Technische Universität Braunschweig

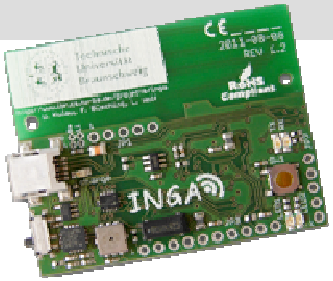
Institut für Betriebssysteme und Rechnerverbund

Vorkurs Informatik

Einige Themen aus „Informatik der Systeme“

Lars Wolf

Kurzvorstellung: zur Person & zum Institut

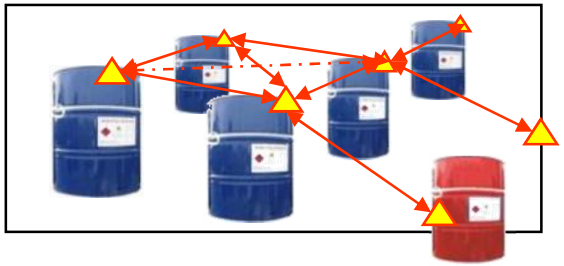


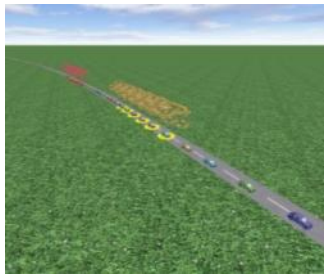
Prof. Dr. Lars Wolf

- Professor@TUBS seit 2002
- Leiter des IBR

IBR: 3 Abteilungen

- Algorithms (ALG)
  - Prof. Dr. Sándor Fekete
- Communication & Multimedia (CM)
  - Prof. Dr. Lars Wolf
- Distributed Systems (DS)
  - Prof. Dr. Rüdiger Kapitza





Microprocessor lab

Aufbau des Bachelorstudiengangs

Pflichtbereich Informatik: Grundlagen der Informatik (44 LP)

- Algorithmen u. Datenstrukturen, Programmieren I und II, Logik, Theoretische Informatik I und II, Technische Informatik I und II

Pflichtbereich Informatik: Informatik der Systeme (27 LP)

- Softwaretechnik/Softwareentwicklungspraktikum, Betriebssysteme, Computernetze, Relationale Datenbanksysteme I

Wahlpflichtbereich Informatik (20–25 LP)

Pflichtbereich Mathematik (25 LP)

- Lineare Algebra, Diskrete Mathematik, Analysis

Wahlpflichtbereich Mathematik (10 LP)

- 2 aus 4: Algebra, Stochastik, Numerik, Statistik

Seminar (5 LP), Teamprojekt (5 LP), Bachelorarbeit (15 LP)
Nebenfach (14–19 LP), Schlüsselqualifikationen (10 LP)

Aufbau des Bachelorstudiengangs : Beginn Wintersemester

Musterstudienplan Bachelorstudium Informatik (Beginn: Wintersemester)

Bereich	1. Semester (WS)	2. Semester (SS)	3. Semester (WS)	4. Semester (SS)	5. Semester (WS)	6. Semester (SS)	
Kompetenz-bereich Informatik [116-121 LP]	Orientierungstage Programmieren 1 – 6 LP Algorithmen und Datenstrukturen – 8 LP	Programmieren 2 – 6 LP Einführung in die Logik – 5 LP Elektrotechnische Grundlagen der Technischen Informatik – 4 LP "Schwerpunkte im Informatikstudium" – Ringvorlesung	Software Engineering – 12 LP Technische Informatik 1 + 2 – 8 LP Theoretische Informatik 1 – 5 LP Betriebssysteme – 5 LP Relationale Datenbanksysteme 1 – 5 LP Informatik: Wahlpflicht-Modul – 5 LP	Theoretische Informatik 2 – 6 LP Computernetze 1 – 5 LP Informatik: Wahlpflicht-Modul – 5 LP	Seminar – 5 LP Teamprojekt – 5 LP Informatik: Wahlpflicht-Modul – 5 LP	Bachelorarbeit – 15 LP Informatik: Wahlpflicht-Modul – 5 LP	
Kompetenz-bereich Mathematik [35 LP]	Lineare Algebra – 10 LP Diskrete Mathematik – 5 LP	Analysis – 10 LP Mathematik: Wahlpflicht-Modul – 5 LP			Mathematik: Wahlpflicht-Modul – 5 LP		
Nebenfach [14 - 19 LP]				Nebenfach-Modul – 5 LP	Nebenfach-Modul – 5 LP	Nebenfach-Modul – 5 LP	
Schlüssel-qualifikationen [10 LP]	Wissenschaftliches Arbeiten – 2 LP				Schlüsselqualifikationen – 4 LP	Schlüsselqualifikationen – 4 LP	
∅ 180 LP	31	30	30	31	29	29	

Pflichtmodule | Wahlpflichtmodule | Wahlmodule

Aufbau des Bachelorstudiengangs : Beginn Sommersemester

Musterstudienplan Bachelorstudium Informatik (Beginn: Sommersemester)

Bereich	1. Semester (SS)	2. Semester (WS)	3. Semester (SS)	4. Semester (WS)	5. Semester (SS)	6. Semester (WS)	
Kompetenz-bereich Informatik [116-121 LP]	Orientierungstage Einführung in die Logik 5 LP Computernetze 1 5 LP "Schwerpunkte im Informatikstudium" Ringvorlesung	Programmieren 1 6 LP Algorithmen und Datenstrukturen 8 LP Theoretische Informatik 1 5 LP	Programmieren 2 6 LP Theoretische Informatik 2 6 LP Informatik: Wahlpflicht-Modul 5 LP Informatik: Wahlpflicht-Modul 5 LP Informatik: Wahlpflicht-Modul 4 LP	Software Engineering 1 4 LP Technische Informatik 1 + 2 4 LP Betriebssysteme 5 LP Relationale Datenbanksysteme 1 5 LP	SEP* 8 LP Seminar 5 LP Informatik: Wahlpflicht-Modul 5 LP	Teamprojekt 5 LP Bachelorarbeit 15 LP Informatik: Wahlpflicht-Modul 5 LP	
Kompetenz-bereich Mathematik [35 LP]	Analysis 10 LP Mathematik: Wahlpflicht-Modul 5 LP	Lineare Algebra 10 LP Diskrete Mathematik 5 LP		Mathematik: Wahlpflicht-Modul 5 LP			
Nebenfach [14 - 19 LP]			Nebenfach-Modul 5 LP	Nebenfach-Modul 5 LP	Nebenfach-Modul 5 LP		
Schlüssel-qualifikationen [10 LP]	Wissensch. Arbeiten 2 LP Wiss. Textver. Mit Latex 3 LP					Schlüsselqualifikationen z.B. Gesch. d. Mathematik 5 LP	
∅ 180 LP	30	34	31	28	27	30	

Pflichtmodule | Wahlpflichtmodule | Wahlmodule | ***SEP** - Softwareentwicklungspraktikum

Verschiebungen im Lehrplan durch Beginn Sommer- statt Wintersemester

Insbesondere Computernetze verschoben

- Im 1. Semester statt im 4. Semester
- Bei Beginn im Wintersemester kommt Computernetze nach
  - Algorithmen und Datenstrukturen
  - Programmieren
  - Betriebssysteme
  - Technische Informatik
  - Software-Engineering
  - Theoretische Informatik
  - Relationale Datenbanksysteme
  - Mathematik, Einführung in die Logik
- Bei Beginn im Sommersemester dagegen vor all diesen

Computernetze vor anderen Lehrveranstaltungen ...

nicht so kritisch, denn

formal gab es auch bislang keine Voraussetzungen

Aber implizit Annahmen, dass Verständnis vorhanden ist für

- Aufbau von Rechnern
- Abläufe in Rechnern und deren „Grund-Software“
- Datenstrukturen, -verwaltung, -speicherung, ...

→hier kurze Übersicht zu einigen solchen Aspekten

Computernetze vor anderen Lehrveranstaltungen ...

Positiver Aspekt:

- Frühzeitig im Studium sehen, dass Informatik nicht nur Programmieren, Lösung von mathematisch motivierten Problemen, ... ist

- Schauen wir als Beispiel an, was die Gesellschaft für Informatik dazu sagt, was Informatik eigentlich ist

Informatik als Disziplin

(aus GI-Positionspapier "Was ist Informatik?" , Mai 2006)

Im Zentrum der Informatik steht die **Information**. Sie bezieht sich auf Fakten, Wissen, Können, Austausch, Überwachen und Bewirken; sie will erzeugt, dargestellt, abgelegt, aufgespürt, weitergegeben und verwendet werden; sie ist meist komplex und undurchschaubar mit anderen Informationen vernetzt.

In der Regel hat die Information sich selbst als Bearbeitungsobjekt: Um Information zu nutzen, werden konkrete Gegebenheiten und Vorgänge, aber auch abstrakte Bereiche – mit Hilfe von Information – in geeigneter Weise modelliert und simuliert.

...

Informatik als Disziplin (2)

(aus GI-Positionspapier "Was ist Informatik?" , Mai 2006)

Hierfür werden Werkzeuge konzipiert, entwickelt und eingesetzt. Es werden Sprachen und Systeme zur Realisierung der abstrakten Vorgehensweisen und Verarbeitungsvorschriften – mit Hilfe von Information – konstruiert, hergestellt und genutzt. Alle auf diese Art gewonnenen Erkenntnisse, Methoden und Ergebnisse werden überall dort, wo Information eine Rolle spielt, in ständig wachsendem Maße verwendet – und kontrolliert, wiederum mit Hilfe von Information. Diese *starke innere Vernetzung, der hohe Abstraktionsgrad, die digitale Darstellung, die Mischung aus Analyse und Synthese, aus Konstruktion und Integration beherrschen das Denken und Arbeiten in der Informatik*.

...

Informatik als Disziplin (3)

(aus GI-Positionspapier "Was ist Informatik?" , Mai 2006)

Die Wissenschaft Informatik befasst sich mit der **Darstellung, Speicherung, Übertragung und Verarbeitung von Information.** Dabei untersucht sie die unterschiedlichsten Aspekte: elementare Strukturen und Prozesse, Prinzipien und Architekturen von Systemen, Interaktionen in kleinen, mittleren und weltumspannenden Netzen, die Konzeption, Entwicklung und Implementierung von Hardware und Software bis hin zu hochkomplexen Anwendungssystemen und der Reflexion über ihren Einsatz und die Auswirkungen.

...

Informatik als Disziplin (4)

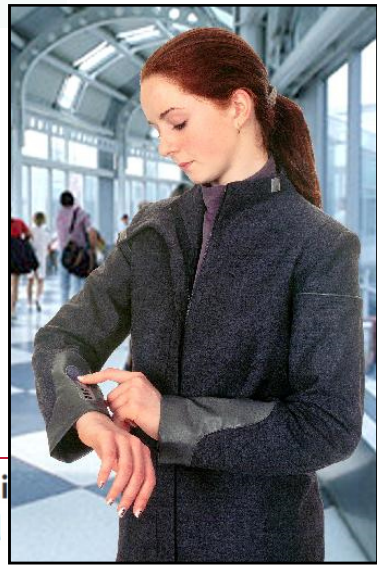
(aus GI-Positionspapier "Was ist Informatik?" , Mai 2006)

Die Informatik ist sowohl eine **Grundlagenwissenschaft** als auch eine **Ingenieurwissenschaft.** Darüber hinaus besitzt sie Aspekte einer **Experimentalwissenschaft.** Ihre Produkte sind zwar überwiegend abstrakt, haben aber sehr konkrete Auswirkungen. Ihre Denkweisen dringen in alle anderen Wissenschaften ein, führen zu neuen Modellen und Darstellungsweisen und lassen neuartige Hard- und Softwaresysteme entstehen.

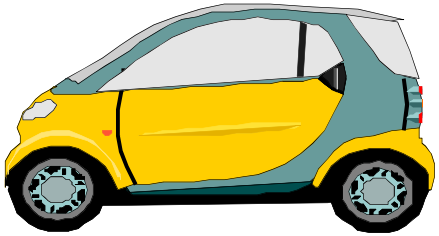
Beispiele für Vernetzte Informatik-Systeme

IP picture frame
http://www.ceiva.com/

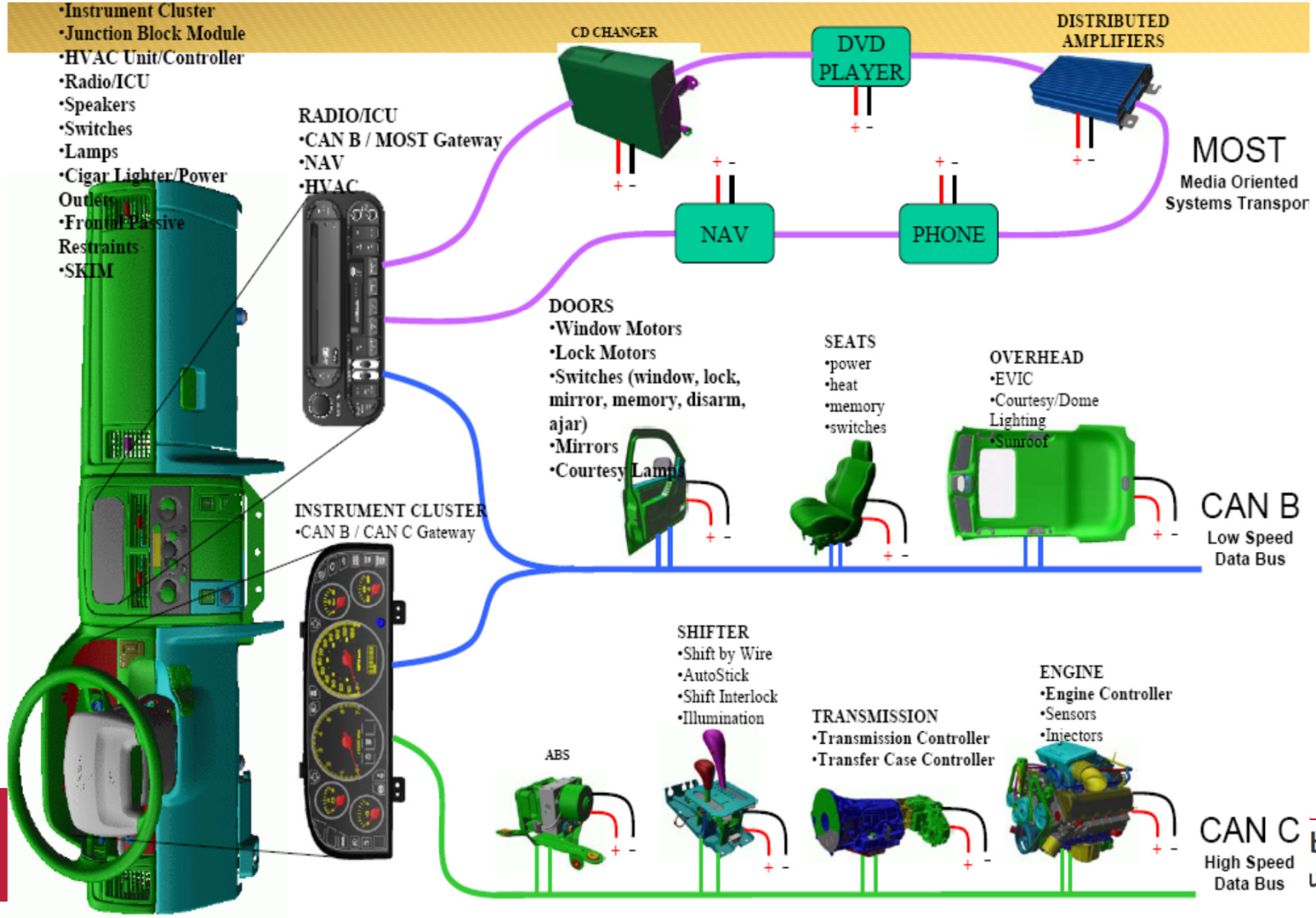
Web-enabled toaster +
weather forecaster



Noch ein Beispiel eines typischen Informatiksystems ...



Fahrzeug als Verteiltes System



bssysteme und

Einige Themen aus „Informatik der Systeme“

Insbesondere aus Betriebssysteme

- Was ist ein „Betriebssystems“
- Computer-Hardware – ein Überblick
- Architektur von Betriebssystemen

Begriffsbildung „Betriebssystem“

Silberschatz:

„An operating system is similar to a government… Like a government the operating system performs no useful function by itself.“

DIN 44300:

Die Programme eines digitalen Rechensystems, die zusammen mit den Eigenschaften dieser Rechenanlage die Basis der möglichen Betriebsarten des digitalen Rechensystems bilden und die insbesondere die Abwicklung von Programmen steuern und überwachen.

Beispiele heutiger Betriebssysteme

Generalisten

- UNIX (in verschiedenen Varianten, z.B. Linux, BSD, Solaris, HP-UX, AIX, ...)
- Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows 2003 Server, Vista, Windows 7, Windows 8,... (Microsoft)
- OS/2, MVS (IBM)
- MacOS (Apple)
- BS2000 (Siemens)

aber auch viele Betriebssysteme aus

- Forschungsprojekten
- für speziellere Zwecke (Echtzeit, Mobile Geräte, ...)

Betriebssysteme: Ziele und Randbedingungen

Sichten auf ein Betriebssystem

- Betriebssystem als erweiterte/virtuelle/abstrakte Maschine
- Betriebssystem als Ressourcenmanager

Ziele eines Betriebssystems:

- Anpassung der Benutzerwelt an die Maschinenwelt
- Organisation und Koordination des Betriebsablaufs
- Steuerung und Protokollierung des Betriebsablaufs

Randbedingungen:

- Die vorhandenen Betriebsmittel sind so einzusetzen, dass Engpässe bzw. Überlastungssituationen vermieden werden.
- Das Betriebssystem soll für die Durchführung seiner Aufgaben einen geringen Rechenaufwand benötigen.
- Ein Betriebssystem hat robust gegen fehlerhafte Benutzerprogramme zu sein.
- Daten und Programme sollten gegen unerlaubten Zugriff und gelegentliche Ausfälle der Datenträger abgesichert werden.

Betriebssystem-Dienste (I)

Dienste für Anwendungsprogramme:

- Laden, Starten, Verwalten und Beenden von Programmen
- Erzeugen von neuen Prozessen
- Abwicklung von Ein- und Ausgabeoperationen
- Bereitstellung eines logischen Dateisystems
- Kommunikation zwischen Prozessen (lokal oder über Kommunikationsnetze)
- Steuerung von Geräten (Terminals, Drucker, Plotter, ...)
- Abwicklung von Aufträgen im Hintergrund

Betriebssystem-Dienste (II)

Dienste für den Betreiber des Systems:

- Erfassung von Verbrauchsdaten (Accounting)
- Sicherheitsüberwachung (Zugangskontrolle, Zugriffsschutz, Verschlüsselung)
- Operationen zur Administration (z.B. Abbruch von Prozessen)
- Testfunktionen (z.B. Entdeckung fehlerhafter Spuren auf Festplatten)
- Funktionen zur Initialisierung des Systems

Betriebssystem als abstrakte Maschine

Ein Betriebssystem stellt eine virtuelle Ablaufumgebung (abstrakte Maschine) bereit, die durch geeignete Abstraktionen die Anwendungsprogramme von der Komplexität der vorhandenen Hardware abschirmt.

Beispiele:

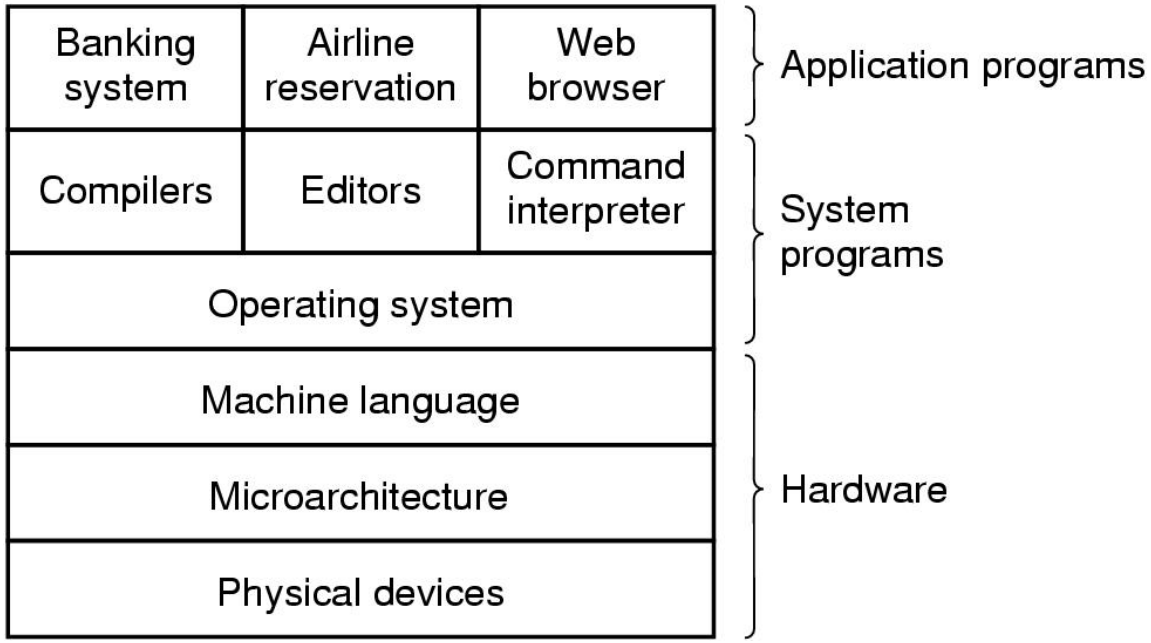
- Ein Betriebssystem bietet Funktionen zur Manipulation eines logischen Dateisystems, die die Applikation von den Details der Hardware (Adressen, Zylinder, Sektoren, Datenblockgrößen) abschirmen.
- Ein Betriebssystem stellt Applikationen einen linearen Adressraum zur Verfügung und verbirgt die interne Organisation des physikalisch vorhandenen Hauptspeichers.
- Ein Betriebssystem stellt Applikationen verlässliche Kommunikationsendpunkte zum Datenaustausch bereit, ohne die erforderlichen Sicherungsmechanismen sichtbar zu machen.

Schichten eines Rechnersystems

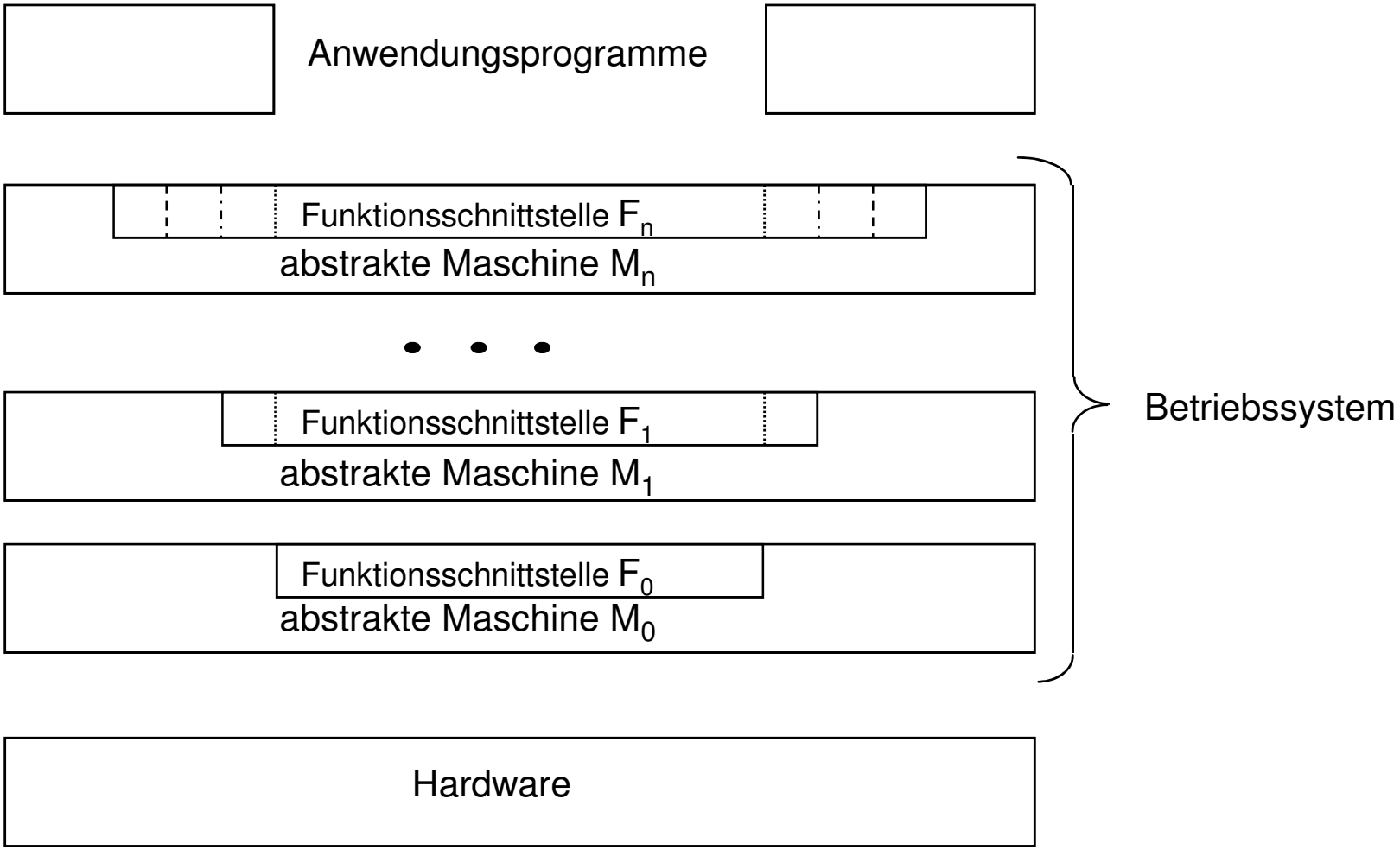
Schichtung dient der Beherrschung der Komplexität (durch Abstraktion) und macht Systeme flexibler und anpassungsfähiger

Schichten werden oft auch als abstrakte oder virtuelle Maschinen bezeichnet

Schichten erzeugen oftmals Leistungsverluste (overhead)



Hierarchie abstrakter Maschinen



Andere Sichtweise:
Betriebssystem als Ressourcenmanager

Programme benötigen Ressourcen zur Ausführung

- Zeit
- Platz
- ...

Betriebssystem koordiniert Zugriff, teilt Ressourcen zu, überwacht Nutzung, ...

- Prozessverwaltung / Prozessorzuteilung (Scheduling)
- Speicherverwaltung
- ...

Betriebssystem-Struktur

Ein Betriebssystem ist ein großes und komplexes System

- modularisiert in Komponenten und Subsysteme
- typischerweise überwiegend in C geschrieben (kleine Teile des Kerns und einige Gerätetreiber oftmals in Assembler)

Der Kern („kernel“) ist ein Programm, das die ganze Zeit parallel zu den Anwendungs- und anderen Systemprozessen läuft

Systemprogramme (z.B. Linker, Kommandointerpreter) werden bei Bedarf geladen

Betriebssystem-Struktur (II)

Dämone („daemons“) sind Hilfsprozesse, die ständig existieren, aber meist passiv sind.

- Dämonen erfüllen damit eine Server-Funktion
- Dämonen werden durch Ereignisse „aufgeweckt“ („Datei zum Drucken eingetroffen“)
- oder schauen selbst von Zeit zu Zeit nach, ob Arbeit da ist

Formen von Betriebssystemen

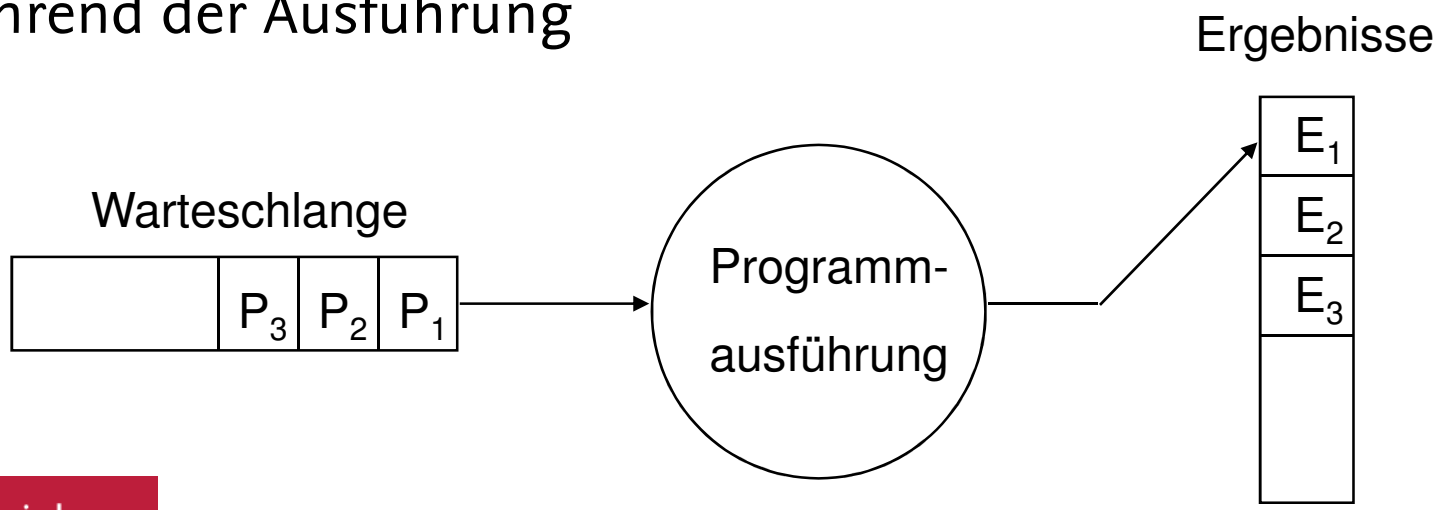
- Stapelverarbeitung (batch processing):
- Dialogverarbeitung (time sharing)
- Echtzeitverarbeitung (real-time processing)
- Verteilte Verarbeitung (distributed processing)
- Eingebettete Betriebssysteme
- Smartphone Betriebssysteme
- Smart-Card Betriebssysteme
- ...

Figure 1. IBM System/360 Model 40 Data Processing System

Formen von Betriebssystemen (I)

Stapelverarbeitung (batch processing):

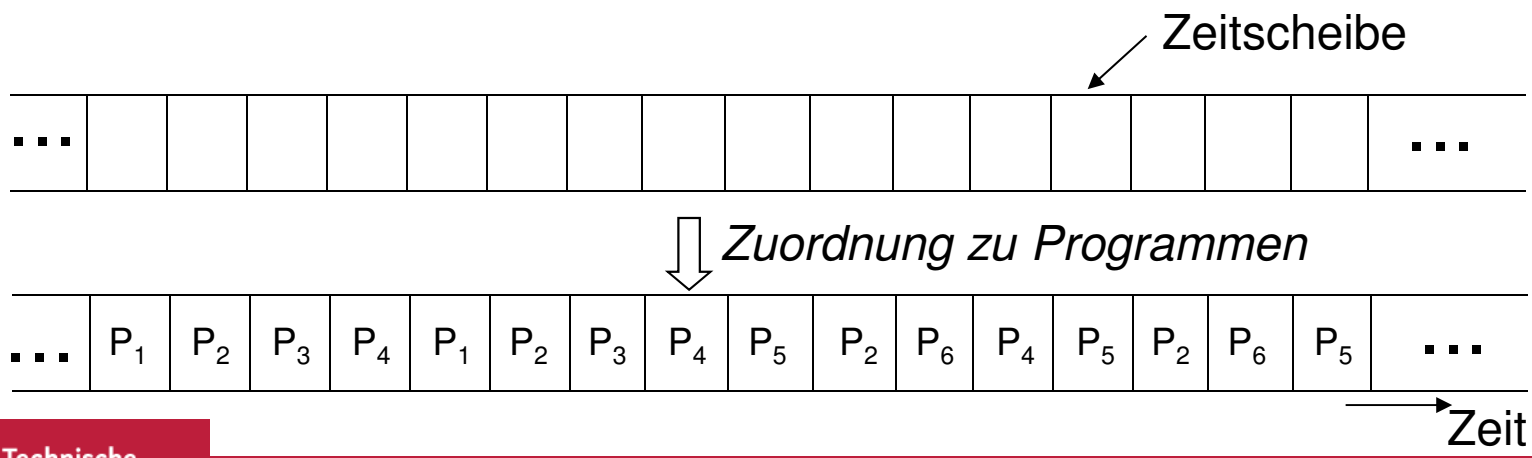
- Alle Aufträge werden in eine Warteschlange eingereiht.
- Aufträge werden ausgeführt, sobald sie den Kopf der Warteschlange erreichen und ein Prozessor verfügbar ist.
- Ergebnisse werden in Dateien protokolliert oder ausgedruckt.
- Keine Interaktion zwischen dem Benutzer und dem Programm während der Ausführung



Formen von Betriebssystemen (II)

Dialogverarbeitung (time sharing):

- Mehrere Programme werden quasi-parallel ausgeführt (multi-programming, multi-tasking).
- Mehrere Benutzer können Programme gleichzeitig starten und benutzen (multi-user).
- Benutzer können ihre Programme während der Ausführung interaktiv beeinflussen.
- Ausführungszeit wird in „kleinen Portionen“, sog. Zeitscheiben, zugeteilt.



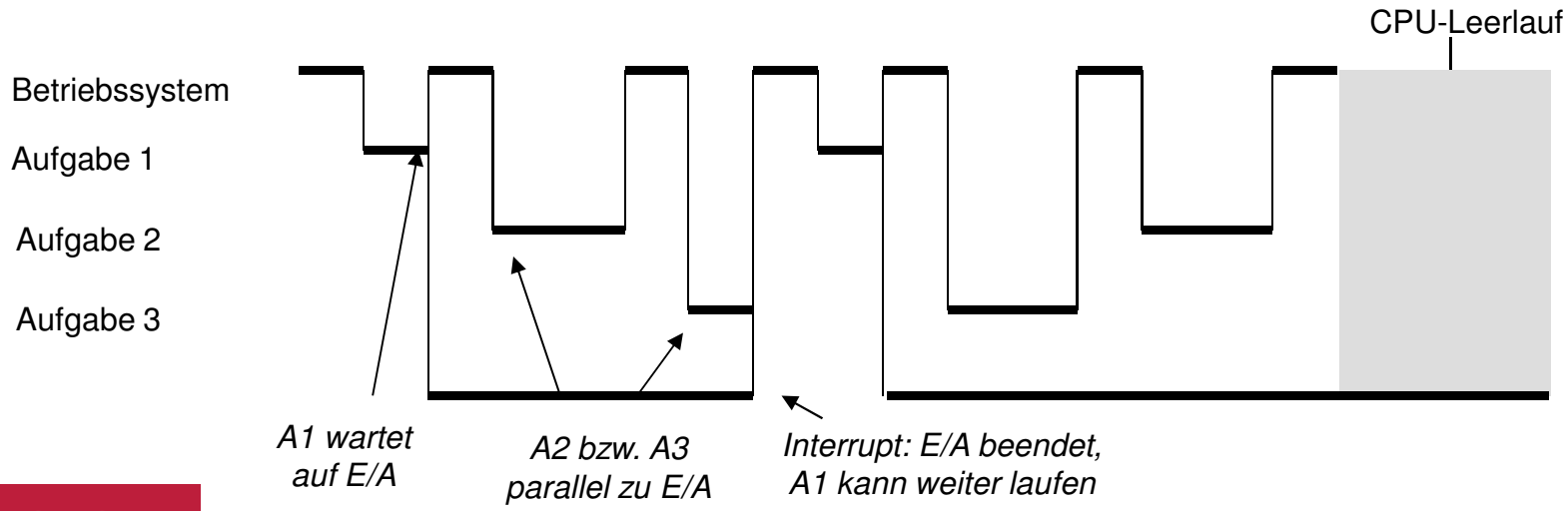
Beispiel: Time-Sharing

Ablauf normalerweise nicht reproduzierbar:

- Interrupts hängen von Umgebungsbedingungen ab
- Prozesse sollten nicht unter Zeitannahmen realisiert sein.

E/A-Geräte können parallel zur CPU arbeiten:

- Optimierungsziel: möglichst alle E/A-Geräte und CPU ständig beschäftigen
- typische Strategie: E/A-intensive Aufgaben bekommen höhere Priorität

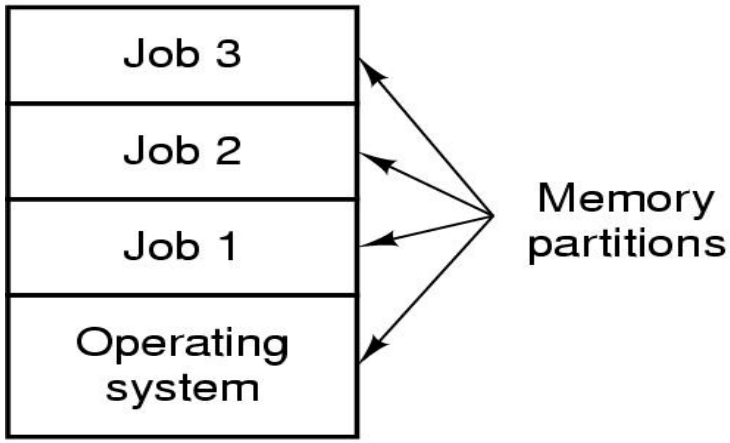


Multiprogramming u. Time-Sharing

Multiprogramming: Mehrere Programme gleichzeitig im Speicher bzw. in Bearbeitung

Spooling: Parallel zur Verarbeitung Einlesen neuer Jobs

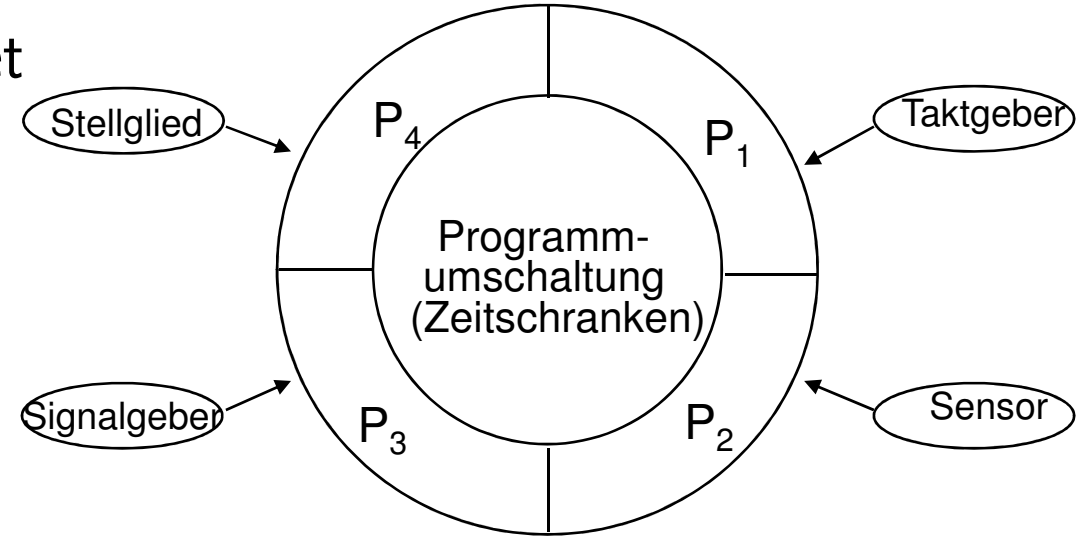
Time-Sharing: Variante von Multi-Programming mit Online-Terminals; kurze interaktive Jobs teilen sich die CPU mit lang laufenden Batch-Jobs



Formen von Betriebssystemen (III)

Echtzeitverarbeitung (real-time processing):

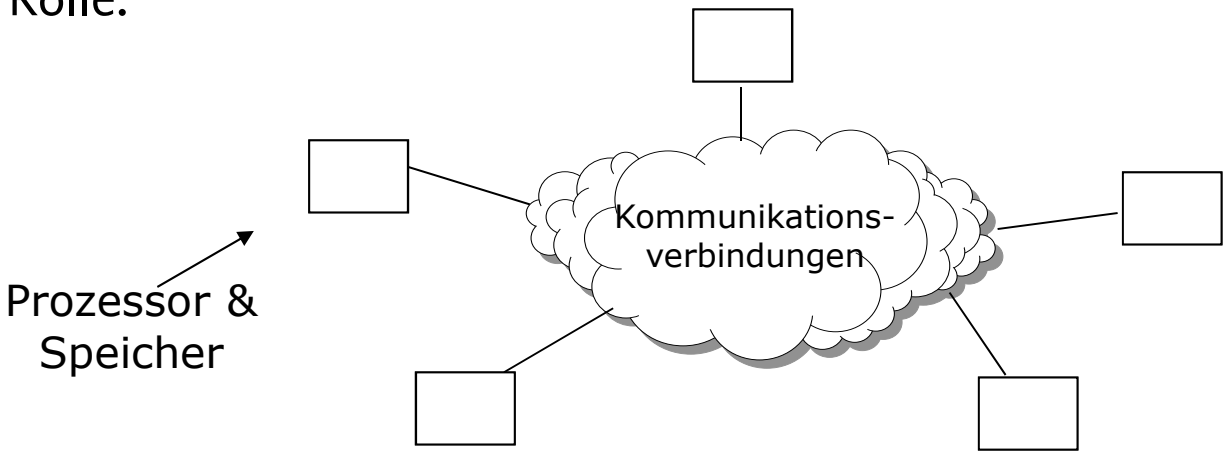
- Steuerung von Maschinen in zeitsensitiven Anwendungen (Robotersteuerung, medizinische Geräte, Kommunikation in Verkehrsmitteln)
- Reaktionszeiten unterliegen definierten, festen Zeitschranken.
- Das Betriebssystem schaltet bspw. Zeitschranken-gesteuert zwischen Programmen um.



Formen von Betriebssystemen (IV)

Verteilte Verarbeitung (distributed processing):

- Verarbeitung wird auf mehreren Prozessoren verteilt.
- In eng gekoppelten Systemen (Multiprozessorsysteme) benutzen die Prozessoren einen gemeinsamen Speicher und Takt.
- In lose gekoppelten Systemen teilen die Prozessoren weder Speicher noch Takt. Die Prozessoren kommunizieren über explizite Kommunikationsverbindungen.
- Hier spielen beide Komponenten, Betriebssysteme und Netze, eine ausschlaggebende Rolle.



Formen von Betriebssystemen (V)

Eingebettete Betriebssysteme

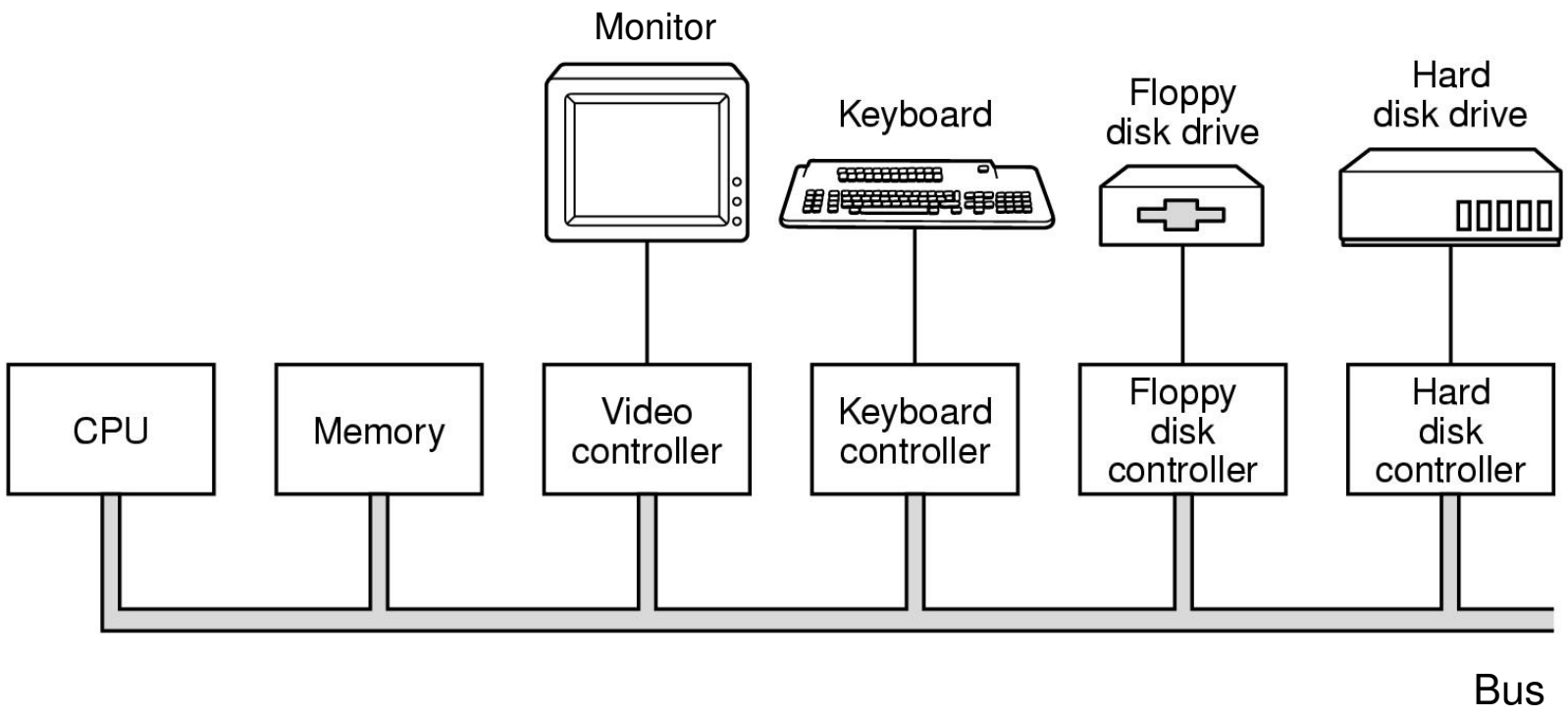
- Auf mobilen Geräten, aber auch in Haushaltsgeräten oder in kleinen Sensorknoten

- Beispiele: , Contiki, Windows CE, Windows Mobile, iOS, Android, NewtonOS

Smart-Card Betriebssysteme

- kreditgartengroße Geräte mit Chip
- Teilweise nur eine Funktion wie elektronisches Bezahlen, teilweise multifunktional
- Zum Teil Java-orientiert, so dass die Karten eine kleine virtuelle Java-Maschine enthalten

Computer-Hardware: Überblick



Komponenten eines einfachen, alten PCs (vereinfacht)

Computer-Hardware: Prozessoren

‚Gehirn' des Computers:

- holt Instruktionen aus dem Speicher und führt sie aus

Spezifische Befehle für jeden Prozessor

- bspw. Pentium, PowerPC

Interne Register mit schnellem Zugriff

- Daten aus langsamen Hauptspeicher zunächst in Register laden

Wichtige Spezial-Register der CPU

- Programmzähler, Stack-Pointer, ...

Ablage und Adressierung von Daten

- Reihenfolge von bits&bytes, Adressraum, Wortlänge, ...

CPU-Modi: User und Kernel Mode

Trennung von Betriebsystem-(Kern-)Prozessen und Anwendungsprozessen

- Schutz des Systems vor den Folgen von Software-Fehlern

Wechsel vom User Mode in den Kernel Mode

- Ereignisse: Interrupts oder Traps
- CHM (CHange Mode) Operation (nur privilegierte Prozesse)

User Mode

Benutzermodus (user mode)

- Der Prozessor bearbeitet Instruktionen von Anwendungsprogrammen.
- Der Befehlssatz des Prozessors und die verfügbaren Register sind auf eine Teilmenge beschränkt (unprivileged instruction set).
- Zugriff auf die Hardware ist entsprechend den Möglichkeiten der verwendeten Hardware beschränkt (hardware protection).
- Direkte Zugriffe auf andere in Ausführung befindliche Programme sind verboten.

Kernel Mode

Betriebssystemmodus (system/kernel mode)

- Der Prozessor arbeitet Instruktionen des Betriebssystems ab.
- Alle Befehle und Register des Prozessors sind verfügbar (privileged instruction set).
- Direkter, uneingeschränkter Zugriff auf die gesamte Hardware.
- Manipulation interner Daten laufender Programme möglich.
- Fehler gefährden die Systemstabilität.

Einstiegspunkte in ein Betriebssystem: Traps

Systemaufrufe (system calls, supervisor calls, software traps):

- Synchron zur laufenden Applikation (z.B. count = read(file,buffer,size)).
- Parameter werden über Register, den Stack oder einen Parameterblock übergeben.

Hardware Traps:

- Synchron zur laufenden Applikation (z.B. Division durch 0).
- Fehler wird vom Betriebssystem an die verursachende Applikation weitergeleitet.

Einstiegspunkte in ein Betriebssystem: Interrupts

Hardware Unterbrechungen (hardware interrupts):

- Asynchron zur laufenden Applikation (z.B. Empfang einer Bitfolge an einem Gerät).

Software Unterbrechung (software interrupts):

- Asynchron zur laufenden Applikation (oftmals intern erzeugt, um aufwendige Berechnungen auf Unterbrechungen mit geringerer Priorität zu verlagern).

Achtung: Asynchrone Unterbrechungen können Systemaufrufe oder andere Betriebssystemvorgänge unterbrechen!

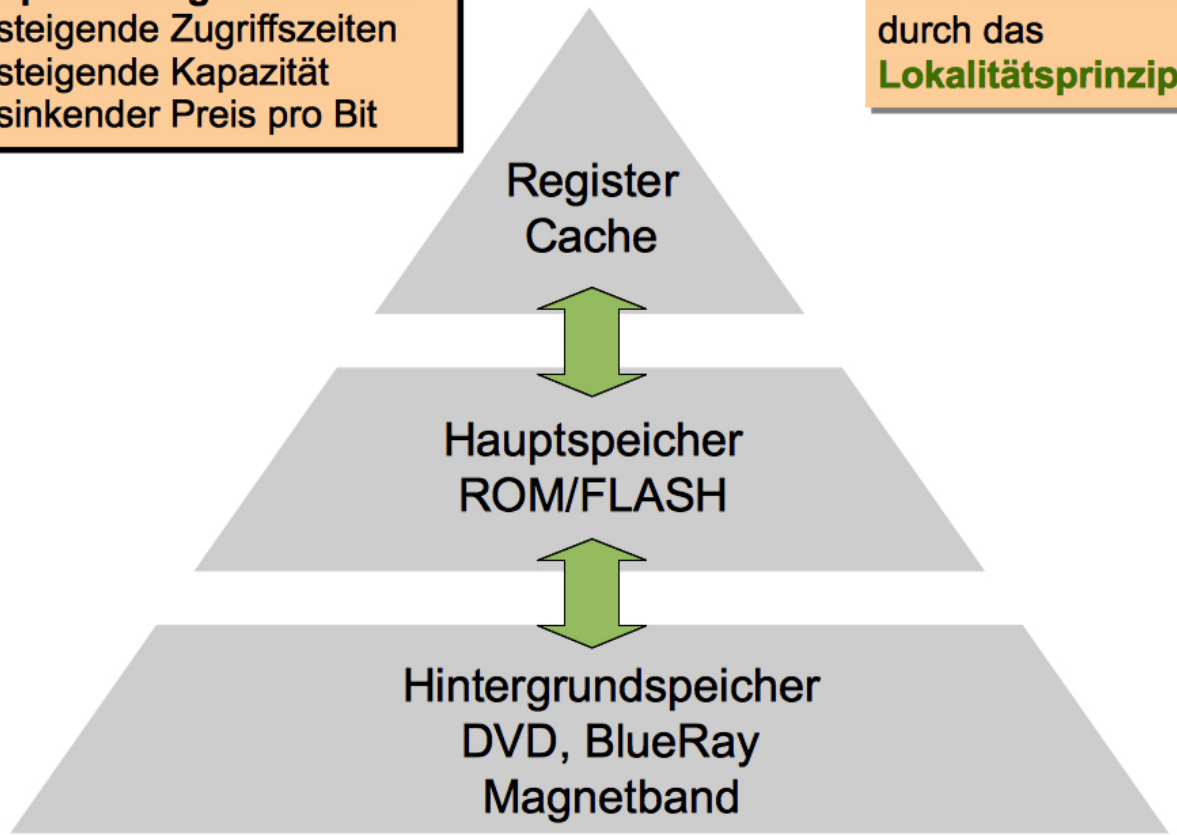
Computer-Hardware: Speicher

Idealerweise wäre Speicher unendlich groß, sehr schnell und billig – das gibt es aber nicht ☹

Speicher wird deshalb hierarchisch organisiert – wenig schneller und teurer Speicher, sehr viel langsamer und billiger Speicher

Speichereigenschaften
- steigende Zugriffszeiten
- steigende Kapazität
- sinkender Preis pro Bit

Sehr wirtschaftlich durch das **Lokalitätsprinzip**!



Ein- und Ausgabegeräte

Geräte bestehen aus der Hardware und einem Controller, der das Gerät steuert

Zur Einbindung eines Geräts in ein Betriebssystem und somit zur Ansteuerung und Nutzung benötigt man einen Gerätetreiber, der den Controller anspricht.

Dieser wird vom Gerätehersteller bereit gestellt und muss ins Betriebssystem eingebunden werden, z.B. durch

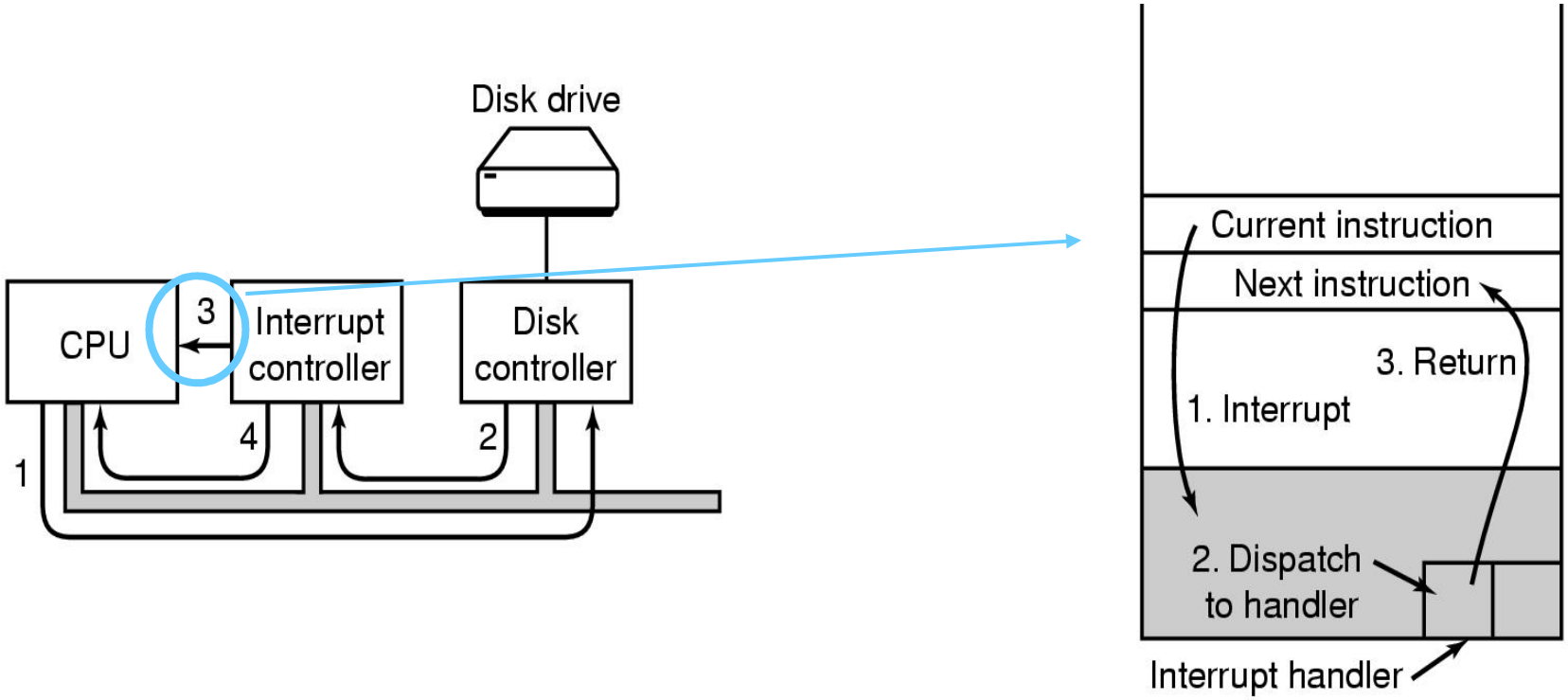
- Neu-Linken des Kernels, Rebooten des Systems (Unix),
- Eintragen des neuen Treibers in eine Konfigurationsdatei und Rebooten (Windows); oder
- Manuelles Laden von Kernel-Modulen
- Installation „On-the-Fly“ (z.B. USB, IEEE 1394)

Ablauf der Ein- und Ausgabe

Die eigentliche Ein- und Ausgabe kann durch drei verschiedene Methoden vorgenommen werden:

- **Busy Waiting:**
  Treiber startet I/O, fragt dann ständig das Device ab, ob es fertig ist. Wenn ja, schreibt er die Daten an die richtige Stelle
- **Interrupt-gesteuert:**
  Treiber stößt Gerät an. Dieses arbeitet dann unabhängig. Wenn es fertig ist, benachrichtigt es den Interrupt-Controller, der die CPU informiert.
- **Direct Memory Access (DMA):**
  ein spezieller Chip übernimmt die Steuerung der Ein- und Ausgabe ohne ständige Beteiligung der CPU. Vorteil: wesentlich schnellere Arbeit, aber natürlich auch teurer

Ein- und Ausgabegeräte: Interrupts



(a) Steps in starting an I/O device and getting interrupt

(b) How the CPU is interrupted

Noch einmal: Betriebssystem als Ressourcenmanager

Insbesondere bei Mehrprogrammbetrieb:

Programme benötigen Ressourcen zur Ausführung

- Zeit
- Platz
- ...

Betriebssystem koordiniert Zugriff, teilt Ressourcen zu, überwacht Nutzung, ...

- Prozessverwaltung / Prozessorzuteilung (Scheduling)
- Speicherverwaltung
- ...

Prozesse

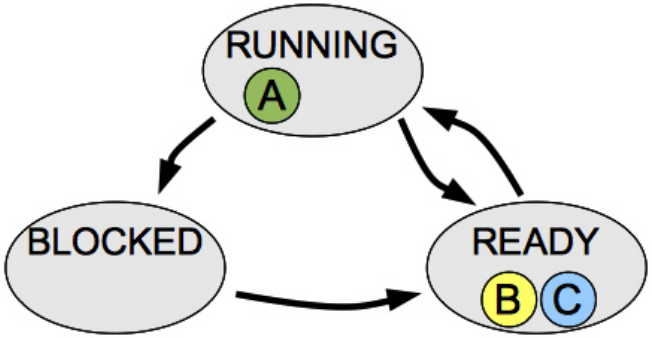
Ein Prozess … „ist ein Programm in Ausführung.“

- unbekannte Referenz, „Mundart“

Dazu gehört ein **Prozesskontext**, u.a.

- Genutzte Speicherbereiche
- CPU-Registerinhalte
- Prozesszustand
- Benutzerkennung
- Zugriffsrechte
- Aktuell belegte Betriebsmittel
- …

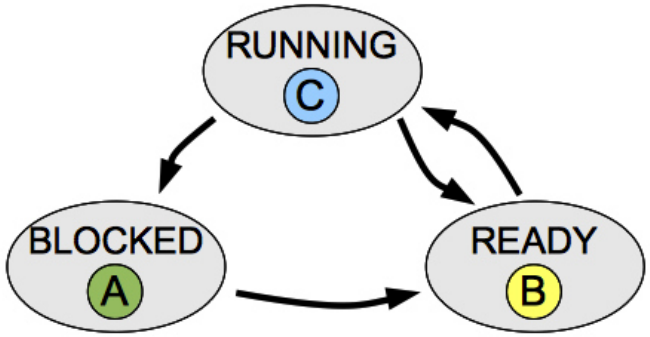
Prozessverhalten und -zustände



Prozesszustände

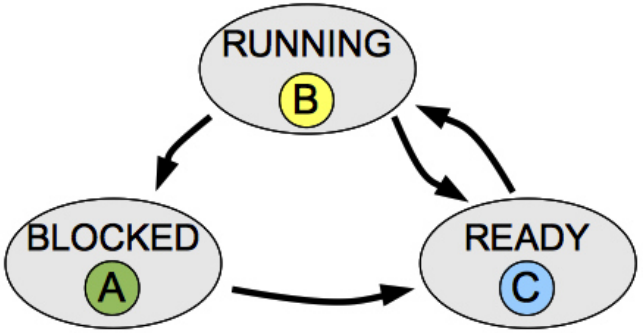
- **RUNNING**
  - Prozess wird gerade ausgeführt
- **READY**
  - Prozess ist rechenbereit, wartet auf die CPU
- **BLOCKED**
  - Prozess wartet auf die Beendigung einer E/A-Aktivität

Prozessverhalten und -zustände

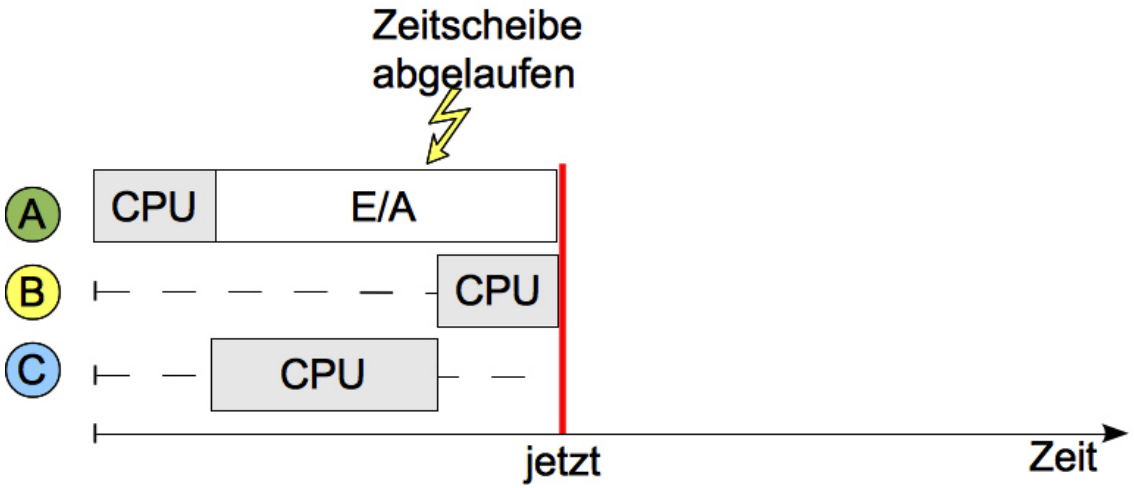


Prozess A hat einen E/A-Vorgang gestartet und ist in den Zustand BLOCKED übergegangen. Da A die CPU nun nicht benötigt, hat das Betriebssystem den Prozess C ausgewählt und von READY in RUNNING überführt. Es fand ein **Kontextwechsel** von A zu C statt.

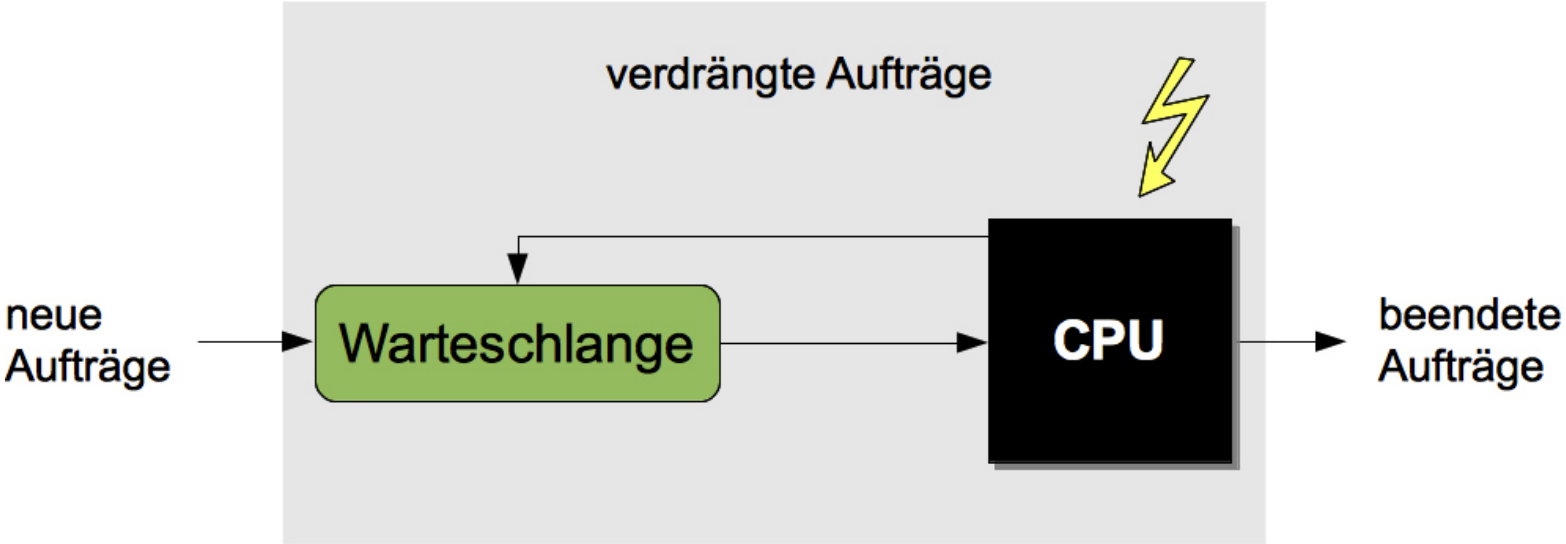
Prozessverhalten und -zustände



C hat die CPU zu lange „besessen“, wurde „**verdrängt**“ und ist daher nun wieder im Zustand READY. Damit kann jetzt endlich auch B bearbeitet werden und wird RUNNING.



CPU-Zuteilung (Scheduling)



Ein einzelner ***Scheduling*-Algorithmus** charakterisiert sich durch die Reihenfolge von Prozessen in der Warteschlange und die Bedingungen, unter denen die Prozesse der Warteschlange zugeführt werden.

Man könnte noch auf viele weitere Dinge eingehen …

Fragen, Anmerkungen, …?

Kontakt: wolf@ibr.cs.tu-bs.de