

Script generated by TTT

Title: Baumgarten: GBS (22.11.2013)

Date: Fri Nov 22 08:31:44 CET 2013

Duration: 87:55 min

Pages: 55

A leading institution at the heart of the digital society



Top reasons to study at EURECOM

Exchange options for TUM students

November 2013

A UNIQUE CONSORTIUM MODEL

- EURECOM is a Graduate school and Research Center created in 1992, specialized in **Mobile Communications, Multimedia Communications, Network & Security**. It is administered by a **CONSORTIUM**.

EURECOM's CONSORTIUM
A successful international joint venture between

Renowned industrial partners



Prestigious academic partners



Each member guides EURECOM's teaching and research strategy

- Ensure that EURECOM's teaching is professionally relevant and line with the demands of the industry.
- Provide internship opportunities, research projects
- Participate in the courses
- Regularly assess and validate curricula at EURECOM
- Ensure an easy transfer of credits and total mutual recognition
- EURECOM is more than an academic partner, it is the offshore campus of its academic partners.

EURECOM and TUM: a rich cooperation

- 2007: TUM enters EURECOM's Consortium

Two major recent projects

Smart Mobility 2020

July 2013: Franco-German Summer School 2013 for Junior scientists



- ICT for sustainable mobility
- Vehicular cloud computing and big data analysis
- Vehicular urban detection in smart cities
- Mobility - Combined transportation

PROTON-PLATA

Research project of DEUFRAKO, a Franco-German cooperation in traffic research

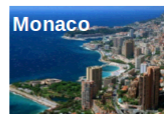


- programmable telematics unit based on SDR (software defined radio)
- project funded by the German Federal Ministry for Economics and Technology and the Agence Nationale de la Recherche (ANR)

1. STUDY IN AN UNCOMPARABLE SETTING

French Riviera - Cote d'Azur

- Enjoy 300 days of sunshine per year ☀️
- Ideal for sport lovers:
- 10 min from the beaches / 1 hour from the sky slopes
- 40 min from Nice airport (2nd international airport in France)
- Easy access: 1h30 flight from MUNICH to NICE



« it's one of the best places to be in Europe ».
Manuel, TUM student at EURECOM 2013

The "French Silicon Valley" SOPHIA ANTIPOLIS

- EURECOM is located in the heart of the largest European High Tech Park
 - Dedicated to information technology, life and environmental sciences, fine chemicals
- 1,452 companies, 40% of companies involved in R&D activities
- 31,000 jobs
- 4,500 researchers
- More than 5,000 students
- 70 nationalities



Some of Sophia Antipolis' leading companies



EURECOM

EURECOM BUILDING SophiaTech CAMPUS

First class facilities : a brand new building



EURECOM

2. GAIN A TRULY INTERNATIONAL EXPERIENCE

« This made me gain not only acquaintances and friends, but additionally a professional network, which is a very unique aspect of EURECOM. This is especially interesting in the light of EURECOM's international diversity »

Fabian, TUM student at EURECOM 2012

« Here at EURECOM the international experience exchange students usually quest for is lived in an intensity I couldn't imagine before. Due to the fact, that the majority of students and professors comes from a multitude of other countries, I also got the opportunity to get a picture of many different places of the world every day ».

Denis, TUM student at

EURECOM

A CUTTING EDGE AND FLEXIBLE CURRICULUM

To help students select their courses, the curriculum is organized around **7 teaching tracks**, made of mandatory and optional courses.

« Despite the small size of the institute, I was surprised by the breadth of the courses offered. I found a lot more interesting topics than I could possibly fit into my schedule.»
Denis, TUW student at EURECOM 2010



Bernard Merialdo

- [Web Engineering](#) (Mobile application and service, and internet..)



David Gesbert

- [Mobile Communications](#) (Digital communications, Statistical signal processing...)



Jean-Luc Dugelay

- [Multimedia](#) (Multimedia indexing and retrieval; 3-D and virtual imaging (analysis and synthesis)



Ernst Biersack

- [Networking](#) (Software development methodologies, Distributed Systems and Cloud Computing...)



Raymond Knopp

- [Real-time and embedded systems](#) (Modern computer architecture, Operating systems...)



Refik Molva

- [Communication System Security](#) (Cyber-crime and Computer Forensics; Secure communications...)



Dirk Slock

- [Transmission technologies](#) (Digital communications, information theory...)

EURECOM

EXCHANGE OPTIONS at EURECOM

1. ERASMUS STUDIES, 2 options:

2. Master's Thesis

3. Double degree (French Master's degree delivered by TelecomParistech)



EURECOM

EXCHANGE OPTIONS at EURECOM

1. ERASMUS STUDIES, 2 options:

- 6 months (October- February)

**1 Academic semester
30 ECTS**

2. Master's Thesis

3. Double degree (French Master's degree delivered by TelecomParistech)



EURECOM

CURRICULUM'S STRUCTURE

Technical courses

Innovative courses, designed in partnership with EURECOM's members and regularly assessed and updated

Non technical courses

« Soft skills » classes taught by experts from companies or professors from renowned Business Schools in Law, Management, Business simulation...

EURECOM

CURRICULUM's STRUCTURE

Technical courses

Innovative courses, designed in partnership with EURECOM's members and regularly assessed and updated

Non technical courses

« Soft skills » classes taught by experts from companies or professors from renowned Business Schools in Law, Management, Business simulation...

Semester Project

Hands-on practise and development of an innovative study project under the supervision of a professor. (200h)

CURRICULUM's STRUCTURE

Technical courses

Innovative courses, designed in partnership with EURECOM's members and regularly assessed and updated

Non technical courses

« Soft skills » classes taught by experts from companies or professors from renowned Business Schools in Law, Management, Business simulation...

Semester Project

Hands-on practise and development of an innovative study project under the supervision of a professor. (200h)

Master's thesis

Thesis on a specific industrial or research subject carried out in a company or a laboratory during 6 months under the supervision of both TUM and EURECOM's professors.

CURRICULUM's STRUCTURE

Technical courses

Innovative courses, designed in partnership with EURECOM's members and regularly assessed and updated

Non technical courses

« Soft skills » classes taught by experts from companies or professors from renowned Business Schools in Law, Management, Business simulation...

Semester Project

Hands-on practise and development of an innovative study project under the supervision of a professor. (200h)

Master's thesis

Thesis on a specific industrial or research subject carried out in a company or a laboratory during 6 months under the supervision of both TUM and EURECOM's professors.

Foreign languages

Any foreign language can be taught provided there are 5 students min. (20h)

SEMERSTER's PROJECTS



During your studies at EURECOM, student have to pick up a project which is highly related to research activity and/or industrial partnership.

Duration : 200 h/ student

The project is a blend of theoretical and practical work, which often involves design of a prototype.

A recent example of a EURECOM's project led by TUM Student, Marie-Lena LECKERT and presented during a IEEE conference:

“Facial cosmetics database and impact analysis on automatic face recognition”



15th International Workshop on Multimedia Signal Processing, September 30-October 2, 2013, Pula, Italy

« My semester project at EURECOM had a big impact on my life. It wasn't only fun and great to follow and realize my own ideas under professional guidance; I also gained a lot of experience in research through the following paper and conference attendance. »

Marie Lena, EURECOM 2013  EURECOM

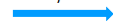
SMALL SIZE = GREATER ACCESSIBILITY

▪ Accessible professors:

EURECOM has one of the **best students/professors ratio** among French higher education institutions (**9:1**). Professors are easily accessible.

Each exchange student has a tutor professor whose role is to advise about classes, study materials, notation....

TUM STUDENTS



Professor Ernst BIRSACK



▪ A bilingual administration team totally dedicated to international students:

Our staff can provide you with personal assistance at all stages of your studies:

- Help you find accommodation and apply for accommodation financial aid* (CAF allocation)
- Open a bank account
- Find an internship
- Find good deals (students discounts on bus...)



* Each student in France, both French and foreign, is entitled to receive financial aid for housing (20% off the rent). More information: www.caf.fr



STUDENT ASSOCIATION CALENDAR

The student's association organises many events all year long:



Weekend of Integration



EURECOM's discovery weekend

Graduation Ceremony

International study trip

International lunch

End of the year weekend



Many sport activities

Famous Melody Villa parties!



Week End trips



MAIN ASSETS

- ✓ 100% teaching in English: **no need to speak French!**
- ✓ Coursework fully evaluated/recommended with TUM faculty, easy credit transfer
- ✓ Benefiting from teaching at the cutting-edge linked to internationally-leading research led by **renowned researchers**
- ✓ An excellent student/ faculty ratio: easy access to professors
- ✓ A human-scale structure: favors long-term friendship with international classmates and strong professional network
- ✓ Expertise of renowned industrial partners
- ✓ Unique location: **Sophia Antipolis** is considered a major European region for telecommunications research
- ✓ Enjoy some time in one of the world's most popular tourist areas

« I would say that these years have been the best of my life and can only recommend you to come here if you are interested in living in an **international environment** as well as doing studies in the field of networking and security. I decided in the end to stay for three more years and do my PhD at **Eurecom** ».



CONTACTS

▪ EURECOM

Academic information	Prof. Ernst BIRSACK Ernst.biersack@eurecom.fr
Administrative information	Caroline Hanras caroline.hanras@eurecom.fr
List of tracks and course content	http://www.eurecom.fr/en/teaching/engineering-studies/18-months-curriculum

▪ TUM

Study directions Informatics Department	Dr Angelika REISER reiser@in.tum.de Martina VON IMHOFF imhoff@in.tum.de
Study directions Electrical Engineering and Information Technology Department	Prof. Eckehard STEINBACH eckehard.steinbach@tum.de Heike ROTH Abroad@ei.tum.de



novembre TUMEURECOM.ppt - LibreOffice Impress

Datei Bearbeiten Ansicht Einfügen Format Extras Bildschirmpräsentation Fenster Hilfe

Folien

Normal Gliederung Notizen Handzettel Foliensortierung

Aufgabenbereich Anzeige

Masterfolien
Layouts
Tabellen

Designs

Anzeigen

Kopfzeile
 Summenzeile
 Wechselnde Zeilen
 Erste Spalte
 Letzte Spalte
 Wechselnde Spalten
 Benutzerdefinierte Animation
 Folienubergang

CONTACTS

EURECOM

Academic information Prof. Ernst **BIERSACK**
Ernst.biersack@eurecom.fr

Administrative information **Caroline Hanras**
caroline.hanras@eurecom.fr

List of tracks and course content <http://www.eurecom.fr/en/teaching/engineering-studies/18-months-curriculum>

TUM

Study directions **Dr Angelika REISER**
reiser@in.tum.de

Informatics Department **Martina VON IMHOFF**
imhoff@in.tum.de

Study directions **Prof. Eckehard STEINBACH**
eckehard.steinbach@tum.de

Electrical Engineering and Information Technology Department **Heike ROTH**
Abroad@ei.tum.de

8,64 / 15,68 0,00 x 0,00 Folie 15 / 15 Standard 53%
Menu [TUM... [gbs... nove... Fr, 22. Nov, 08:41

novembre TUMEURECOM.ppt - LibreOffice Impress

TUM-GBS-Fol-2013-ALLE-1-145.pdf — TUM-GBS-Fol-2013.ppt [Kompatibilitätsmodus]

Datei Bearbeiten Ansicht Gehe zu Lesezeichen Hilfe

Vorherige Nächste 1 (1 von 145) Auf Seitenbreite einpassen

Vorschaubilder

Technische Universität München

**Grundlagen:
Betriebssysteme und Systemsoftware
(GBS)**

Uwe Baumgarten

6,59 / 22,36 0,00 x 0,00 Folie 15 / 15 Standard 53%
Menu [TUM... [gbs... nove... Fr, 22. Nov, 08:41

Technische Universität München

**Grundlagen:
Betriebssysteme und Systemsoftware
(GBS)**

Uwe Baumgarten

© UB TUM GBS WS 2013/14 Grundlagen:
Betriebssysteme und Systemsoftware (IN0009)

1

novembre TUMEURECOM.ppt - LibreOffice Impress

TUM-GBS-Fol-2013-ALLE-1-145.pdf — TUM-GBS-Fol-2013.ppt [Kompatibilitätsmodus]

Datei Bearbeiten Ansicht Gehe zu Lesezeichen Hilfe

Vorherige Nächste 1 (1 von 145) Auf Seitenbreite einpassen

Vorschaubilder

Technische Universität München

**Grundlagen:
Betriebssysteme und Systemsoftware
(GBS)**

Uwe Baumgarten

6,59 / 22,36 0,00 x 0,00 Folie 15 / 15 Standard 53%
Menu [TUM... [gbs... nove... Fr, 22. Nov, 08:42

Einführung

- Motivation
 - Unmittelbare Nutzung des Arbeitsspeichers (physische Adressen) nicht empfehlenswert
 - Abstraktion der Hardware-Eigenschaften nötig
 - Umgang mit Kapazitätsengpässen
 - Adressräume
 - Maschinenadressraum (Arbeitsspeicher, AS)
 - Fortlaufend nummerierte Bytes
 - Start mit Maschinen-Adresse 0 (reale Adresse)
 - Programmadressraum (i.d.R. virtueller Adressraum)
 - Programmadressen, prozessspezifisch
- Aufgabe der Abbildung der Adressräume: Adressierung (Adr.)
- Direkte Adr., Basis-Adr., Seiten-Adr., Segment-Seiten-Adr.

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Einführung

- Organisation von Adressräumen
 - Programmcode
 - Datenbereich (statisch)
 - Datenbereich (dynamisch, Halde (Heap))
 - Laufzeitkeller (Stack)
- Single-Threaded Adressraum
 - [JS12, Kap. 5, p. 115]
- Multi-Threaded Adressraum
 - [JS12, Kap. 5, p. 116]
- Beispiele
 - Windows, Linux
 - [JS12, Kap. 5, p. 116]

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Schlichter, TU München

5.2. EINFÜHRUNG

Programm	statische Daten	dynamische Daten (Halde)	Keller 1	Keller n
----------	-----------------	--------------------------	----------	----------

niedrige Adresse

hohe Adresse

Für jeden Kontrollfluss (Thread) wird ein eigener Kellerbereich vorgesehen. Der Abstand zwischen den einzelnen Kellern wird meist standardmäßig vom System vorgegeben. Unabhängig von der Anzahl der Laufzeitkeller muss eine Überschneidung zwischen mehreren Kellern oder zwischen dem untersten Keller und der Halde vermieden werden.

Beispiel - Adressräume

Moderne Betriebssysteme stellen wenigstens 32 Bit große virtuelle Adressräume für die Anwendungen zur Verfügung, die jeweils in mehrere Bereiche unterteilt sind. Programmcode, statische Daten, Halde und Laufzeitkeller der Anwendung...

gbs_course-student.pdf

TUM-GBS-Fol-2013-ALLE-1-145.pdf — TUM-GBS-Fol-2013.ppt [Kompatibilitätsmodus]

Datei Bearbeiten Ansicht Gehe zu Lesezeichen Hilfe

Vorherige Nächste 133 (133 von 145) Auf Seitenbreite einpassen

Vorschaubilder

130

131

132

133

Technische Universität München

Einführung

- Organisation von Adressräumen
 - Programmcode
 - Datenbereich (statisch)
 - Datenbereich (dynamisch, Halde (Heap))
 - Laufzeitkeller (Stack)
- Single-Threaded Adressraum
 - [JS12, Kap. 5, p. 115]
- Multi-Threaded Adressraum
 - [JS12, Kap. 5, p. 116]
- Beispiele
 - Windows, Linux
 - [JS12, Kap. 5, p. 116]

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Beispiel - Adressräume

Moderne Betriebssysteme stellen wenigstens 32 Bit große virtuelle Adressräume für die Anwendungen zur Verfügung, die jeweils in mehrere Bereiche unterteilt sind. Programmcode, statische Daten, Halde und Laufzeitkeller der Anwendung

© UB TUM GBS WS 2013/14 Grundlagen

Fr. 22. Nov. 08:54

gbs_course-student.pdf

Datei Bearbeiten Ansicht Gehe zu Lesezeichen Hilfe

Vorherige Nächste 116 (123 von 228) 150%

Vorschaubilder

130

131

132

133

Schlitter, TU München

5.2. EINFÜHRUNG

Programm	statische Daten	dynamische Daten (Halde)	Keller 1	Keller n
----------	-----------------	--------------------------	----------	----------

niedrige Adresse

hohe Adresse

Für jeden Kontrollfluss (Thread) wird ein eigener Kellerbereich vorgesehen. Der Abstand zwischen den einzelnen Kellern wird meist standardmäßig vom System vorgegeben. Unabhängig von der Anzahl der Laufzeitkeller muss eine Überschneidung zwischen mehreren Kellern oder zwischen dem untersten Keller und der Halde vermieden werden.

32

Beispiel - Adressräume

Moderne Betriebssysteme stellen wenigstens 32 Bit große virtuelle Adressräume für die Anwendungen zur Verfügung, die jeweils in mehrere Bereiche unterteilt sind. Programmcode, statische Daten, Halde und Laufzeitkeller der Anwendung

© UB TUM GBS WS 2013/14 Grundlagen

Fr. 22. Nov. 08:55

Technische Universität München

Einführung

- Verwaltung der Halde (Heap)
 - Operation „malloc“
- Bereiche
 - Belegbereiche
 - Freibereiche
- Verwaltung der Freibereiche
 - Implizite (alle Bereiche) oder explizite Liste (nur Freibereiche)
- Strategie zur Auswahl
 - first-fit-Verfahren
 - next-fit-Verfahren
 - best-fit-Verfahren
 - worst-fit-Verfahren
- Fragmentierung

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

© UB TUM GBS WS 2013/14 Grundlagen:
Betriebssysteme und Systemsoftware (IN0009)

134

Technische Universität München

Einführung

- Verwaltung der Halde (Heap)
 - Operation „malloc“
- Bereiche
 - Belegbereiche
 - Freibereiche
- Verwaltung der Freibereiche
 - Implizite (alle Bereiche) oder explizite Liste (nur Freibereiche)
- Strategie zur Auswahl
 - first-fit-Verfahren
 - next-fit-Verfahren
 - best-fit-Verfahren
 - worst-fit-Verfahren
- Fragmentierung

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

© UB TUM GBS WS 2013/14 Grundlagen:
Betriebssysteme und Systemsoftware (IN0009)

134

Einführung

- Verwaltung der Halde (Heap)
 - Operation „malloc“
- Bereiche
 - Belegbereiche
 - Freibereiche
- Verwaltung der Freibereiche
 - Implizite (alle Bereiche) oder explizite Liste (nur Freibereiche)
- Strategie zur Auswahl
 - first-fit-Verfahren
 - next-fit-Verfahren
 - best-fit-Verfahren
 - worst-fit-Verfahren
- Fragmentierung

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Einführung

- Garbage Collection (in Java)
- Aufammeln von nicht mehr benötigtem Speicher
- Auffinden von nicht mehr benötigten Objekten
 - Hilfestellung: Pointer dorthin existieren nicht
- Repräsentation als Graph
 - Blöcke (Objekte) sind Knoten, Pointer sind Kanten
 - Mark and Sweep
 - Bestimmung erreichbarer Knoten

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Einführung

- Fragmentierung
 - Zerstücklung des freien Teils des Adressraums
- Externe Fragmentierung
 - Freibereiche zwischen belegten Bereichen
 - [JS12, Kap. 5, p. 120]
- Interne Fragmentierung
 - Freibereiche innerhalb fester (Rahmen-) Bereiche
 - [JS12, Kap. 5, p. 121]

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

gbs_course-student.pdf

TUM-GBS-Fol-2013-ALLE-1-145.pdf – TUM-GBS-Fol-2013.ppt [Kompatibilitätsmodus]

Technische Universität München

Einführung

- Fragmentierung
 - Zerstücklung des freien Teils des Adressraums
- Externe Fragmentierung
 - Freibereiche zwischen belegten Bereichen
 - [JS12, Kap. 5, p. 120]
- Interne Fragmentierung
 - Freibereiche innerhalb fester (Rahmen-) Bereiche
 - [JS12, Kap. 5, p. 121]

0 1 GByte 2 GByte 3 GByte 4 GByte

spezieller Adressbereich (Größe nicht proportional)

Fr. 22. Nov. 09:15

Einführung

- Forderungen an (Programm-) Adressraumrealisierung
 - Homogene und zusammenhängende Adressbereiche
 - Unabhängig vom Arbeitsspeicher
 - Erkennen fehlerhafter Zugriffe
 - Erkennen von Überschneidungen (Halde, Keller)
 - Schutz der Anwendungen gegeneinander (insbs. von funktionstüchtigen gegenüber fehlerhaften)
 - Kontrollierbares und kontrolliertes Aufteilen der Speicherressourcen
 - Minimale Fragmentierung

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Speicherabbildungen

- Direkte Adressierung
 - Programmadresse werden direkt als Maschinenadressen interpretiert
 - Nachteile
 - Lücken (Fragmentierung) nach Programmende
 - Forderung: Verschiebbarkeit von Programmen
 - Programmgröße auf AS beschränkt
 - Forderung: Unabhängigkeit von AS-Größe
 - Speicherausnutzung schlecht
 - Forderung: Nutzung der Lokalitätseigenschaften von Programmen

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Speicherabbildungen

- Basisadressierung
 - Maschinenadresse = Basisadresse + Programmadresse
 - Basisadresse ist programmspezifisch
 - Programmadressen starten mit 0
 - Speicherwaltungsstrategien
 - first-fit, next-fit, best-fit, worst-fit *tx*
 - Buddy-Systeme
 - Vergabe in Größen von Zweierpotenzen *Jn*

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Seitenadressierung – Überblick

- Ansatz
- Adressabbildung
- Seiten-Kacheltabelle
- Seitenfehlerbehandlung
- Seitenverwaltungsstrategien

5. Speicherverwaltung

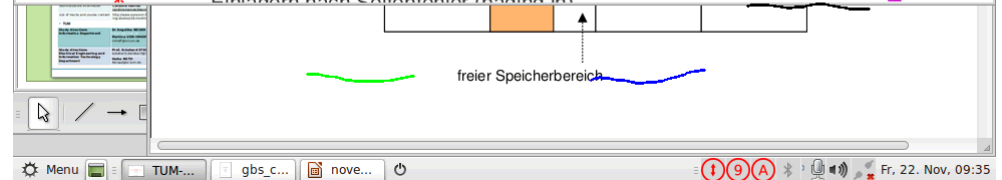
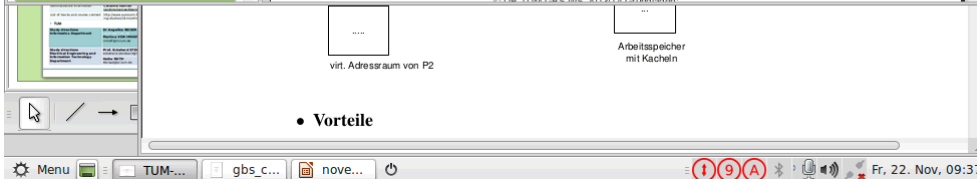
Quelle: [JS12] Kap. 5

Seitenadressierung – Ansatz

- Prozessadressraum
 - Einteilung in Seiten (pages) gleicher Größe
- Maschinenadressraum
 - Einteilung in Kacheln (Seitenrahmen, frames)
 - Gleicher Größe (i.d.R. wie Seitengröße)
- Eigenschaften der Seitenadressierung
 - Seiten gespeichert im AS oder/und HS
 - Bei Nutzung im AS erforderlich
 - Zuordnung über Seitendeskriptoren in Seitentabelle
 - Einlagern nach Seitenfehler (paging in)
 - Auslagern bei Platzbedarf (paging out)
- Abbildung: Virtueller Speicher – Arbeitsspeicher
 - [JS12, Kap. 5, p. 126]

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Seitenadressierung – Adressabbildung

- Virtuelle Adresse
 - $v = (s, w)$
- Reale Adresse
 - $r = (k, w)$
 - [JS12, Kap. 5, p. 127]
- Beispiel
 - mit 16-Bit Adressen, Seitengröße von 4KB
 - [JS12, Kap. 5, p. 128]
- Umsetzung
 - Memory Management Unit (MMU)
 - [JS12, Kap. 5, p. 128]
 - Translation Lookaside Buffer (TLB) für kürzere Zugriffszeit
 - [JS12, Kap. 5, p. 129]

((mit $r = p$ im Skript))

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

TUM-GBS-Fol-2013-ALLE-1-145.pdf — TUM-GBS-Fol-2013.ppt [Kompatibilitätsmodus]

Technische Universität München

Seitenadressierung – Adressabbildung

- Virtuelle Adresse
 - $v = (s, w)$
- Reale Adresse
 - $r = (k, w)$ ((mit $r = p$ im Skript))
 - [JS12, Kap. 5, p. 127]
- Beispiel
 - mit 16-Bit Adressen, Seitengröße von 4KB
 - [JS12, Kap. 5, p. 128]
- Umsetzung

• Vorteile

Fr, 22. Nov, 09:43

TUM-GBS-Fol-2013-ALLE-1-145.pdf — TUM-GBS-Fol-2013.ppt [Kompatibilitätsmodus]

Technische Universität München

Seitenadressierung – Adressabbildung

- Virtuelle Adresse
 - $v = (s, w)$
- Reale Adresse
 - $r = (k, w)$ ((mit $r = p$ im Skript))
 - [JS12, Kap. 5, p. 127]
- Beispiel
 - mit 16-Bit Adressen, Seitengröße von 4KB
 - [JS12, Kap. 5, p. 128]
- Umsetzung
 - Memory Management Unit (MMU)
 - [JS12, Kap. 5, p. 128]
 - Translation Lookaside Buffer (TLB) für kürzere Zugriffszeit
 - [JS12, Kap. 5, p. 129]

Quelle: [JS12] Kap. 5

Beispiel für Adressrechnung

Fr, 22. Nov, 09:46

TUM-GBS-Fol-2013-ALLE-1-145.pdf — TUM-GBS-Fol-2013.ppt [Kompatibilitätsmodus]

Technische Universität München

* Es ist ein differenzierter Zugriffsschutz innerhalb eines Prozesses möglich.

5.4.2 Adressabbildung

Bei der Seitenadressierung erfolgt die Umsetzung der Programmadressen, also die Umsetzung der virtuellen Adressen in Maschinenadressen durch eine Abbildung von Seiten auf Kacheln. Eine virtuelle Adresse v ist gegeben durch $v = (s, w)$, wobei s die Seitennummer und w das Offset in der Seite angibt. Die Adresse v wird abgebildet auf die reale Adresse $p = (k, w)$, wobei k die Kachelnummer angibt, die die Seite enthält.

Beispiel für Adressrechnung

Fr, 22. Nov, 09:46

TUM-GBS-Fol-2013-ALLE-1-145.pdf — TUM-GBS-Fol-2013.ppt [Kompatibilitätsmodus]

Technische Universität München

Seitenadressierung – Adressabbildung

- Virtuelle Adresse
 - $v = (s, w)$
- Reale Adresse
 - $r = (k, w)$ ((mit $r = p$ im Skript))
 - [JS12, Kap. 5, p. 127]
- Beispiel
 - mit 16-Bit Adressen, Seitengröße von 4KB
 - [JS12, Kap. 5, p. 128]
- Umsetzung
 - Memory Management Unit (MMU)
 - [JS12, Kap. 5, p. 128]
 - Translation Lookaside Buffer (TLB) für kürzere Zugriffszeit
 - [JS12, Kap. 5, p. 129]

Quelle: [JS12] Kap. 5

Beispiel für Adressrechnung

Fr, 22. Nov, 09:50

Seitenadressierung – Adressabbildung

- Virtuelle Adresse
 - $v = (s,w)$
- Reale Adresse
 - $r = (k,w)$ ((mit $r = p$ im Skript))
 - [JS12, Kap. 5, p. 127]
- Beispiel
 - mit 16-Bit Adressen, Seitengröße von 4KB
 - [JS12, Kap. 5, p. 128]
- Umsetzung
 - Memory Management Unit (MMU)
 - [JS12, Kap. 5, p. 128]
 - Translation Lookaside Buffer (TLB) für kürzere Zugriffszeit
 - [JS12, Kap. 5, p. 129]

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Memory Management Unit

Die Adressrechnung wird von der Hardware, der *MMU* (Memory Management Unit), durchgeführt.

Meist ist die MMU auf dem CPU-Chip integriert.

Seitenadressierung – Seiten-Kacheltabelle

- Seitendeskriptor
 - Zugriffsrechte (rwx, lesend – schreiben – ausführend)
 - Existenzattribut (e)
 - Geladenattribut (V)
 - Zugriffattribut (r)
 - Verändertattribut (m)
 - Seitenadresse (s als Index)
 - Kacheladresse (ggf.) (Kachel k)
 - Hintergrundspeicheradresse (ggf.) (Block b)
- Große Seiten-Kacheltabellen
 - Mehrstufige Tabellen [JS12, Kap. 5, p. 131]
- Realisierungsvarianten
 - Prozess-spezifisch oder global, indiziert oder assoziativ

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

gbs_course-student.pdf

TUM-GBS-Fol-2013-ALLE-1-145.pdf — TUM-GBS-Fol-2013.ppt [Kompatibilitätsmodus]

Datei Bearbeiten Ansicht Gehe zu Lesezeichen Hilfe

Vorherige Nächste 144 (144 von 145) Auf Seitenbreite einpassen

Vorschaubilder

Technische Universität München **TUM**

Seitenadressierung – Seiten-Kacheltabelle

- Seitendeskriptor
 - Zugriffsrechte (rwx, lesend – schreiben – ausführend)
 - Existent-Attribut (e)
 - Geladen-Attribut (v)
 - Zugriffen-Attribut (r)
 - Verändert-Attribut (m)
 - Seitenadresse (s als Index)
 - Kacheladresse (ggf.) (Kachel k)
 - Hintergrundspeicheradresse (ggf.) (Block b)
- Große Seiten-Kacheltabelle
 - Mehrstufige Tabellen [JS12, Kap. 5, p. 131]
- Realisierungsvarianten
 - Prozess-spezifisch oder global, indiziert oder assoziativ

Quelle: [JS12] Kap. 5

Zugriffszeit
Sei

© UB TUM GBS WS 2013/14 Grundlagen:

Fr, 22. Nov, 09:57

Seitenadressierung – Seiten-Kacheltabelle

- Seitendeskriptor
 - Zugriffsrechte (rwx, lesend – schreiben – ausführend)
 - Existent-Attribut (e)
 - Geladen-Attribut (v)
 - Zugriffen-Attribut (r)
 - Verändert-Attribut (m)
 - Seitenadresse (s als Index)
 - Kacheladresse (ggf.) (Kachel k)
 - Hintergrundspeicheradresse (ggf.) (Block b)
- Große Seiten-Kacheltabelle
 - Mehrstufige Tabellen [JS12, Kap. 5, p. 131]
- Realisierungsvarianten
 - Prozess-spezifisch oder global, indiziert oder assoziativ

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Seitenadressierung – Seitenfehlerbehandlung

- Grundlage: Seitenfehler (page fault)
- Ablauf
 - Beim Zugriff auf eine virtuelle Adresse tritt ein Seitenfehler auf
 - MMU löst Alarm aus
 - BS stellt freie Kachel zur Verfügung
 - Seite wird eingelagert
 - Seitendeskriptor wird aktualisiert
 - Der unterbrochene Befehl wird erneut gestartet
- [JS12, Kap. 5, p. 132]

5. Speicherverwaltung

Quelle: [JS12] Kap. 5

Seitenadressierung – Seitenfehlerbehandlung

- Grundlage: Seitenfehler (page fault)
- Ablauf
 - Beim Zugriff auf eine virtuelle Adresse tritt ein Seitenfehler auf
 - MMU löst Alarm aus
 - BS stellt freie Kachel zur Verfügung
 - Seite wird eingelagert
 - Seitendeskriptor wird aktualisiert
 - Der unterbrochene Befehl wird erneut gestartet
- [JS12, Kap. 5, p. 132]

5. Speicherverwaltung

Quelle: [JS12] Kap. 5