

Kapitel 1: Organisation und Einführung

1.1 Termine

Die Vorlesung findet im Audimax statt, montags von 9:45 bis 11:15 Uhr, mittwochs von 14:00 bis 15:30 Uhr.

1.2 Tutorien

Neben der Vorlesung werden wöchentlich Tutorien angeboten. Registrieren Sie sich bitte hierfür mittels des Registrierungsformulars auf der Webseite "Systemarchitektur Tutorien". Ausgangspunkt für alle Online-dokumente ist unsere Gruppenhomepage:

<http://i30www.ira.uka.de/>

Von dort gelangen Sie über die Navigationshilfen "Teaching" bzw. auf Deutsch "Lehre" zu den jeweiligen Kursseiten des aktuellen WS bzw. auch zu Kursseiten früherer Semester.

In den Tutorien werden die aktuellen Übungsblätter (assignments) besprochen. Aktive Mitarbeit erhöht die Qualität der Tutorien ungemein, nicht umsonst sind diese Tutorien als beste Übungen zu Wahlpflichtveranstaltungen in den vergangenen Semestern ausgezeichnet worden. Wir versuchen, die Übungsblätter rechtzeitig ins Web zu stellen, so dass auch die Teilnehmer an den Mittwochstutorien genügend Einarbeitungszeit in die Problemstellungen haben sollten. Die Tutorien werden für die Studierenden deswegen angeboten, damit sie dort mit ihren Kolleginnen und Kollegen Probleme der Systemarchitektur diskutieren und Lösungen erarbeiten können. Je früher sich die Studierenden hierbei aktiv engagieren, desto besser fallen die Ergebnisse in den Klausuren aus.

Während die in den Tutorien behandelten Übungsaufgaben eher von theoretischer Natur sind, dafür aber inhaltlich mit den Aufgabenstellungen der Klausur übereinstimmen, werden zusätzlich noch praktische Programmierübungen angeboten, die auf dem Lehrbetriebssystem OS/161 der University of Harvard (US) aufbauen.

1.3 Klausuren

Die abschließende Klausur findet am 23. März 2009 statt. Es sind keine Hilfsmittel zugelassen. Maximal können in der Klausur 60 Punkte erreicht werden, der folgende Notenspiegel wird unverändert seit 1990 verwendet:

Note	Punkte
1	50 - 60
1,3	46 - 49,5
1,7	43 - 45,5
2	40 - 42,5
2,3	36 - 39,5
2,7	33 - 35,5
3	30 - 32,5
3,3	26 - 29,5
3,7	23 - 25,5
4	20 - 22,5
5	< 20

Neben dieser Klausur im März wird voraussichtlich in der vorlesungsfreien Zeit nach dem Sommersemester 2009 eine weitere Nachklausur angeboten.

1.4 Bonuspunkte

In den 4 Programmierübungen können bis zu 4 Bonuspunkte bei sehr guten Leistungen erworben werden, die dann bei der Benotung für die Klausur angerechnet werden.

1.5 Forum

Neben den mündlichen Aussprachemöglichkeiten in den Tutorien bieten wir zusätzlich ein Diskussionsforum zur Systemarchitektur an, in dem jede(r) Studierende die Möglichkeit hat, Fragen zu stellen, Anmerkungen zu machen, alternative Vorschläge zu den Musterlösungen zur Diskussion zu stellen. In der Vergangenheit wurde dieses Forum vorzugsweise kurz vor der jeweiligen Klausur genutzt, wir empfehlen jedoch, davon möglichst früh Gebrauch zu machen. Da wir in der Vorlesung den Stoff nicht vollständig und erschöpfend besprechen können, wird trotz ausreichend vorhandenen Literaturquellen das eine oder andere Problem offen bleiben. Wenn Sie dies rechtzeitig im Forum ansprechen, kann ich in der Vorlesung oder in den Übungen noch darauf eingehen.

1.6 Lehrveranstaltungsstil

Sie befinden sich nun im Hauptstudium, Sie gehören somit zur Elite in Deutschland, denn dies ist nur wenigen Ihres Jahrgangs möglich. Spätestens jetzt müssen Sie sich aber auch davon lösen, dass Sie in Ihrem letzten Studienabschnitt alles auf dem Silberteller angeboten bekommen. Im Gegensatz zum Studium in anderen Ländern, das dort erheblich verschulter ist (insbesondere in USA, aber auch in Frankreich), müssen Sie Ihr Hauptstudium selbst planen, wozu Ihnen allerdings die acht Wahlpflichtfächer bei der Auswahl der Vertiefungsfächer helfen sollen. Die Veranstaltung Systemarchitektur zeichnet sich dadurch aus, dass wir sehr viel eigenes Lehrmaterial haben, wenngleich wir kein Skriptum publizieren (wozu auch, wenn es ein so hervorragendes Lehrbuch gibt: Andy Tanenbaum: "Modern Operating Systems" oder auf Deutsch "Moderne Betriebssysteme").

Stattdessen bieten wir diese Vorlesungskommentare (lecture notes) -vorläufig nur in Deutsch- an, haben zusätzlich einen recht umfangreichen Foliensatz (vorläufig nur auf Englisch), der im pps- und im pdf-Format abgelegt wird. Hinzu kommt noch eine Reihe von teilweise animierten Online-Tutorien. In den Übungen und auch in der Klausur wird aber zusätzlich auch noch Information aus bestimmten Originalpublikationen benötigt, auf die ich in der Vorlesung explizit hinweisen werde. Wir erwarten, dass Sie diese Artikel sorgfältig studieren. Dabei werden wir zwischen zwei wesentlichen Publikationsarten unterscheiden: Grundsatzartikel, die einen Meilenstein der Betriebssysteme darstellen, und aktuelle Veröffentlichungen, die voraussichtlich die zukünftige Systementwicklung bestimmen werden. Somit sind wir wieder am Ausgangspunkt dieses Abschnitts angekommen: Hauptdiplomstudium heißt: "Selbständig studieren"!

Anregungen zum selbständigen Studieren sind im Internet zwar rar gesät, gleichwohl hierzu einige Literaturhinweise bzw. (z.Z. nicht ganz ernst gemeinten) Links

Literatur:

Ingrid Herrmann: Richtig Studieren, Handbuch 7.55 €

Stephan Becher: Schnell und erfolgreich studieren: Organisation, Zeitmanagement, Arbeitstechniken 17 €
Taschenbuch

Links:

<http://www.ira.uka.de/> (Studienführer der Fakultät für Informatik Karlsruhe)

<http://i30www.ira.uka.de/teaching/examinations/hints/> (Effizientes Studieren auf der Examenshinweiseite der Gruppe Systemarchitektur)

<http://www.uni-duesseldorf.de/HHU/zsb/Erfolgreich>

<http://www.tu-berlin.de/zuv/asb/studieren/studenten.html>

http://www.phil.uni-erlangen.de/economics/bwl/studium/method/erf_stud/erf_stud.htm
<http://www.marin.cc.ca.us/~don/Study/Hcontents.html>
http://ncnc.essortment.com/successfulstudy_rlry.htm
http://sfinf.fsinf.de/index.php?Example_Session=93317bf0f6826177023b6661a1d38e66

1.7 Meilensteine der Betriebssysteme

Die folgende Liste erhebt keinen Anspruch auf Vollständigkeit und wird gegebenenfalls auch im Lauf der Veranstaltung noch ergänzt werden. Einige der genannten Meilensteine gehen über den Horizont von lokalen Systemen hinaus, auf die wir uns in der Veranstaltung beschränken wollen.

1956	IBM 704 OS	
1968	THE (layer)	Dijkstra (U Eindhoven, Netherlands)
1968	Arpanet	Shapiro, Crocker, Carr, Rulifson, Stoughton (UCLA, NWG)
1969	Multics	Organick, Saltzer et al. (MIT)
1970	RC4000	Brinch Hansen (Kopenhagen, Denmark)
1971	Unix	Ritchie, Thompson (Bell Labs)
1978	VM/370	Seawright, MacKinnon (IBM)
1981	Hydra (OOS)	Wulf (MIT)
1986	Amoeba (DS)	Mullender (CWI; Netherlands)
1989	Mach (μ K)	Rashid (CMU)
1990	WWW	Berners-Lee, Cailliau (Cern, Switzerland)

Es gibt eine Reihe von interessanten Beiträgen zur Historie der Betriebssystementwicklung, siehe u.a. auch:

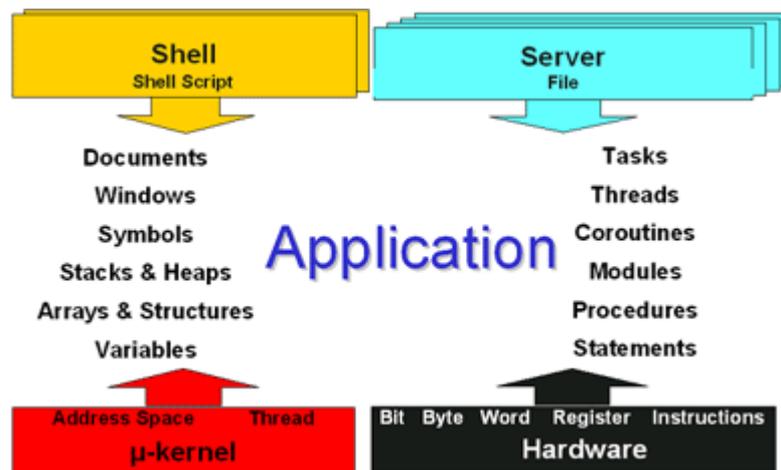
<http://www.netfeatures.de/frames/startfraset.html?../content/historie.html>
<http://www.winhistory.de/index.html>
<http://www.osdata.com/kind/history.htm>
<http://www.cbi.umn.edu/iterations/haigh.html>
http://en.wikipedia.org/wiki/History_of_operating_systems
<http://courses.cs.vt.edu/~cs1104/VirtualMachines/OS.1.html>
http://www.fact-index.com/h/hi/history_of_operating_systems.html
<http://www2.sjsu.edu/faculty/watkins/opsys.htm>
<http://www.osdata.com/>
http://www.campusprogram.com/reference/en/wikipedia/o/op/operating_system.html

Einige dieser Links enthalten nicht nur eine mehr oder weniger vollständige Liste von Betriebssystemen, sondern auch eine Art Klassifikation dieser Systeme.

Ähnlich wie sich Architekten im Rahmen ihrer Universitätsausbildung mit der Baugeschichte auseinandersetzen, kann man auch aus den frühen Ansätzen der Betriebssysteme Lehren ziehen, anstatt das Rad noch mal zu erfinden. Aber an und für sich ist das Studium vergangener Systeme nicht Thema dieser Veranstaltung, es sei denn, wir werden ganz explizit auf ein spezielles System gesondert eingehen. Mit entsprechenden Schlüsselworten bekommt man viele weitere Links zum Thema Historie von Computern, Software etc. geliefert.

1.8 Lehrziele

Am Ende der Veranstaltung sollten Sie in der Lage sein, die passenden Konzepte für ein sorgfältig spezifiziertes System auszuwählen, so dass es die aus Anwendersicht wichtigen Systemziele auch erreicht. Das wichtigste Merkmal für eine formvollendete Systemarchitektur ist erreicht, wenn die Anwendung im Blickpunkt aller Architekturentscheidungen steht und nie vom Architekten außer Acht gelassen wird.



Auch hierzu gibt es mehrere Gegenbeispiele aus der traditionellen Architektur, in der sich Architekten auf Kosten der Anwender mitunter ein sehr fragwürdiges Denkmal gesetzt haben. In unserer Veranstaltung Systemarchitektur geht es also primär um den geeigneten Systementwurf und sekundär auch über potentielle Systemimplementierungen, was dann praktisch im SS in der gleichnamigen Veranstaltung auch in die Praxis umgesetzt werden kann.

1.9 Literatur

Ein Lehrbuch ragt dabei hinsichtlich Originalität, Sprachwitz und bestechender Beispiele deutlich aus der Menge der übrigen heraus, so dass es noch einmal gesondert genannt werden soll:

Tanenbaum, A.: Modern Operating Systems, Prentice Hall, 2007

Weitere allgemeine Textbücher werden hier alphabetisch nach ihren zuerst genannten Autoren aufgeführt, u.a.

Bacon, J.: Operating Systems, Addison-Wesley, 2003
 Davis, W.: Operating Systems: A Systematic View, Addison-Wesley, 2004
 Nehmer, J.: Grundlagen moderner Betriebssysteme, dpunkt, 2001
 Silberschatz, A.: Operating Systems Concepts with Java, John Wiley & Sons, 2007
 Stallings, W.: Operating Systems, Prentice Hall, 2005

Literatur zu speziellen Systemen:

Beck, M.: Linux Kernel Internals, A-W, 1998
 Bovet, D.: Understanding the Linux Kernel, O'Reily, 2000
 Leffler, S.: Design and Implementation of the 4.3BSD Unix OS, Addison-Wesley, 1996
 Mohr, J.: SCO Companion: The essential Guide to Users and System Administrators, Prentice Hall, 1997
 Pham, D.: Multithreaded Programming with Windows NT, P H, 1996
 Solomon, D.: Inside Windows NT, Microsoft Press (3rd ed.), 2001
 Tanenbaum, A.: Operating Systems: Design and Implementation, Minix, P H, 1997
 Vahalia, U.: Unix Internals, The New Frontiers, Prentice Hall, 1996