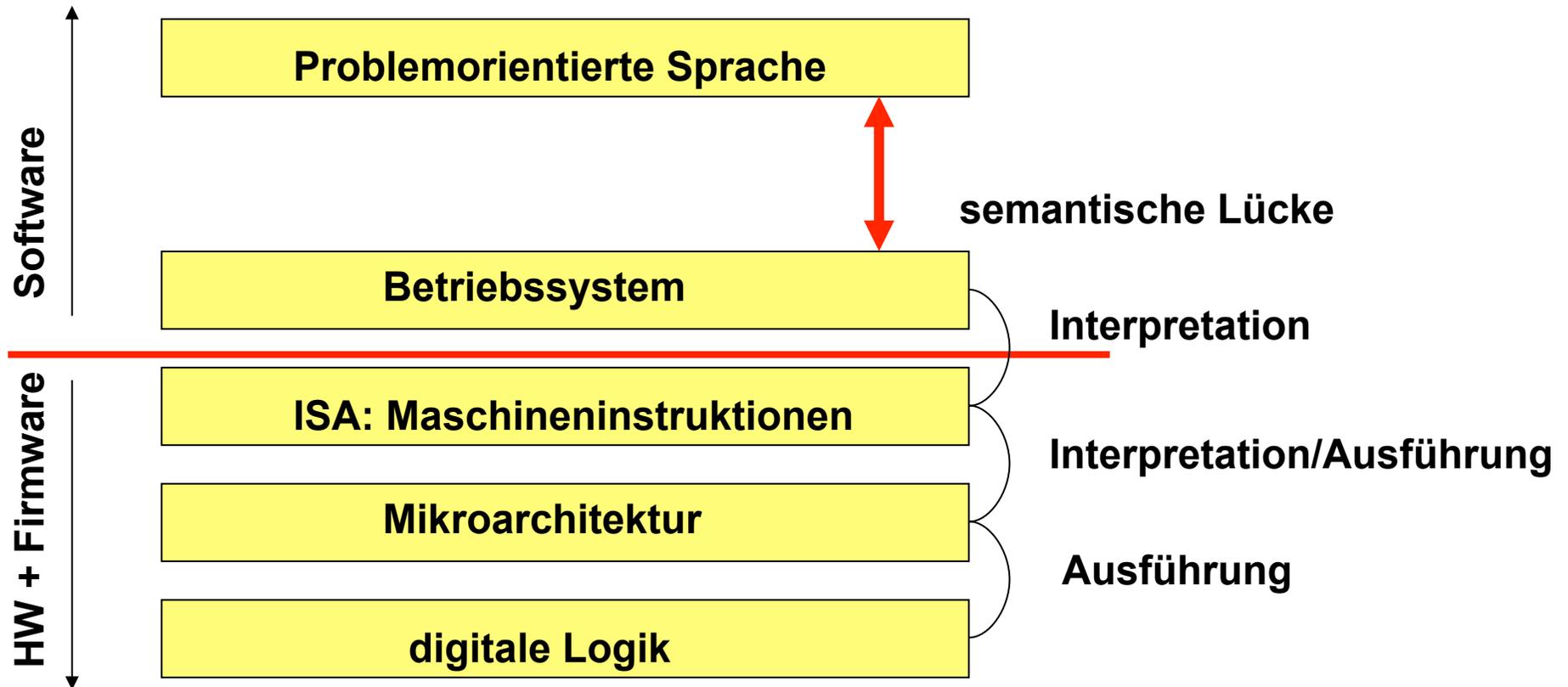
Ein Computersystem als Hierarchie virtueller Maschinen



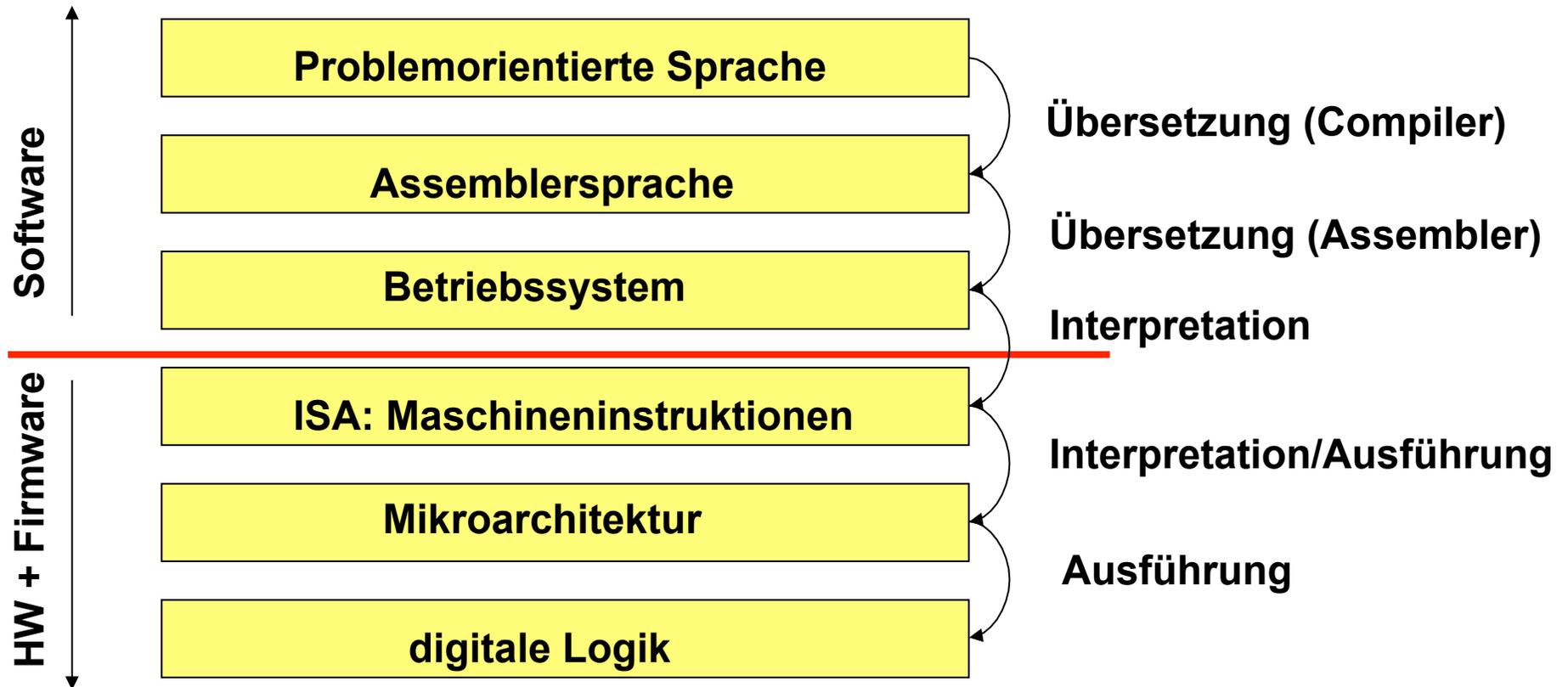
Ein Computersystem als Hierarchie virtueller Maschinen



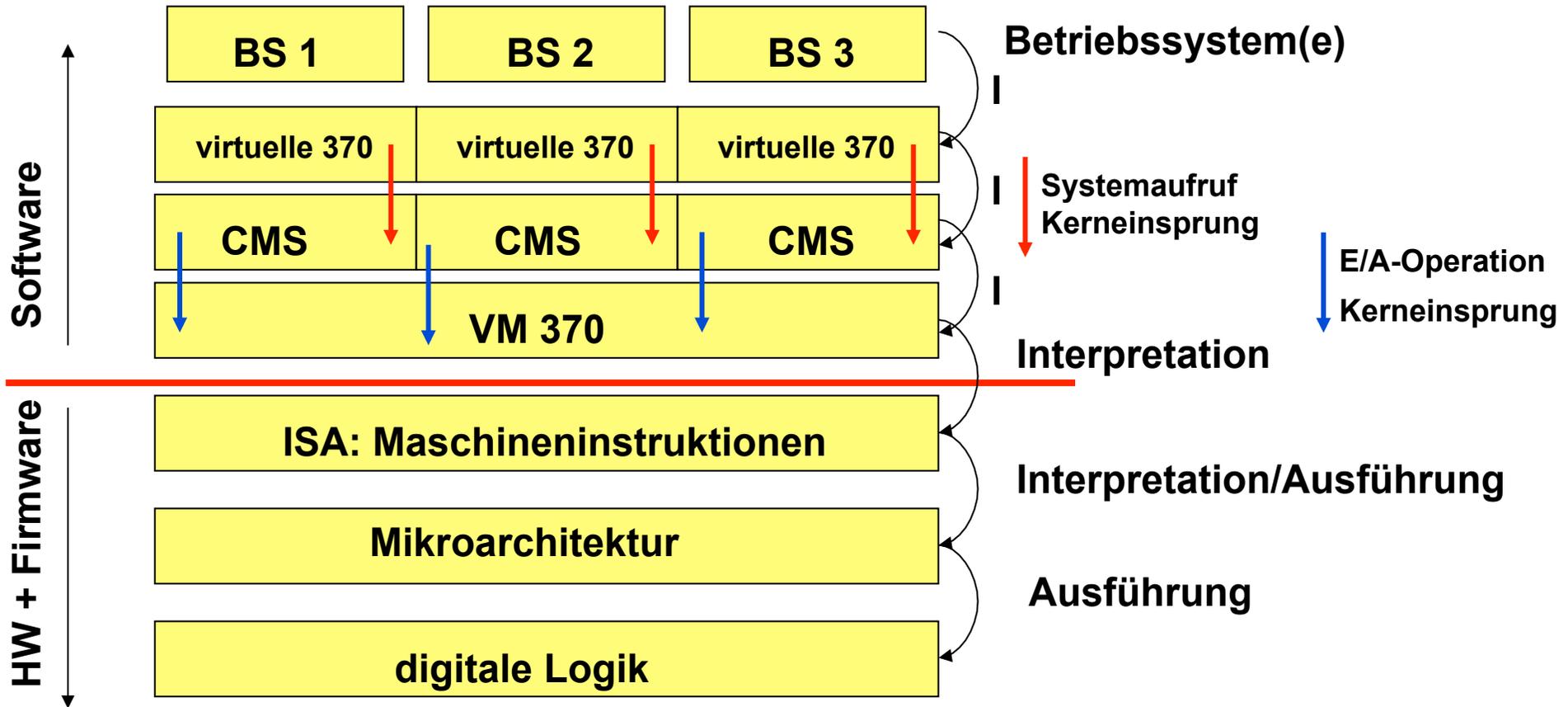
Ein Computersystem als Hierarchie virtueller Maschinen



Ein Computersystem als Hierarchie virtueller Maschinen



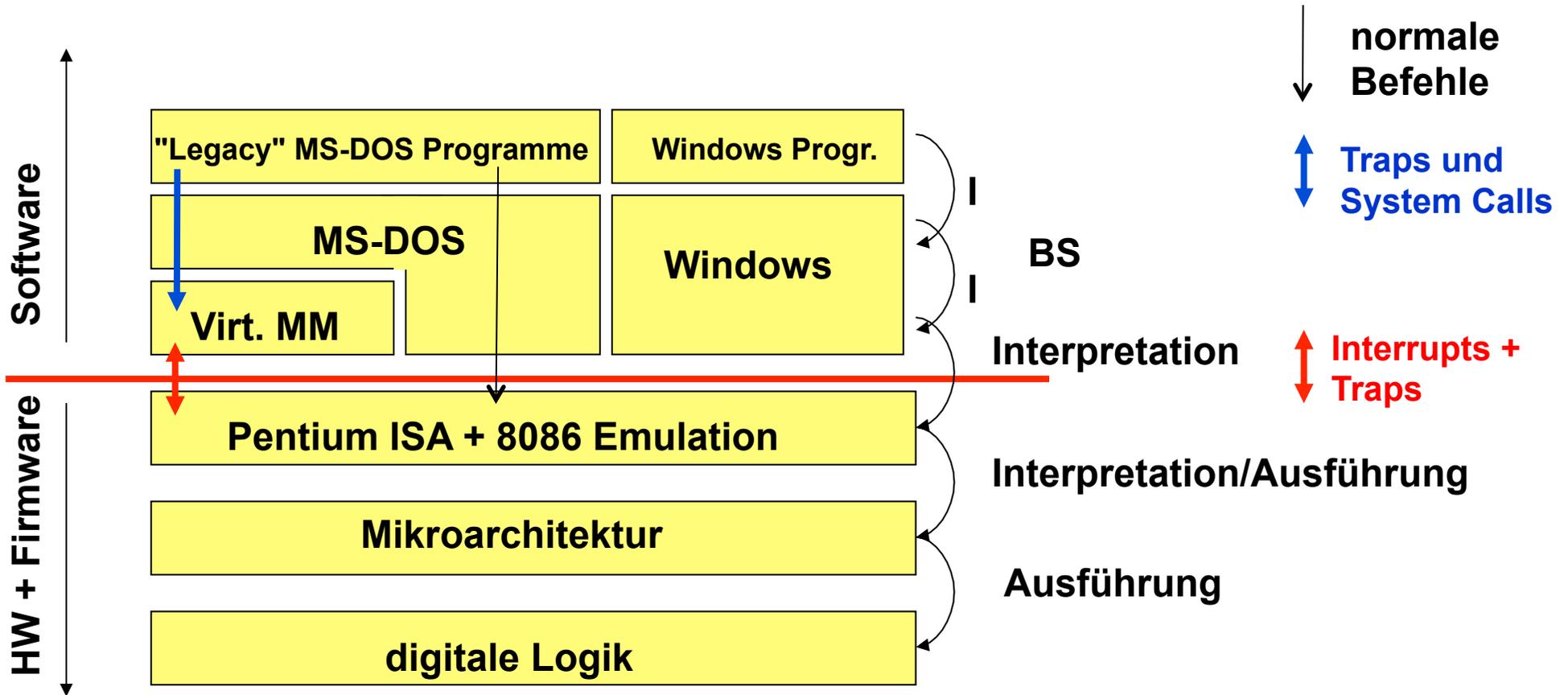
Hierarchie virtueller Maschinen im IBM VM/370-System mit CMS



CMS: Conversational Monitor System



Hierarchie virtueller Maschinen zur 8086 Emulation



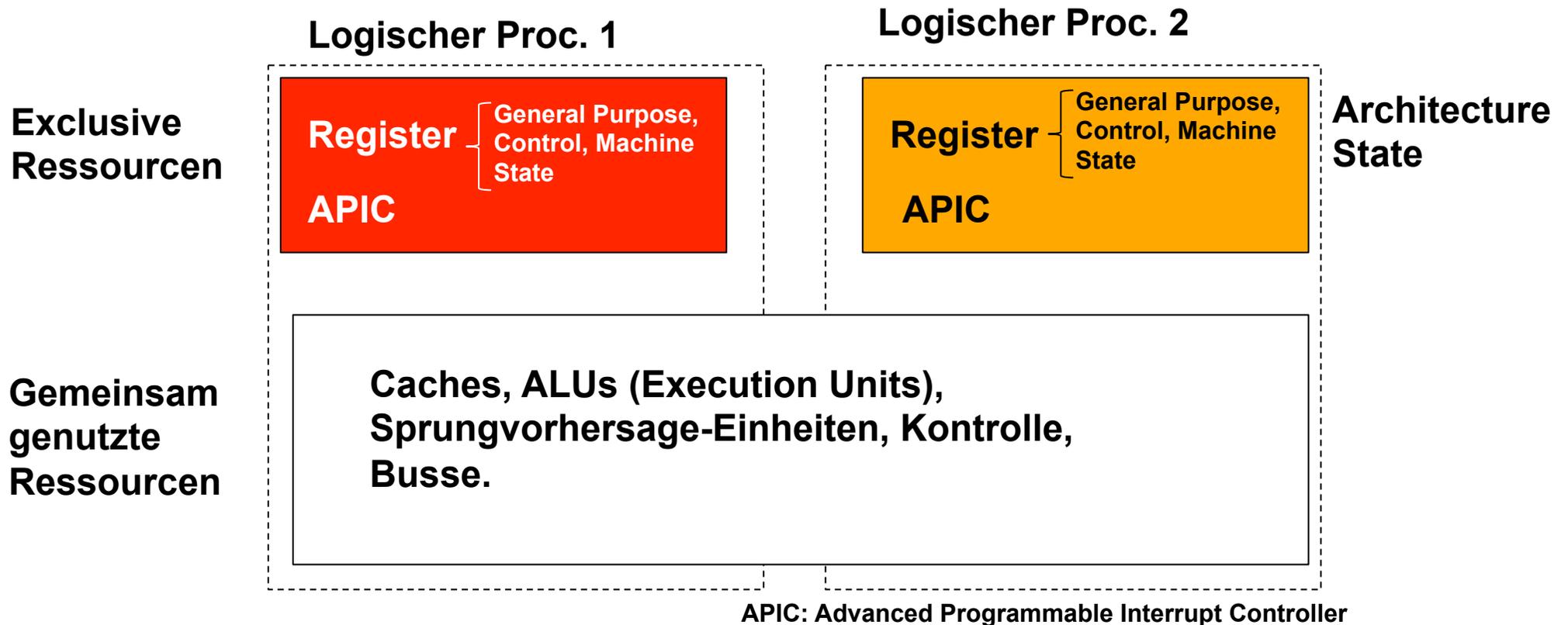
Virt. MM: Virtual Machine Monitor, emuliert 8086 Trap+Interruptverhalten



Zur Erinnerung: Parallelität auf Thread-Ebene (RS)

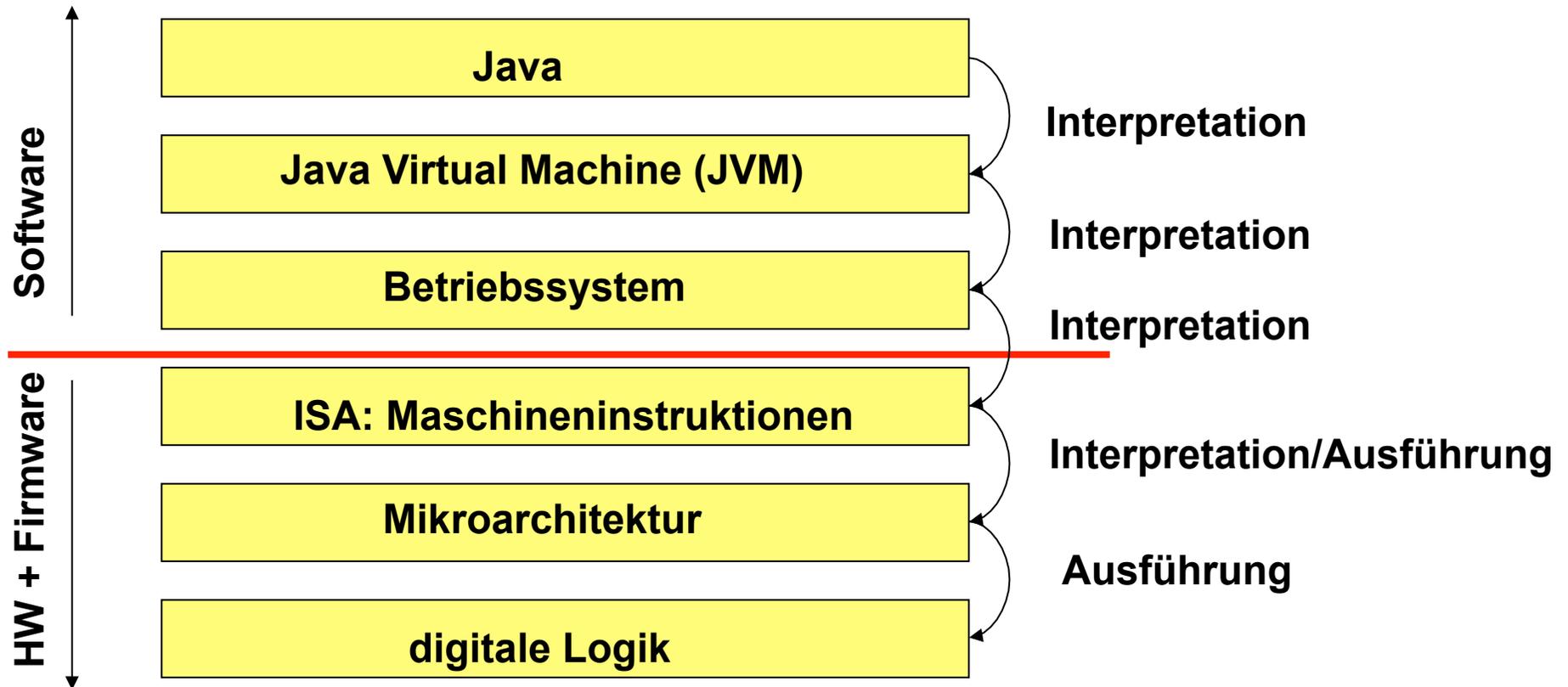
Motivation: Pipeline-Verzögerungen wie Speicherzugriffe, Cache-Fehler, falsche Sprungvorhersage, und Datenabhängigkeiten

Idee: Mehrere logische Prozessoren



Ein Computersystem als Hierarchie virtueller Maschinen

Beisp. JVM

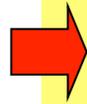


Vorläufer: BASIC, Modula 2.

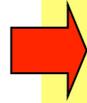


Aufgaben und Ziele eines Betriebssystems

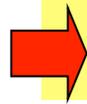
Aufgaben



Anpassung der Maschinenwelt an die Benutzerwelt



Organisation und Steuerung des Betriebs:
Verwaltung aktiver und passiver Ressourcen
Einhaltung von Leistungsanforderungen
Unterstützung bei Fehlern und Ausfällen



Langfristige Datenhaltung

Ziele



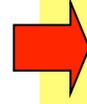
Bequemlichkeit



Ermöglichen von Nebenläufigkeit



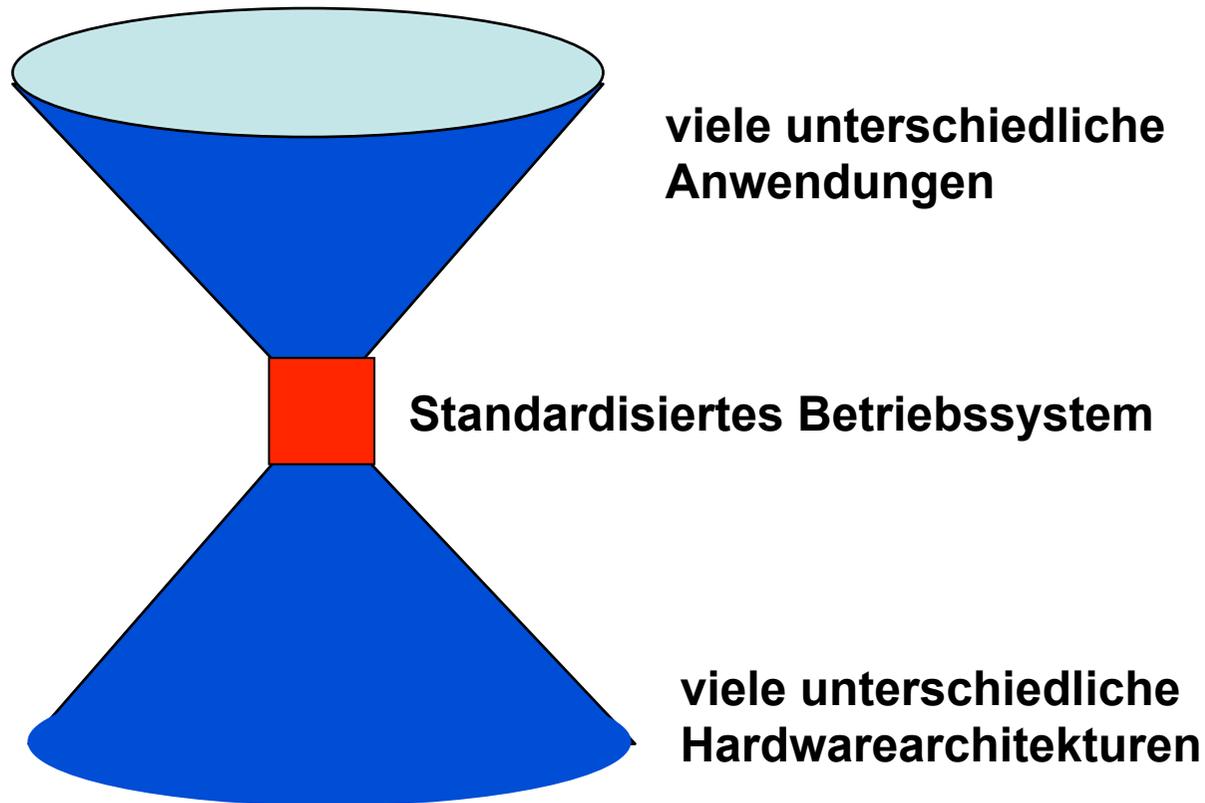
Effizienz, Portierbarkeit



Fähigkeit zur Erweiterbarkeit und Skalierbarkeit



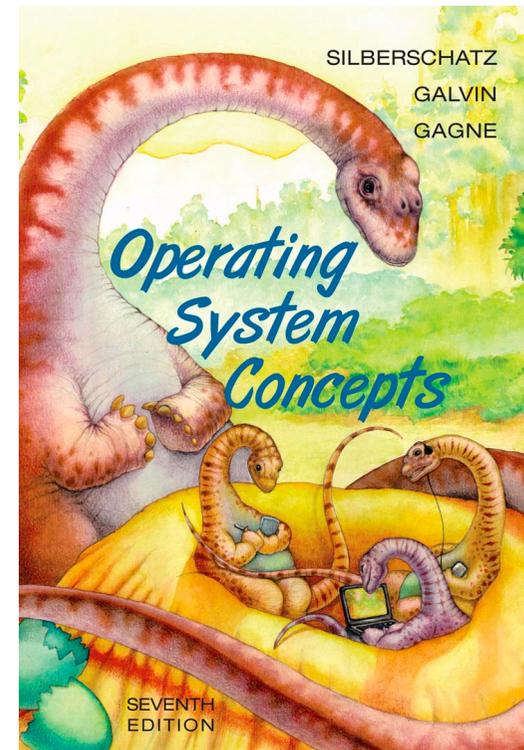
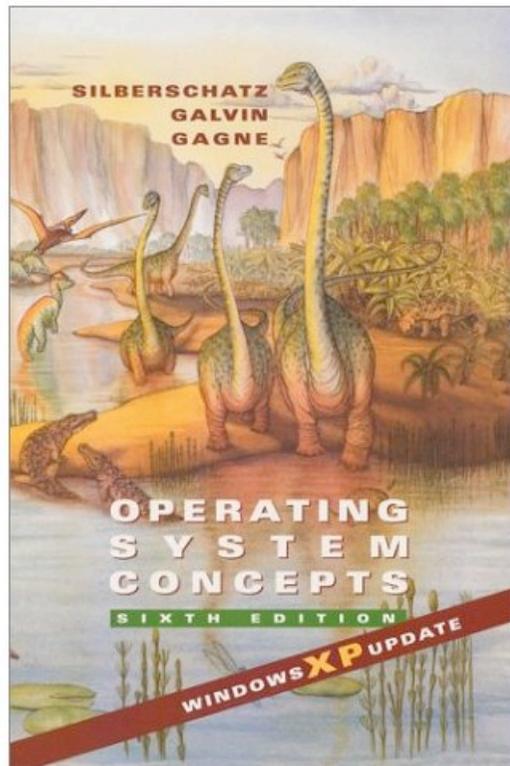
Stundenglasmodell



**BS maskiert
Heterogenität der
Hardware.**



Entwicklung der Betriebssysteme



Entwicklung der Betriebssysteme

Die Fünziger

Ein Programm wird von einem Prozessor abgearbeitet.

Stapel-Verarbeitung (Batch-Processing)

Die Betriebssystemfunktion beschränkt sich auf:

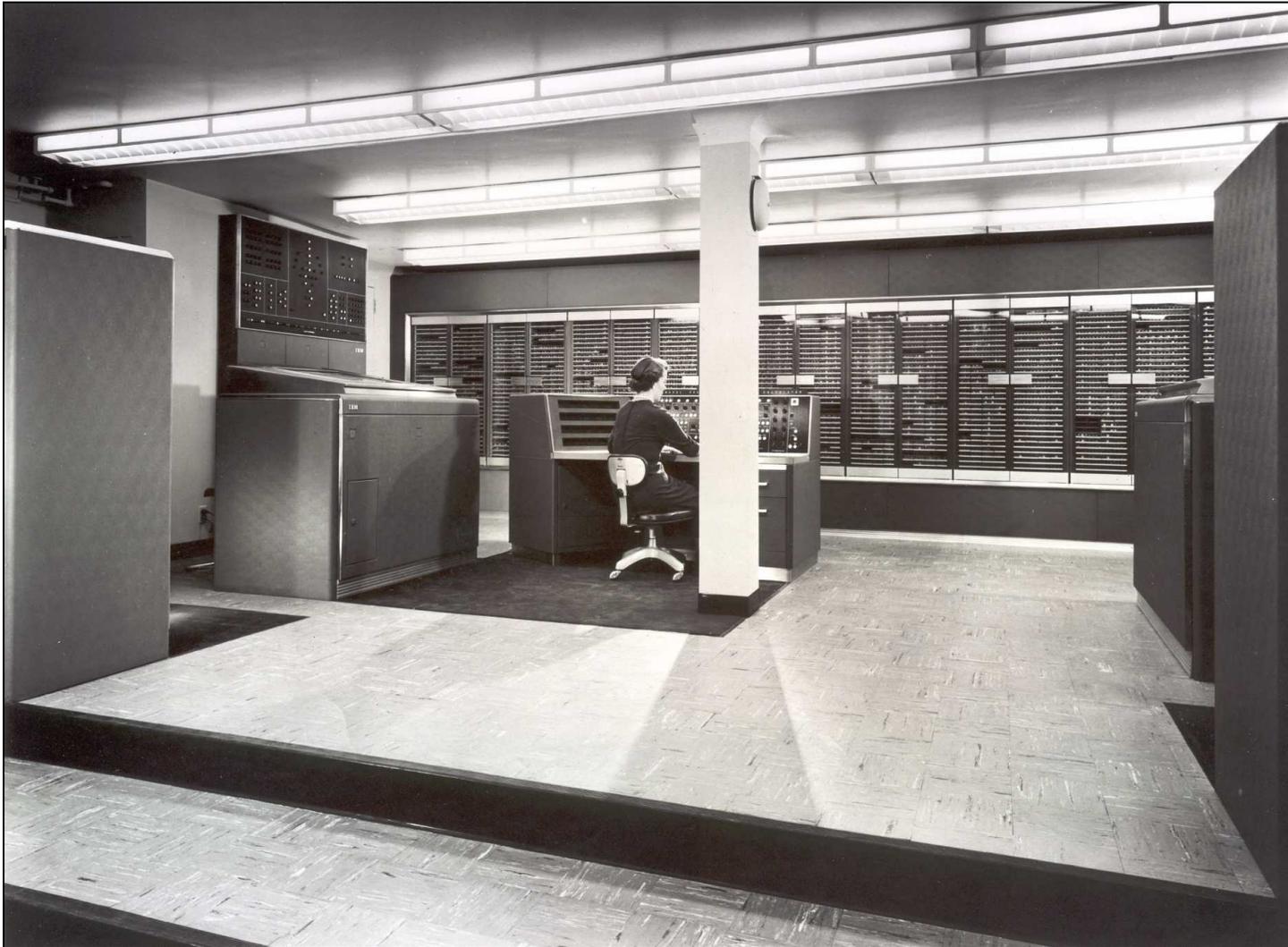
Unterstützung beim Laden der Programme, der Ein-/Ausgabe (Spooling), Umwandlung von Zahl- und Zeichendarstellungen.



Univac-1, gebaut von Reminton Rand, 1950 wurde die erste Maschine an das U.S. Census Bureau ausgeliefert



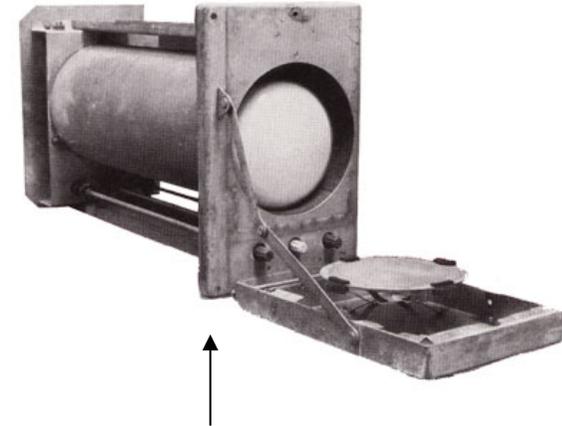
NORC (Naval Ordnance Research Calculator) Supercomputer



NORC (Naval Ordnance Research Calculator) Supercomputer

schnellster Rechner der Welt von 1954 - 1963

- Decimal integer and floating-point notation and operation.
- Word size: 16 decimal digits + check digit (64 + 2 bits).
- 64 three-address instructions.
- Clock: 1 μ sec.
- 15,000 operations per second with automatic error checking.
- Two universal registers, one million digits per second.
- Three address/index registers.
- Add time: 15 μ sec. Multiply: 31 μ sec. Divide: 227 μ sec.
- Random-access CRT memory: 3600 words, 8 μ sec access, provided by 264 [Williams-type CRTs](#)
- Magnetic tape: 8 units, 4-track, 510 char/inch, 71,500 char/sec.
- Printers: 2 units, 120 char/line, 150 lines/minute.
- Offline card/tape converter.
- Control console: Decimal display of register contents, manual controls, status lights.
- Swappable components ([pluggable units](#)).
- Cost: approximately \$2.5 million (1950s dollars).
- IBM profit: \$1.00 [\[40\]](#).



Konzepte und Meilensteine in den 60ern: Virtualisierungskonzepte für Prozessoren und Speicher

Steigende CPU-Leistung wird zunehmend durch niedrige E/A-Rate blockiert.

BS unterstützen die Ausführung mehrere unabhängiger Programme (Multiprogramming).

Begriff des virtuellen Speicher eingeführt (single-level-store, ATLAS , 1959).

Der Begriff des Prozesses für nebenläufige Aktivitäten wird eingeführt.

Interaktiver Betrieb durch mehrere Benutzer (Timesharing).

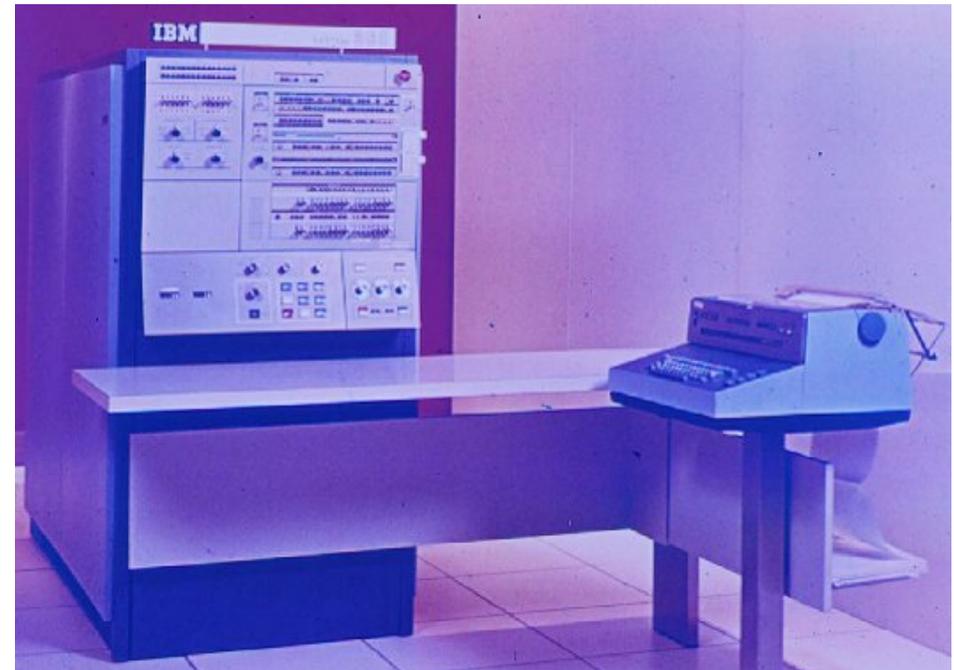
(Hardware-) Parallelität zur Durchsatzsteigerung durch Einsatz von E/A-Prozessoren.

- Spooling
- Direct Memory Access

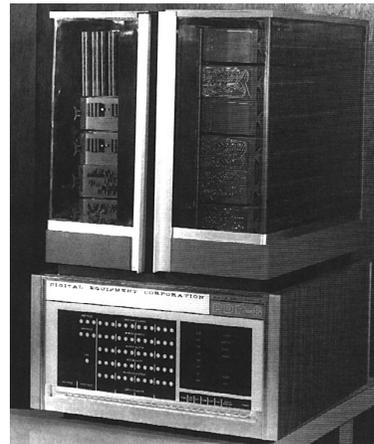


Entwicklung der Betriebssysteme: die "Mainframes"

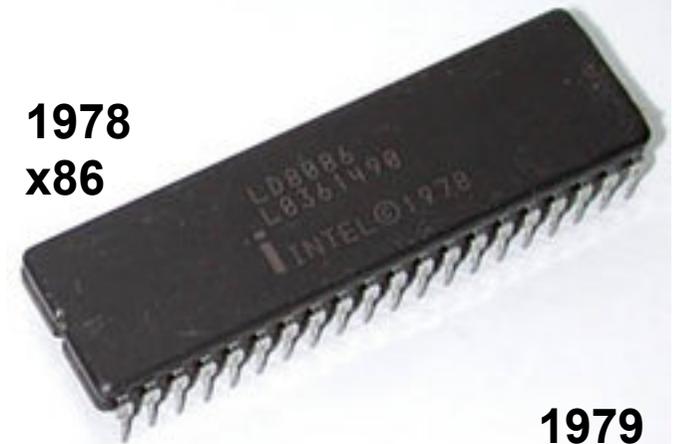
- 1965: IBM System/360
 - **Rechnerfamilie**
 - **Mehrprogrammbetrieb**
 - **Mikroprogrammierung**
 - Verwendung von integrierten Schaltkreisen (**ICs**)
 - 32-Bit Worte
 - 16 Mbyte Adreßraum
 - typischer Großrechner (auch als „*Mainframe*“ bezeichnet)



Entwicklung der Betriebssysteme: die Mini- und Microcomputer



1978
x86



1979
68K



- erster Mini: PDP-1 1961
- Meilensteine: PDP-8 1965 weitverbreiteter Prozeßrechner
- PDP-11 1970 16-bit Minicomputer
- C.mmp Multiprozessor aus 16 PDP-11 (CMU)
- Intel 8080 1972
- Intel 8086 1978 (16-Bit Prozessor)
- Motorola 68K 1979 (intern 32-Bit Prozessor)



Entwicklung der Betriebssysteme, die Mini- und Microcomputer

Neue Konzepte und Meilensteine in den 70ern: Software-Engineering !

- ➔ **"Leichtgewichtige" Betriebssysteme (UNIX 1972).**
- ➔ **Software-Engineering Ansätze.**
- ➔ **Strukturierter Systementwurf, nicht-funktionale Eigenschaften werden wichtig.**
- ➔ **Entwicklung in "höherer" Programmiersprache.**
- ➔ **Konzept der Schutzumgebung mit Zugriffsrechten, Capabilities (Hydra, StarOS: CMU 1974, 1978).**
- ➔ **Modulare Programmierung, Abstrakte Datentypen.**
- ➔ **Anwendung auf das BS.**



1970 ++ UNIX



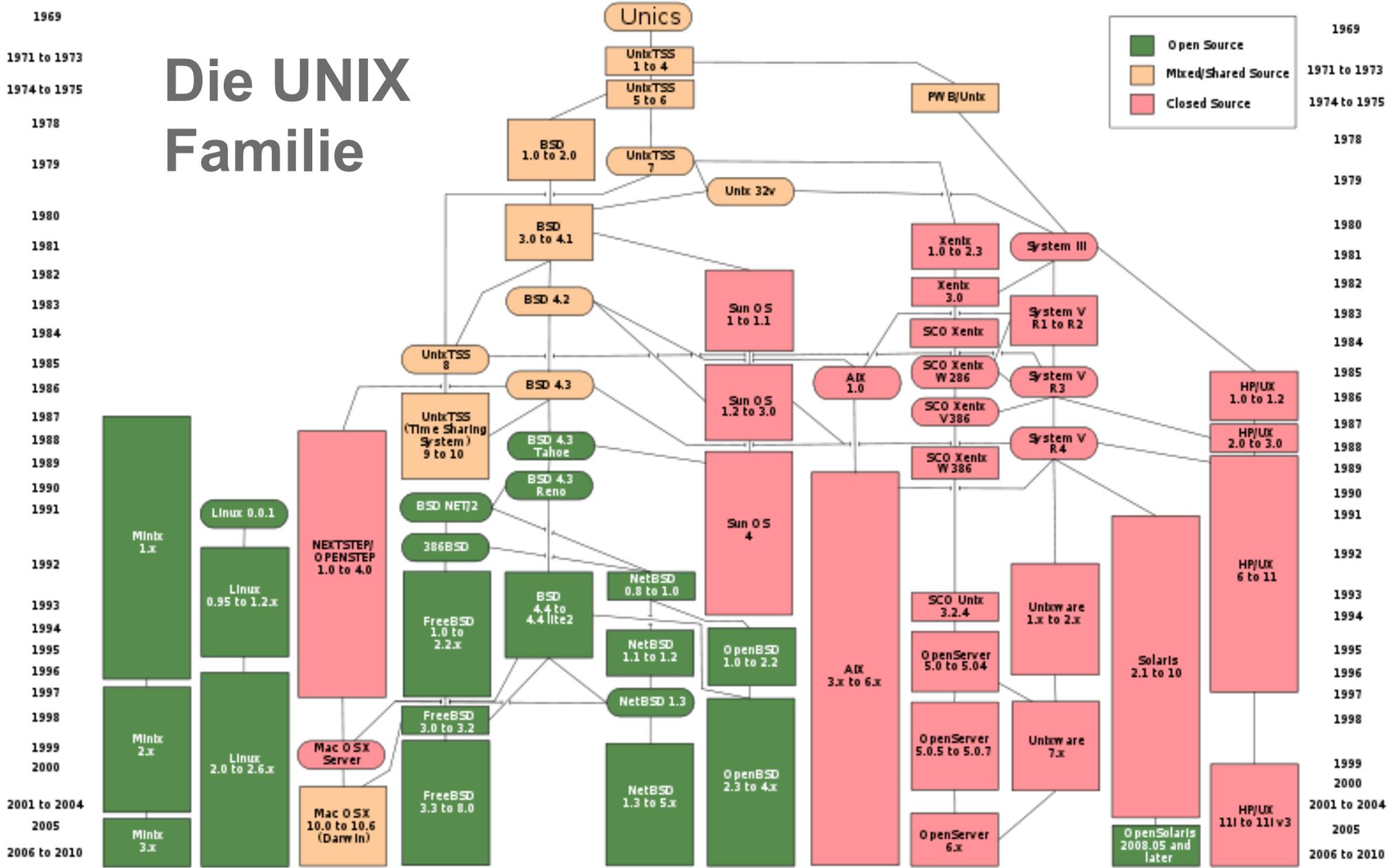
Ken Thomson, Dennis Ritchie

Thompson explains:

The UNIX kernel is an I/O multiplexer more than a complete operating system. This is as it should be.



Die UNIX Familie



Entwicklung der Betriebssysteme: Arbeitsplatzrechner, Personalcomputer und verteilte Systeme

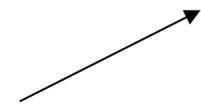
IBM PC (1981)



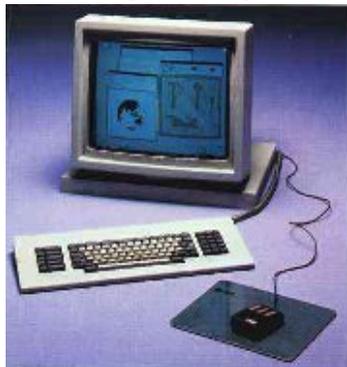
MAC (24.1.1984)



Lisa 1983-84

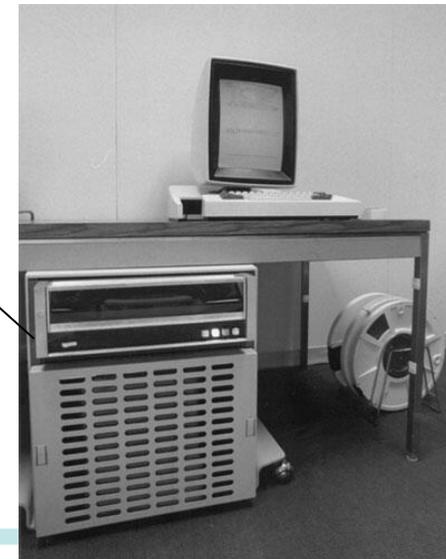
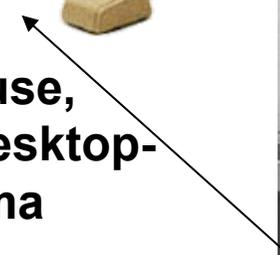


SUN (1984)



Unix, Ethernet, TCP/IP

**GUI, Mouse,
Icons, Desktop-
Paradigma**



**Xerox Alto
Mikroprogr.
LISP, Smalltalk
GUI, Simple
Icons, Mouse,...**

**Nachfolger:
Xerox Star,
Xerox Dorado**



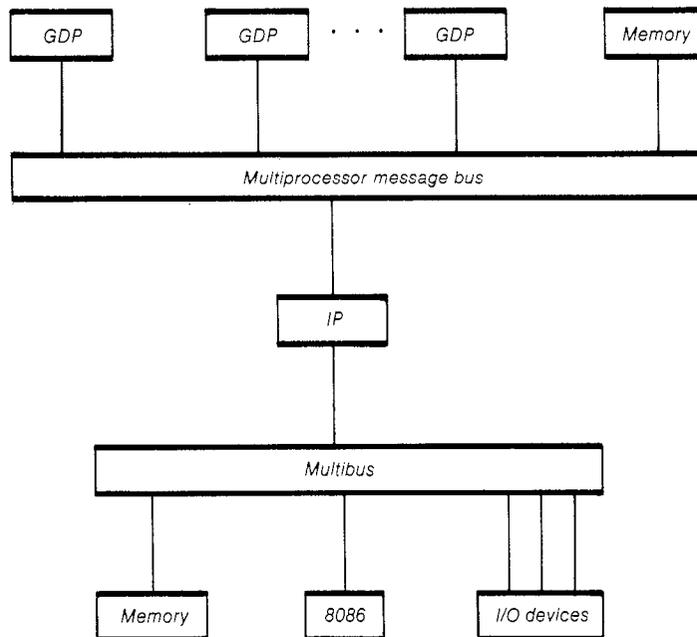
Entwicklung der Betriebssysteme: Workstations und verteilte Systeme

Konzepte und Meilensteine in den 80ern:

- ➔ "Operating System in Silicon" (Intel iAPX 432, 1983)
- ➔ Unix-Derivate werden Standard für Workstations
- ➔ Mikrokern Ansätze (QNX, Mach, Chorus, L3)
- ➔ Integration grafischer Benutzerschnittstellen (X: 1984, X11: 1987)
- ➔ Integration der Kommunikation in das Betriebssystem
- ➔ Homogene verteilte Systeme, Transparenzeigenschaften
- ➔ Leichtgewichtige Prozesse zur Strukturierung von Berechnungen
(Trennung von Prozeß und Adreßraum (--> Thread Konzept)
- ➔ Standardisierung (OSI, TCP/IP, NFS, POSIX, OSF, X/OPEN, OMG, ODP).



Intel iAPX 432



Erste Einführung 1981
Echte 32-Bit Architektur
250.000 Transistorfunktionen
GDP auf zwei Chips
„OS in Silicon“

Sehr komplexer Befehlssatz
Extensive Mikroprogrammierung

Master/Checker-Konfiguration

† komplex, teuer, langsam

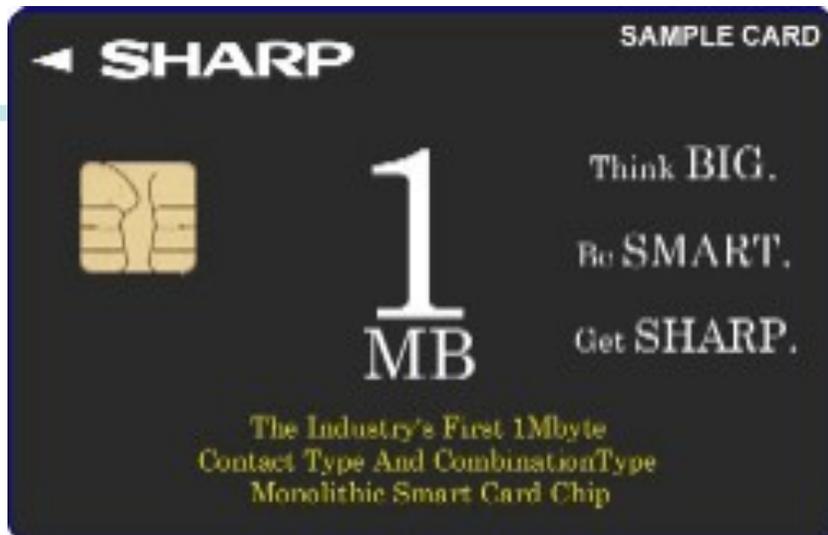


Entwicklung der Betriebssysteme: Internet, Middleware, eingebettete Systeme, Supercomputer

Neue Konzepte und Meilensteine in den 90ern bis heute:

- ➔ **WWW: Tim Berners-Lee (1989). "The explosion started in earnest during 1993, a year in which web traffic over the Internet increased by 300,000%."**
- ➔ **Middleware erlaubt Interoperabilität in heterogenen Systemen (DCE, CORBA).**
- ➔ **Computerisierung der Umwelt --> eingebettete Systeme.**
- ➔ **"Palm-Top Computer" und Handy Betriebssysteme.**
- ➔ **Multimedia erfordert Integration von Audio- und Videofunktionen.**
- ➔ **Supercomputer als Netze von Workstations.**

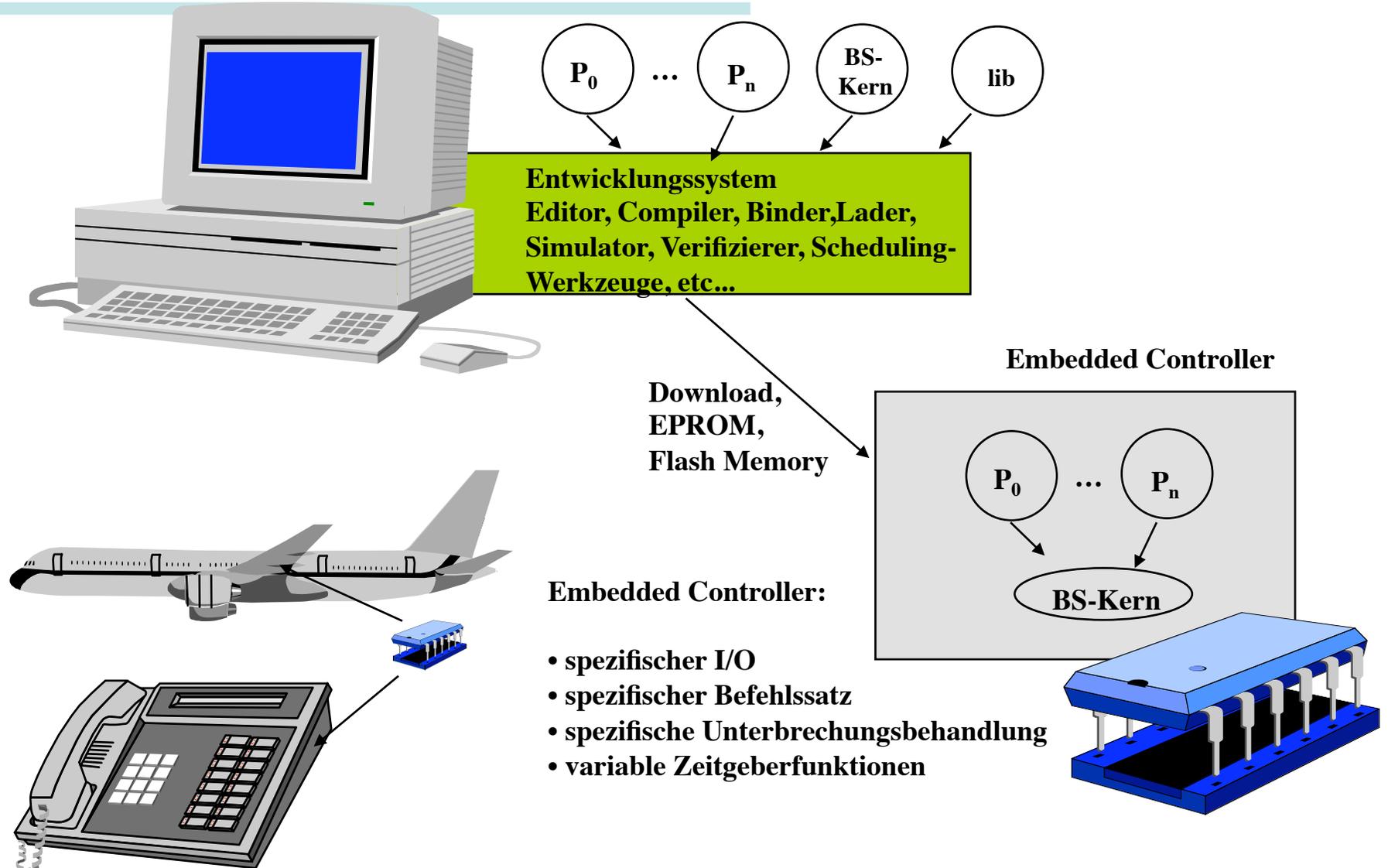




<p>2004 Smart Card (3g)</p>	<p>1981 erster IBM PC (~10kg)</p>
<p>16-32 Bit Prozessor@ 25Mhz 8 KB Hauptspeicher 1-2 MB Flash Kryptographie-Coprozessor Drahtlose Schnittstelle ~500Kb/sec</p>	<p>16 Bit Prozessor@ 5 Mhz 64 KB Hauptspeicher 160 KB Floppy - -</p>



Schema einer Cross-Entwicklungsumgebung: Konfigurierbarer BS-Kern

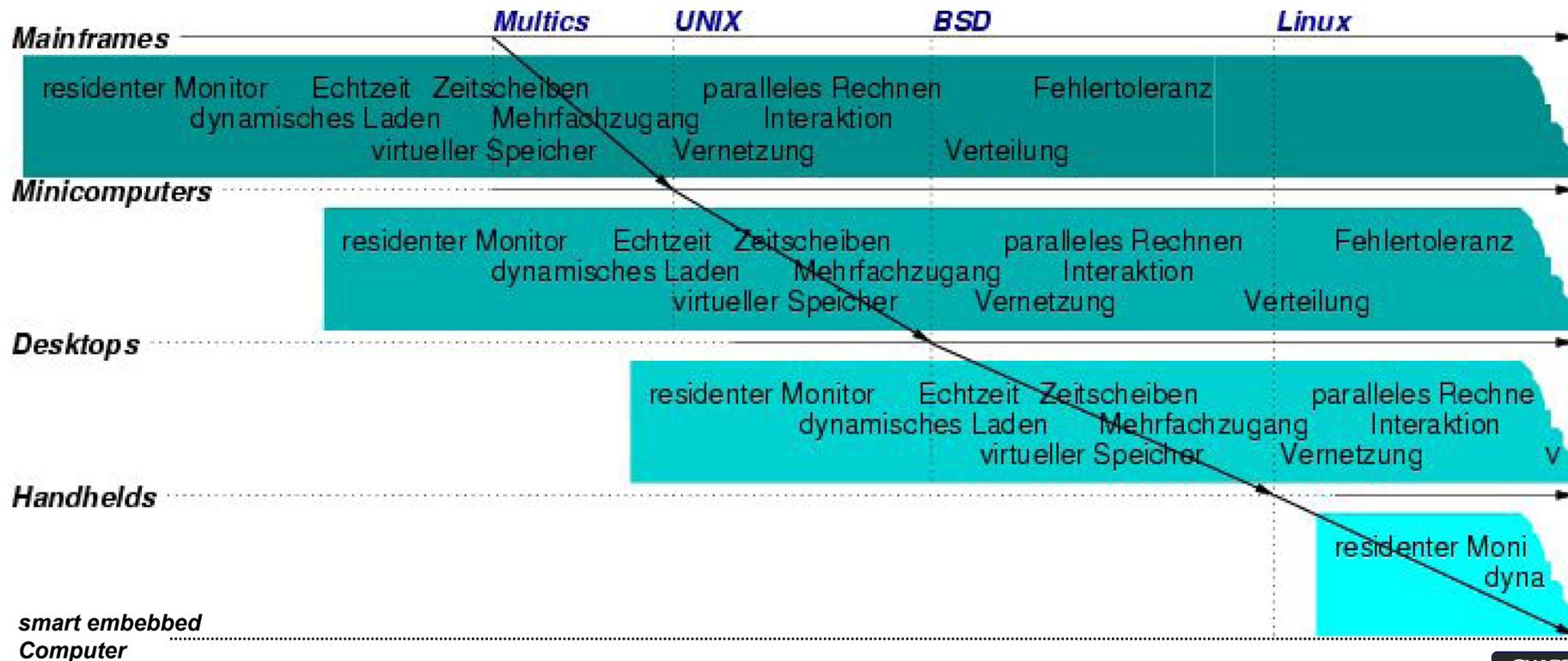


Embedded Controller:

- spezifischer I/O
- spezifischer Befehlssatz
- spezifische Unterbrechungsbehandlung
- variable Zeitgeberfunktionen



Ontogenese wiederholt Phylogenese



Technologie als Antrieb für die Entwicklung schafft die Voraussetzungen



Betriebssysteme für Mobile Anwendungen



- Apple iOS
- Android
- Tizen
- bada
- BlackBerry
- LiMo
- Maemo
- MeeGo
- Microsoft Windows Mobile
- Openmoko
- Palm webOS
- Qualcomm Brew
- Symbian OS

mehrere Netzwerke, Touchscreen, Kamera, Audio, Navigation,



Betriebsarten eines Computersystems

Klassifikation nach der Art der Auftragsbearbeitung

- | | |
|---|---|
| ➔ | Stapelverarbeitung (<i>Batch Processing</i>)
- eine Aufgabe nach der anderen |
| ➔ | Interaktiver Betrieb (<i>Interactive Processing</i>)
- Rechner reagiert sofort auf Befehle |
| ➔ | Time-Sharing-Betrieb
- Rechenzeit wird über mehrere Benutzer oder Programme aufgeteilt |
| ➔ | Echtzeitbetrieb (<i>Real Time Processing</i>)
- Rechner reagiert innerhalb vorgegebener Zeitschranken |

Klassifikation nach Ressourcennutzung:

- | | |
|---|---|
| ➔ | Ein- oder Mehrprogrammbetrieb (<i>Uni-Programming, Multiprogramming</i>) |
| ➔ | Ein- oder Mehrbenutzerbetrieb (<i>Single User, Multiuser</i>) |



Inhaltliche Ausrichtung BS I

Die Hardware-Software Schnittstelle

Assembler, Adressierung, Prozessorstatus, E/A, Unterbrechungen

Prozesse, Nebenläufigkeit und Scheduling

Prozesszustände, Threads, Prozesswechsel, Schedulingstrategien

Prozess- Interaktion und Synchronisation

Wechselseitiger Ausschluss, Schlossvariable, Semaphore, Monitore, Interaktionsmodelle

Hauptspeicherverwaltung

Adressumsetzung, Mehrprogrammbetrieb, Virtueller Speicher, Seitentauschstrategien

Dateiverwaltung

Dateistruktur, Verzeichnisse, effiziente Abbildung auf Plattenstruktur



Weitergehende Vorlesungen

Prinzipien und Komponenten eingebetteter Systeme

Kommunikation und Netze

Betriebssysteme II (Master)

Echtzeitsysteme (Master)

Verteilte Systeme (Master)

