

# Betriebssysteme II

FH München  
WS 2006/07  
Hans-Georg Eßer  
[hans-georg.esser@fhm.edu](mailto:hans-georg.esser@fhm.edu)

Die Folien zu „Betriebssysteme II“ basieren auf den Ausarbeitungen von Prof. Christian Vogt, <http://www.cs.fhm.edu/~vogt> und Prof. Claudius Schnörr, <http://www.cs.fhm.edu/~schoerr>

## Zur Vorlesung (2)

### Voraussetzungen:

- **C** – Grundlagen der Programmierung in C
- **Rechnerarchitekturen** – grober Aufbau eines Computers (Prozessor, Hauptspeicher, Peripherie etc.)
- **Unix-Shell** – Benutzung der Standard-Shell *bash* unter Linux → Bash-Crashkurs
- **Python** – Programme lesen können (für Praktikum: auch programmieren)

## Zur Vorlesung (1)

### Termine

Betriebssysteme I: Mo / Mi 8:15-9:45 Uhr  
Betriebssysteme II: Die 8:15-9:45 Uhr  
5-Minuten-Pause jeweils ca. 9:00 Uhr

### Web-Seite

<http://fhm.hgesser.de/>  
(Folien, Skript, Informationen)

## Zur Vorlesung (3)

### Praktika:

- 2 Termine (mittwochs) in diesem Semester
- An Linux-PCs  
(Rechnerraum 3.026 von Prof. Schnörr)
- Praktische Beispiele in **Python** umsetzen
- Teilnahme freiwillig, nicht prüfungsrelevant

Ankündigung der Praktikumstermine:  
Montags vorher; auf der Web-Seite

## Zur Vorlesung (4)

### Vernetzung:

- Teilnehmerliste ausfüllen  
(nur Teilnehmer, die **nicht** BS I bei mir hören)
- Name, Mail-Adresse (alle Angaben freiwillig)
- News-Verteiler
- Teilnehmerdaten öffentlich

### Mailingliste:

Bei Bedarf Einrichtung einer Mailingliste zur Diskussion der Themen

## Über den Dozenten

### Hans-Georg Eßer

- Dipl.-Math. (RWTH Aachen, 1997)  
Dipl.-Inform. (RWTH Aachen, 2005)
- Chefredakteur Computerzeitschrift (2000- )
- Autor diverser Computerbücher
- Lehrauftrag an der FH München:  
Betriebssysteme I / II
- ~~Betreuung von Diplomarbeiten~~

## Zur Vorlesung (5)

### Sprechstunde: –

**Fragen:**  
direkt in der Vorlesung (Handzeichen)  
oder danach  
oder per E-Mail

## Gliederung

```
Sep 19 14:20:18 amd64 sshd[20494]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61557
Sep 19 14:27:43 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 01:00:01 amd64 /usr/sbin/cron[29278]: (root) CMD (/sbin/evlogmgr -c "severity=DRBUO")
Sep 20 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 02:00:01 amd64 /usr/sbin/cron[30133]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 20 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:46:44 amd64 sshd[6516]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62004
Sep 20 12:46:44 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:48:41 amd64 sshd[6609]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62105
Sep 20 12:54:44 amd64 sshd[6694]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62514
Sep 20 15:27:35 amd64 sshd[9077]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64242
Sep 20 15:27:35 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 sshd[10102]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63375
Sep 20 16:37:11 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:38:10 amd64 sshd[10140]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63546
Sep 21 01:00:01 amd64 /usr/sbin/cron[17055]: (root) CMD (/sbin/evlogmgr -c "severity=DRBUO")
Sep 21 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 02:00:01 amd64 /usr/sbin/cron[17878]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 21 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 sshd[31088]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63397
Sep 21 17:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:53:39 amd64 sshd[31269]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 21 18:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 19:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 01:00:01 amd64 /usr/sbin/cron[4674]: (root) CMD (/sbin/evlogmgr -c "severity=DRBUO")
Sep 22 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 /usr/sbin/cron[5489]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 22 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 20:23:21 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 01:00:01 amd64 /usr/sbin/cron[24739]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 23 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 02:00:01 amd64 /usr/sbin/cron[25555]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 23 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:05 amd64 sshd[6584]: Accepted publickey for esser from ::ffff:87.234.201.207 port 63709
Sep 23 18:04:05 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:34 amd64 sshd[6606]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62093
Sep 24 01:00:01 amd64 /usr/sbin/cron[12436]: (root) CMD (/sbin/evlogmgr -c "severity=DRBUO")
Sep 24 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 02:00:01 amd64 /usr/sbin/cron[13283]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 24 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 11:15:48 amd64 sshd[20998]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64456
Sep 24 11:15:48 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 13:49:08 amd64 sshd[23197]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61330
Sep 24 13:49:08 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_midi_event: unsupported module, tainting kernel.
Sep 24 15:42:07 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_oss: unsupported module, tainting kernel.
Sep 24 20:25:31 amd64 sshd[29399]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62566
Sep 24 20:25:31 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 01:00:02 amd64 /usr/sbin/cron[6621]: (root) CMD (/sbin/evlogmgr -c "severity=DRBUO")
Sep 25 01:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 02:00:01 amd64 /usr/sbin/cron[1484]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 25 02:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:25 amd64 sshd[8889]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64183
Sep 25 10:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:47 amd64 sshd[8921]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64253
Sep 25 11:30:02 amd64 sshd[9372]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62029
Sep 25 11:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:05:37 amd64 sshd[11554]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62822
Sep 25 14:05:37 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:06:10 amd64 sshd[11586]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62951
Sep 25 14:07:17 amd64 sshd[11608]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63392
Sep 25 14:08:33 amd64 sshd[11630]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63709
Sep 25 15:25:33 amd64 sshd[12930]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62778
```

# Gliederung

## Betriebssysteme I:

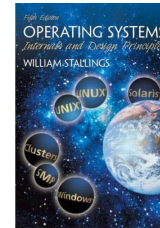
1. Einleitung
2. Prozesse und Threads
3. Interrupts
4. Scheduler
5. Synchronisation
6. Interprozess-Kommunikation (IPC)
7. Deadlocks

## Betriebssysteme II:

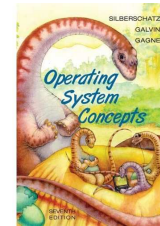
### 8. Speicherverwaltung

# Literatur (2)

**Operating Systems  
Internals and Design Principles  
(Stallings)**  
Prentice Hall, ca. 80 Euro  
(englisch)

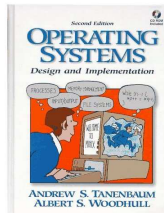


**Operating System Concepts  
(Silberschatz, Galvin, Gagne)**  
Wiley, ca. 52 Euro  
(englisch)

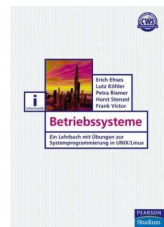


# Literatur (1)

**Operating Systems  
Design and Implementation  
(Tanenbaum, Woodhull)**  
Prentice Hall  
(englisch)



**Betriebssysteme  
Ein Lehrbuch mit Übungen zur System-  
programmierung in Unix/Linux (Ehse et al.)**  
ISBN 3-8273-7156-2  
Pearson Studium, 30 Euro



# Inhalte

```
Sep 19 14:27:41 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 01:00:01 amd64 /usr/sbin/cron[29278]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 20 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 02:00:01 amd64 /usr/sbin/cron[30131]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 20 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:46:44 amd64 sshd[6516]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62004
Sep 20 12:46:44 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:48:41 amd64 sssd[6609]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62105
Sep 20 12:54:44 amd64 sssd[6694]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62514
Sep 20 15:27:35 amd64 sssd[9071]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64242
Sep 20 15:27:35 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 sssd[10102]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63375
Sep 20 16:37:11 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:38:10 amd64 sssd[10140]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63546
Sep 21 01:00:01 amd64 /usr/sbin/cron[17055]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 21 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 02:00:01 amd64 /usr/sbin/cron[17878]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 21 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 sssd[31088]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63397
Sep 21 17:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:53:39 amd64 sssd[31269]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 21 18:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 18:43:26 amd64 sssd[31088]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 22 01:00:01 amd64 /usr/sbin/cron[4674]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 22 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 /usr/sbin/cron[5889]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 22 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 /usr/sbin/cron[24719]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 23 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 02:00:01 amd64 /usr/sbin/cron[25555]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 23 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:05 amd64 sssd[6554]: Accepted publickey for esser from ::ffff:87.234.201.207 port 62093
Sep 23 18:04:05 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:24 amd64 sssd[6561]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62093
Sep 24 01:00:01 amd64 /usr/sbin/cron[12436]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 24 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 02:00:01 amd64 /usr/sbin/cron[13253]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 24 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 11:15:48 amd64 sssd[20998]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64456
Sep 24 11:15:48 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 13:49:08 amd64 sssd[19371]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61330
Sep 24 13:49:08 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: amd_seg_mid1_event: unsupported module, tainting kernel.
Sep 24 15:42:07 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: amd_seg_oss: unsupported module, tainting kernel.
Sep 24 20:25:31 amd64 sssd[29399]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62566
Sep 24 20:25:31 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 01:00:02 amd64 /usr/sbin/cron[662]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 25 01:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 02:00:01 amd64 /usr/sbin/cron[1484]: (root) CMD (/sbin/evlogmgr -c "age > *30d*")
Sep 25 02:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:25 amd64 sssd[8889]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64183
Sep 25 10:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:47 amd64 sssd[9321]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64253
Sep 25 11:30:02 amd64 sssd[9372]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62029
Sep 25 11:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:05:37 amd64 sssd[11554]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62822
Sep 25 14:05:37 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:06:10 amd64 sssd[11586]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62951
Sep 25 14:07:17 amd64 sssd[11608]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63193
Sep 25 14:08:37 amd64 sssd[11611]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63270
Sep 25 14:08:37 amd64 sssd[11611]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63270
```

## Inhaltsübersicht

- Zusammenhängende Speicherzuteilung
  - Partitionen fester Größe
  - Partitionen variabler Größe
  - Methoden zur Verwaltung des freien Speichers
  - Segmentierung
- Nicht zusammenhängende Speicherzuteilung
  - Virtuelle Speicherverwaltung (Paging)
  - Mehrstufiges Paging
  - Segmentierung mit Paging
- Demand Paging
  - Page Faults und deren Behandlung
  - Strategien für die Seitenersetzung
  - Weitere Design-Möglichkeiten
- Swapping

## Arten der Speicherverwaltung

Zwei prinzipielle Arten der Speicherverwaltung:

- **Zusammenhängende Speicherzuteilung**
  - Jede Anforderung eines Prozesses nach einer bestimmten Menge Speicher muss das BS durch zusammenhängenden (contiguous) Speicher erfüllen.
- **Nicht zusammenhängende Speicherzuteilung**
  - BS kann Speicheranforderung durch Zuweisung mehrerer kleiner Speicherbereiche erfüllen, die zusammen die geforderte Menge Speicher ergeben.
  - Wiederauffinden der verstreuten Speicherbereiche ist eine Aufgabe der Speicherverwaltung.

## Motivation

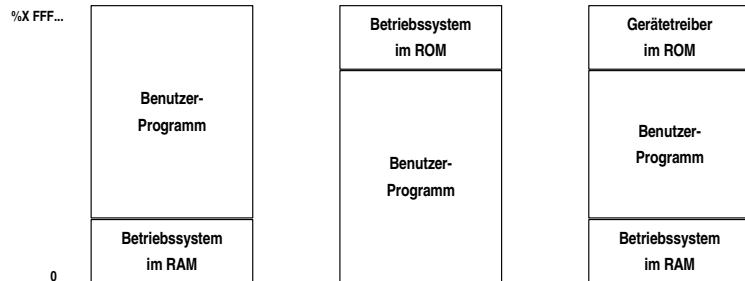
- Wie setzt sich eine **Speicheradresse** zusammen?
- Was geschieht bei einem Adresszugriff?
- Wie kann man als Administrator oder Software-Entwickler Nutzen aus Kenntnissen der Speicher-verwaltung ziehen?
  - wie funktioniert **Shared Memory**?
  - was sind **Memory-mapped Files**?
- Wie entsteht ein **Segmentation Fault**?
- Wie erreicht man einen **virtuellen Adressraum**?

## Arten der Speicherverwaltung

- Heute: Hauptspeicher fast immer **nicht-zusammenhängend** (virtuelle Speicher-verwaltung).
- Zusammenhängende Speicher-verwaltung bei
  - Verwaltung der sog. „Pools“ innerhalb des Hauptspeichers (siehe weiter unten)
  - Verwaltung von Plattenplatz,
  - Verwaltung des Platzes in Page- und Swap-Dateien.

# Single-Tasking ohne Swapping und Paging

- Aufteilung des Speichers (RAM und ROM) in Bereiche für
  - das Betriebssystem
  - ein Benutzerprogramm
- Beim Beenden eines Programmes: Überlagern des Programmbereichs durch ein neues Programm.



# Multiprogramming

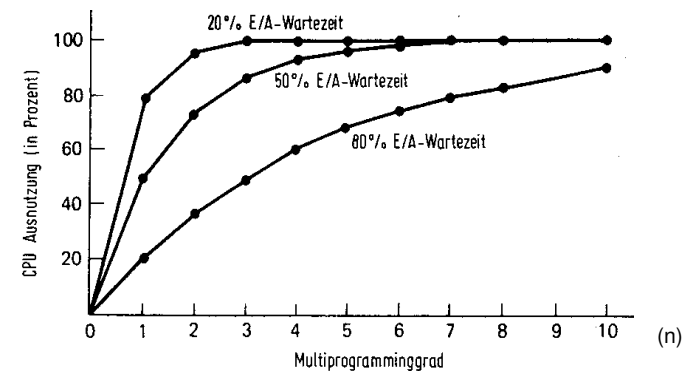
- Programme verbringen einen großen Teil ihrer Ausführungszeit mit Warten (auf I/O etc.).
- **Auslagern (Swapping)** auf Platte bei jedem Warten ist ineffizient.
- Lösung: mehrere Programme gleichzeitig im Speicher.
- Voraussetzung: **verschiebbarer Code** und **Speicherschutz**.

# Speicherbelegung unter MS-DOS

BIOS-ROM	100000 (1 MB)
Andere ROM-Bereiche	C0000
Bildschirm-RAM	A0000 (640 KB)
COMMAND.COM ladbarer Teil	
Freies RAM für die Anwendung	
DOS-Variablen, Puffer etc.	
MSDOS.SYS	
IO.SYS	
BIOS-Variablen	600
Interrupt-Vektoren	400
	0

# Multiprogramming

- Wenn  $n$  Programme den Bruchteil  $p$  ihrer Zeit mit Warten verbringen, ist die CPU-Ausnutzung  $= 1 - p^n$ .



# Code-Verschiebung: Z80-Assembler

```
#base 0xA000 ; Programmstart bei 0xA000

ld hl,data      A000 ld hl,0xA011 ; data
ld a,(hl)      A003 ld a,(hl)
call print     A004 call 0xA008 ; print
ret           A007 ret

:print ; Funktion print
cmp a,' '      A008 cmp a,' '
jne ende      A00A jne 0xA010 ; ende
call OS_PRINT A00D call OS_PRINT
:ende ; Label ende
ret           A010 ret

:data ; Daten
char 'x'      A011 char 'x'
```

# Code-Verschiebung: Z80-Assembler

Probleme mit nachträglicher Relokation:

- Was sind Adressen?
- Sprung- und Funktionsaufruf-Adressen
- Zugriff auf Datenbereiche
- Was sieht nur zufällig wie eine Adresse aus?
- Mehrfach indirekte Adressierung  
(„lade den Wert, der in der Speicherstelle steht, deren Adresse hier steht“)

Besser: Lade-System plant Relokation von vornherein ein, merkt sich Stellen mit Adressen

# Relocator für Z80, 19 Jahre alt



**für 464-664-6128**

```
1000 ' ***** [1164]
1010 ' *** RELOCATOR V1.0 *** [51]
1020 ' *** [518]
1030 ' *** [51]
1040 ' *** [51]
1050 ' *** Geschrieben im *** [2088]
1060 ' *** Oktober 1987 von *** [960]
1070 ' *** [51]
1080 ' *** Hans-Georg Eßer *** [687]
1110 ' *** [51]
1120 ' ***** [1164]
1130 ' [117]
1140 MODE 2:ZONE 3 [1805]
1150 WINDOW #1,48,76,7,10 [1139]
1160 WINDOW #2,48,76,12,18 [1223]
1170 FOR n=1 TO 2:PEN #n,0:PAPER #n,1:NEXT [2164]
1180 PRINT CHR$(24);STRINGS(80,216); [8496]
Relocator V1.0 (c) 1987 by Hans-Geor
g Eßer Relocator V1.0 ;STRINGS
(80,216);CHR$(24);
1190 LOCATE 1,22 [730]
1200 PRINT CHR$(24);STRINGS(80,216); [8829]
Relocator V1.0 (c) 1987 by Hans-Geor
g Eßer Relocator V1.0 ;STRINGS
(80,216);CHR$(24);
1210 WINDOW 1,48,5,20 [1040]
1220 CLS #1:CLS #2 [53]
1230 LOCATE #1,2,2:PRINT #1,"Aktuelle Adre [2664]
ese:"
1240 LOCATE #2,2,2:PRINT #2,"Geänderte Ad [3236]
ressen:";PRINT #2
1250 LOCATE 5,3:LINE INPUT "Name des Binne [4640]
r-Files > ",fil#8
1260 LOCATE 5,5:INPUT "Ursprungsadresse [4501]
> ",adr1
1270 LOCATE 5,7:INPUT "Programm-Laenge (in [3793]
Bytes) > ",length
1280 LOCATE 5,9:INPUT "Neue Adresse [3384]
> ",adr2
1290 LOCATE 5,11:LINE INPUT "Name des neue [2549]
n Files > ",fil#28
1300 LOCATE 5,14:PRINT "Relokation wird du [5399]
rgehört."
1310 ' Einlesen [1453]
1320 MEMORY adr1-1 [1089]
1330 LOAD fil#8,adr1 [518]
1340 diff=adr2-adr1 [566]
1350 ' Alle Adressen muessen um diff nach [4743]
oben verschoben werden.
1360 FOR n=adr1 TO adr1+length [2180]
1370 LOCATE #1,20,2:PRINT #1,"&";HEX$(n,4) [2013]
NEXT n
1380 GOSUB 1480 ' ueberpruefen und ggf. ae [2409]
ndern
Listing Relocator
```

# Code-Verschiebung und Speicherschutz (1/2)

- Ein Programm muss an verschiedenen Stellen im Speicher laufen können.

Zwei Möglichkeiten:

- Linker vermerkt, welche Code-Stellen absolute Adressen sind. Beim Laden des Programms werden diese Stellen entsprechend abgeändert.
- Der Rechner hat ein spezielles Hardware-Register, ein sog. **Basisregister**

Bei jeder Adressberechnung (zur Laufzeit) wird die Adresse im Basisregister zu der Adresse im Programm addiert.



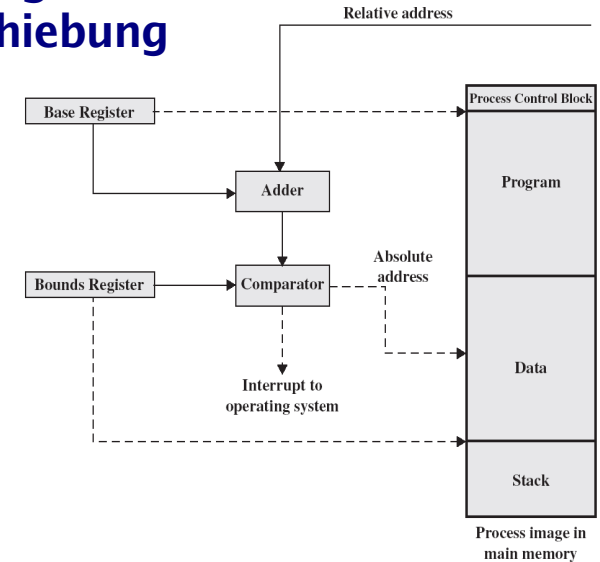
# Code-Verschiebung und Speicherschutz (2/2)

- Ein Programm darf nicht auf den Speicherbereich eines anderen Programms zugreifen.

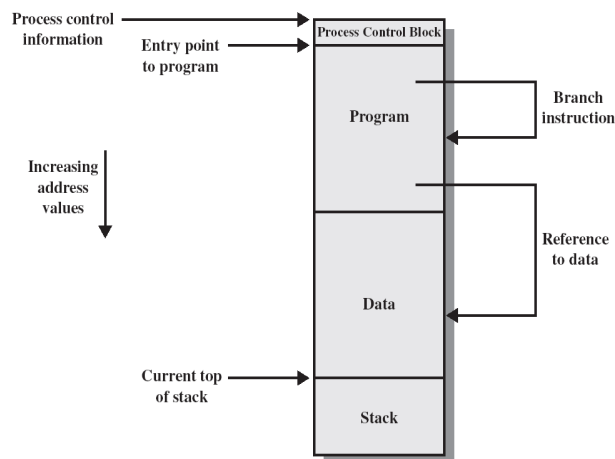
Zwei Möglichkeiten dies zu erreichen:

- Schutzcodes
- Der Rechner hat ein spezielles Hardware-Register, ein sog. **Längenregister**  
Durch Überprüfen des Längenregisters wird festgestellt, ob die zugriffene Adresse in der dem Programm zugewiesenen Partition liegt.

# Unterstützung für Code-Verschiebung und Speicherschutz



# Code-Verschiebung: Prozess - Adressierungsanforderungen



```

Sep 19 14:20:18 amd64 sbdl[20494]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61557
Sep 19 14:27:41 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 01:00:01 amd64 /usr/sbin/cron[29278]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUO")
Sep 20 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 02:00:01 amd64 /usr/sbin/cron[30103]: (root) CMD (/sbin/evlogmgr -c "age > *30d")
Sep 20 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:46:44 amd64 sbdl[6516]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62004
Sep 20 12:46:44 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:48:43 amd64 sbdl[6699]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62105
Sep 20 12:54:44 amd64 sbdl[6694]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62514
Sep 20 15:27:35 amd64 sbdl[9077]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64242
Sep 20 15:27:35 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 sbdl[10102]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63375
Sep 20 16:37:11 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:38:10 amd64 sbdl[10140]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63546
Sep 21 01:00:01 amd64 /usr/sbin/cron[17055]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUO")
Sep 21 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 02:00:01 amd64 /usr/sbin/cron[17878]: (root) CMD (/sbin/evlogmgr -c "age > *30d")
Sep 21 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 sbdl[31088]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63397
Sep 21 17:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:53:39 amd64 sbdl[31699]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 21 18:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 19:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 01:00:01 amd64 /usr/sbin/cron[43242]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUO")
Sep 22 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 /usr/sbin/cron[50555]: (root) CMD (/sbin/evlogmgr -c "age > *30d")
Sep 22 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 20:23:21 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 01:00:01 amd64 /usr/sbin/cron[24730]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUO")
Sep 23 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 02:00:01 amd64 /usr/sbin/cron[25555]: (root) CMD (/sbin/evlogmgr -c "age > *30d")
Sep 23 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:05 amd64 sbdl[8554]: Accepted rsa for esser from ::ffff:192.168.1.5 port 59771 ssh2
Sep 23 18:04:05 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:34 amd64 sbdl[6606]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62093
Sep 24 01:00:01 amd64 /usr/sbin/cron[12436]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUO")
Sep 24 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 02:00:01 amd64 /usr/sbin/cron[13253]: (root) CMD (/sbin/evlogmgr -c "age > *30d")
Sep 24 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 11:15:48 amd64 sbdl[20998]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64456
Sep 24 11:15:48 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 13:49:08 amd64 sbdl[23197]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61330
Sep 24 13:49:08 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 Kernel: and_seq_midi_event: unsupported module, tainting kernel.
Sep 24 15:42:07 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 Kernel: and_seq_midi_event: unsupported module, tainting kernel.
Sep 24 20:25:31 amd64 sbdl[29399]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62566
Sep 24 20:25:31 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 01:00:02 amd64 /usr/sbin/cron[6621]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUO")
Sep 25 01:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 02:00:01 amd64 /usr/sbin/cron[1484]: (root) CMD (/sbin/evlogmgr -c "age > *30d")
Sep 25 02:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:25 amd64 sbdl[8889]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64183
Sep 25 10:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:47 amd64 sbdl[8921]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64253
Sep 25 11:30:02 amd64 sbdl[9372]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62029
Sep 25 11:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:05:37 amd64 sbdl[11554]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62822
Sep 25 14:05:37 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:06:10 amd64 sbdl[11586]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62951
Sep 25 14:07:17 amd64 sbdl[11608]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63392
Sep 25 14:08:33 amd64 sbdl[11630]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63709
Sep 25 15:25:33 amd64 sbdl[12930]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62778
    
```

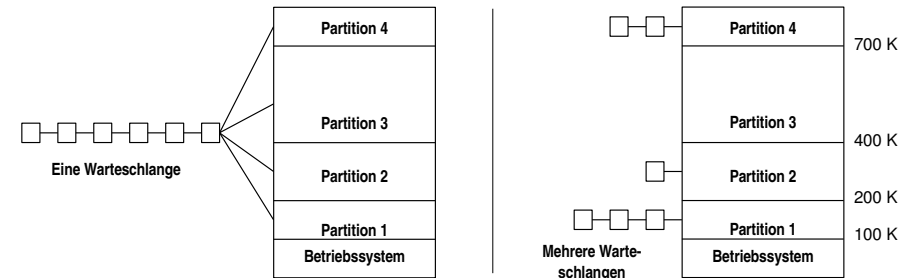
# Zusammenhängende Speicheraufteilung

# Aufteilung in feste Partitionen

- Aufteilung des Speichers in Partitionen fester (gleicher oder ungleicher) Größe.
- Zuweisung eines Programms zu einer freien Partition. Alternativen:
  - Erstes Programm, das in die freie Partition passt (eine Warteschlange)
  - FIFO für jede einzelne Partition (mehrere Warteschlangen)
  - Größtes Programm, das in die freie Partition passt
    - Nachteil: kleine Programme werden zurückgestellt
    - Lösung: kleines Programm nur k-mal überspringen

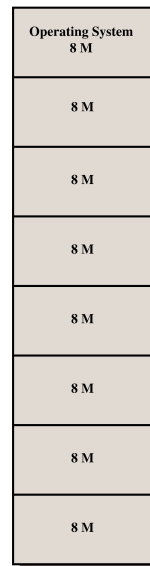
# Aufteilung in feste Partitionen

- Evtl. große freie Bereiche in der Partition: **interne Fragmentierung**.

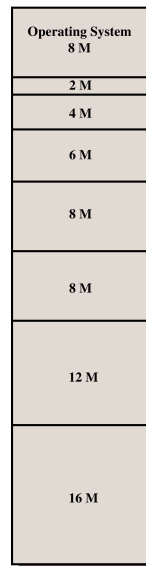


# Aufteilung in feste Partitionen

- **Gleiche Größe**
  - Verschwendung bei kleinen Programmen
  - Programme passen in jede freie Partition
- **Ungleiche Größe**
  - Bessere Speicherausnutzung
  - Evtl. ungeschickte Belegung



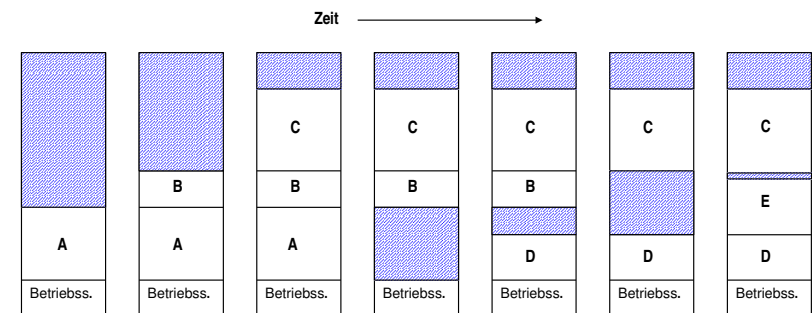
(a) Equal-size partitions



(b) Unequal-size partitions

# Aufteilung in variable Partitionen (1)

- Anzahl und Größe der Partitionen werden dynamisch festgelegt.





## Aufteilung in variable Partitionen (2)

- Es bleiben u. U. viele kleine freie Bereiche im Hauptspeicher (Löcher). Dies wird als **externe Fragmentierung** bezeichnet.
- Evtl. müssen diese Löcher durch Verschieben der Partitionen entfernt werden (**memory compaction**).

## Aufteilung in variable Partitionen (4)

- Wenn keine ausreichend große Partition zur Verfügung steht, müssen ein oder mehrere Prozesse auf Platte ausgelagert werden (**Swapping**).
- Es kann auch dem Prozess von vornherein ein größerer Speicherbereich zugewiesen werden, als angefordert.
  - interne Fragmentierung.
  - Wenn auch dieser größere Bereich nicht ausreicht, muss wieder eines der vorher beschriebenen Verfahren angewandt werden.

## Aufteilung in variable Partitionen (3)

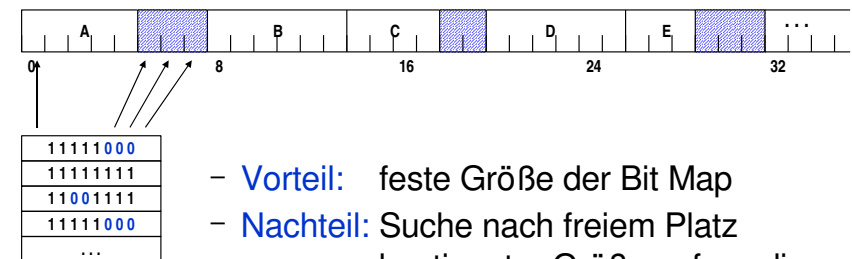
Was ist, wenn der Speicherbedarf eines Prozesses wächst?

- Wenn neben der Partition ein freier Speicherbereich ist, kann die Partition vergrößert werden.
- Es kann eine neue, ausreichend große Partition reserviert und der Inhalt dorthin verschoben werden.

## Welche Bereiche sind frei?

### a) Bit Maps

- Aufteilung des Speichers in Einheiten (einige Byte bis einige KByte)
- Ein Bit pro Einheit: **0** wenn die Einheit frei ist  
**1** wenn die Einheit belegt ist



- **Vorteil:** feste Größe der Bit Map
- **Nachteil:** Suche nach freiem Platz bestimmter Größe aufwendig.

# Welche Bereiche sind frei?

## b) Linked Lists

- Verkettete Liste von Beschreibungen der Speicherbereiche:
  - Von Prozess belegt (P) oder frei (H=hole)
  - Anfangsadresse und Länge des Bereichs
  - Zeiger auf nächsten Eintrag

