CENG334

Introduction to Operating Systems

Section 1

What is The Matrix?
(also known as OS among students)

Unfortunately no one can be told what the matrix is.

You have to see it yourself!

Are you ready?

Let’s get started!

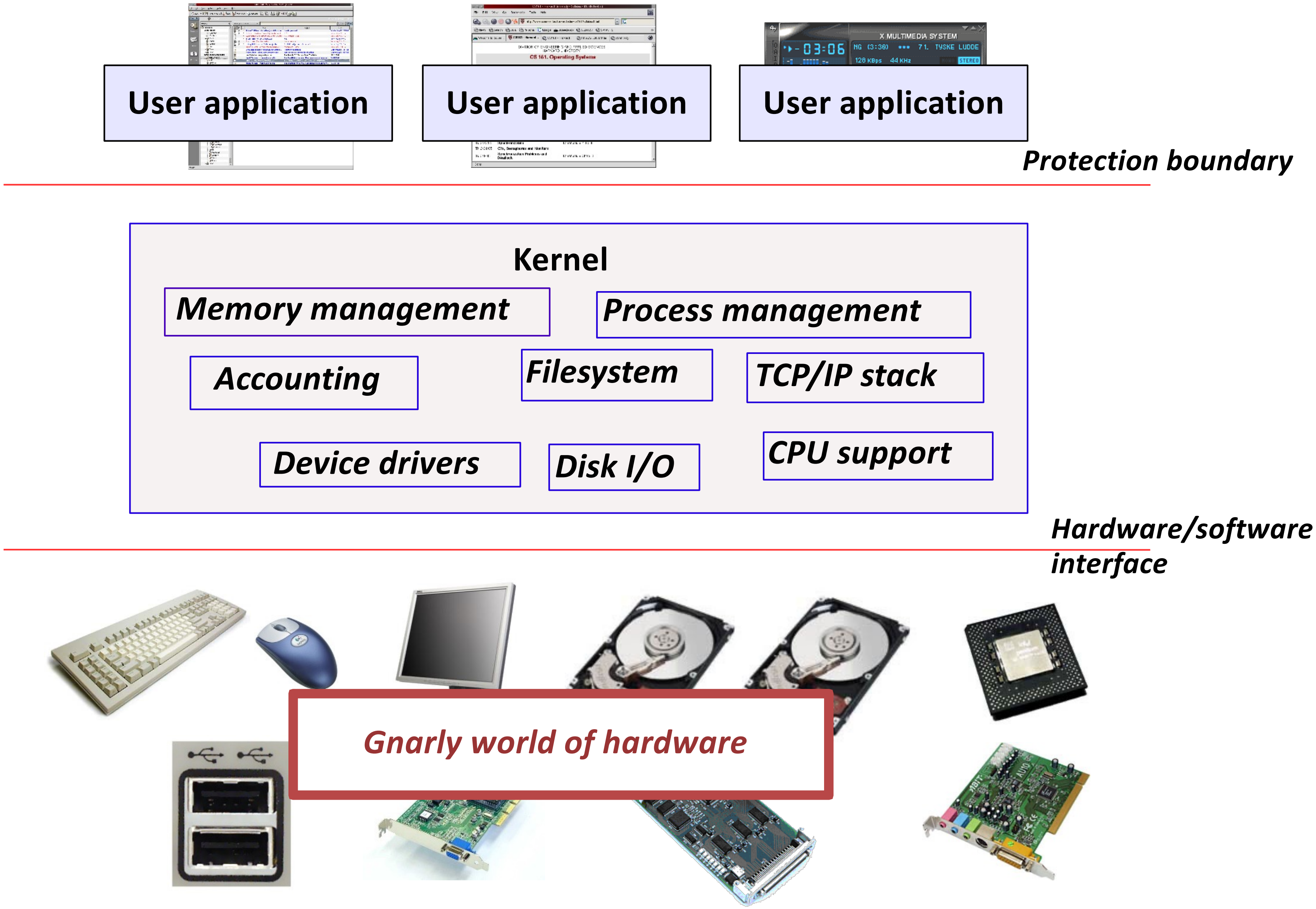
- https://www.youtube.com/watch?v=vKQi3bBA1y8

Welcome to CENG334!

- **What is this course about?**
  - Operating Systems drive the inner workings of virtually every computer in the world today
  - PCs, servers, iPods, cell phones, missile guidance systems, etc. all have an OS that dictate how they operate.
  - The OS manages many aspects of how programs run, and how they interact with hardware and the outside world.
- **Understanding the OS is essential for understanding:**
  - System performance and reliability
  - Resource management
  - Virtualization and abstraction
  - Concurrency and parallelism
  - Hardware interfaces and I/O
- **This course is about more than just “kernel internals”**
  - It is really about learning complex systems design.

What’s an Operating System?

Software that provides an elaborate illusion to applications



User application

User application

User application

Protection boundary

Kernel

Memory management

Process management

Accounting

Filesystem

TCP/IP stack

Device drivers

Disk I/O

CPU support

Hardware/software interface

Gnarly world of hardware

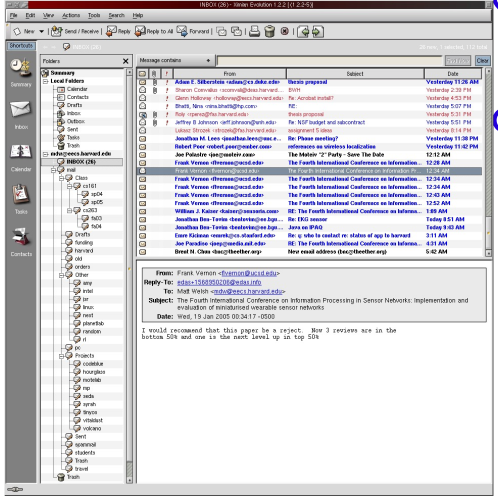
What’s an Operating System?

- **A program that acts as an intermediary between a user of a computer(ized device) and the computer(ized device) hardware.**

- **Abstraction –**
  - Every application thinks it has its own CPU
  - Every application thinks it has access to its memory space
  - Every application thinks it has clean/simple interface to hardware devices
- **Arbitration – Manages access to shared resources.**
  - Which application should be granted CPU now?
  - Which application should be allocated memory?
  - Which application should be granted for hardware access?

One OS Function: Concurrency

Give every application the illusion of having its own CPU!



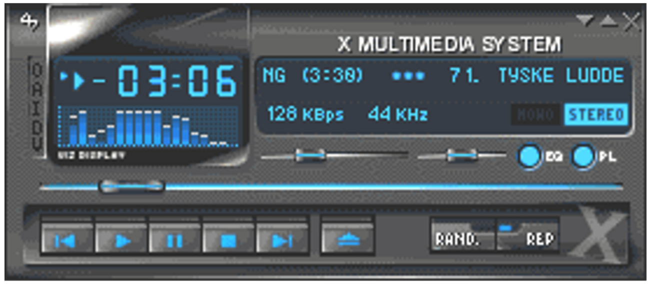
From	Subject	Date
Adam E. Silberstein <adam@cs.duke.edu>	thesis proposal	Yesterday 11:26 AM
Sharon Comvalius <scomvali@deas.harvard....	BWH	Yesterday 2:39 PM
Glenn Holloway <holloway@eecs.harvard.edu>	Re: Acrobat install?	Yesterday 4:53 PM
Bhatti, Nina <nina.bhatti@hp.com>	RE:	Yesterday 5:07 PM
Roly <rperez@fas.harvard.edu>	thesis proposal	Yesterday 5:31 PM
Jeffrey B Johnson <jeff.johnson@unh.edu>	Re: NSF budget and subcontract	Yesterday 5:51 PM
Lukasz Strozek <strozek@fas.harvard.edu>	assignment 5 ideas	Yesterday 8:14 PM
Jonathan M. Lees <jonathan.lees@unc.e...	Re: Phone meeting?	Yesterday 11:38 PM
Robert Poor <robert.poor@ember.com>	references on wireless localization	Yesterday 11:42 PM
Joe Polastre <joe@moteiv.com>	The Moteiv "2" Party - Save The Date	12:12 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information...	12:28 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information Pr...	12:34 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information...	12:34 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information...	12:34 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information...	12:43 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information...	12:52 AM
William J. Kaiser <kaiser@sensoria.com>	RE: The Fourth International Conference on Informa...	1:09 AM
Jonathan Ben-Tovim <bentovim@ee.bgu....	Re: EKG sensor	Today 8:51 AM
Jonathan Ben-Tovim <bentovim@ee.bgu....	Java on IPAQ	Today 9:43 AM
Emre Kiciman <emrek@cs.stanford.edu>	Re: q: who to contact re: status of app to harvard	3:11 AM
Joe Paradiso <joep@media.mit.edu>	RE: The Fourth International Conference on Informa...	4:31 AM
Brent N. Chun <bnc@theether.org>	New email address (bnc@theether.org)	5:42 AM

From: Frank Vernon <flvernon@ucsd.edu>
Reply-To: edas+1568950206@edas.info
To: Matt Welsh <mdw@eecs.harvard.edu>
Subject: The Fourth International Conference on Information Processing in Sensor Networks: Implementation and evaluation of miniaturised wearable sensor networks
Date: Wed, 19 Jan 2005 00:34:17 -0500

```
I would recommend that this paper be a reject.  Now 3 reviews are in the
bottom 50% and one is the next level up in top 50%
```

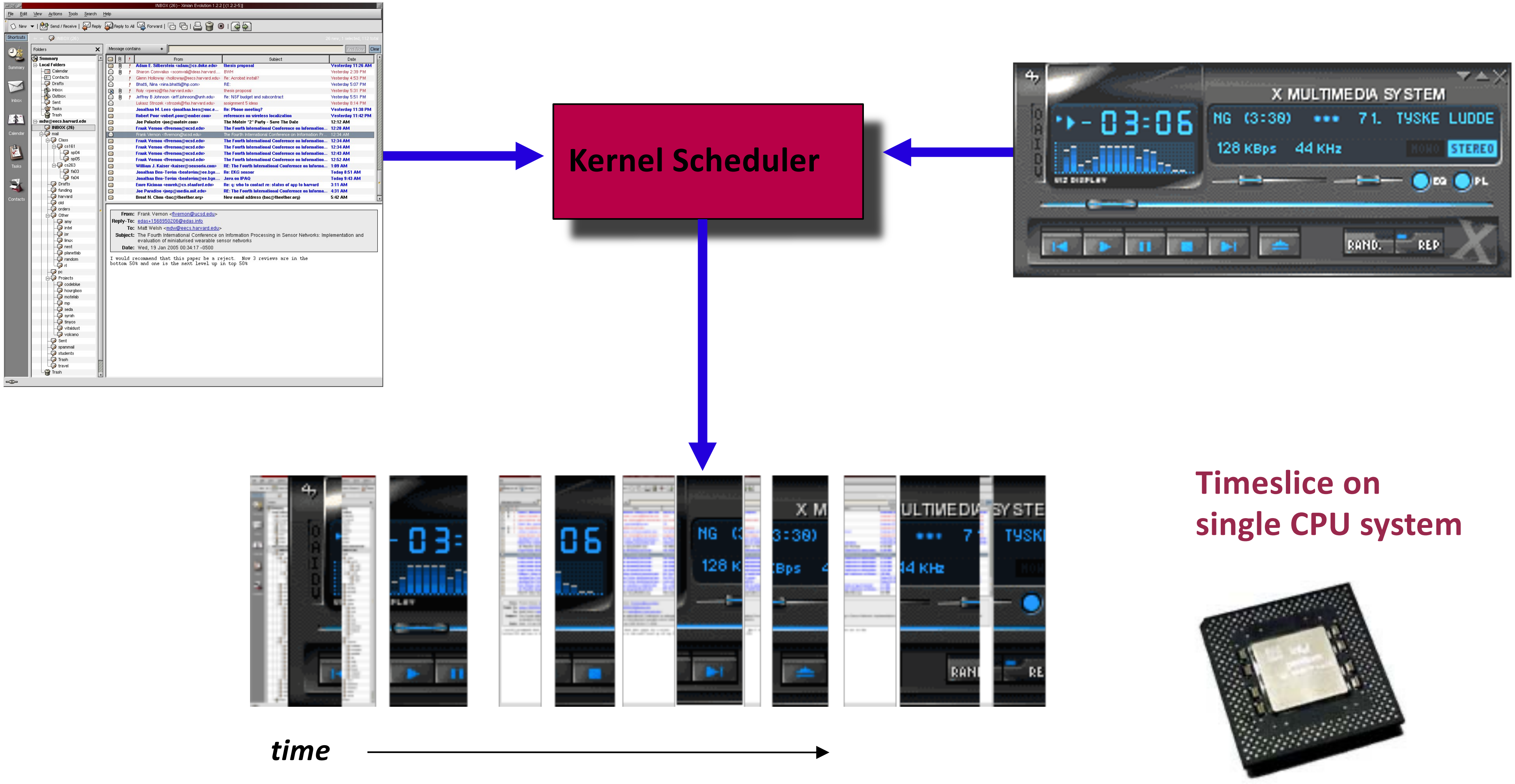
I think I have my own CPU

So do I!



One OS Function: Concurrency

- **The OS *timeslices* each application on a single CPU**
  - Switches between applications extremely rapidly, i.e., 100 times/sec



INBOX (26) - Ximian Evolution 1.2.2 [(1.2.2-5)]

26 new, 1 selected, 112 total

From	Subject	Date
Adam E. Silberstein <adam@cs.duke.edu>	thesis proposal	Yesterday 11:26 AM
Sharon Comvalius <scomvali@deas.harvard....	BWH	Yesterday 2:39 PM
Glenn Holloway <holloway@eecs.harvard.edu>	Re: Acrobat install?	Yesterday 4:53 PM
Bhatti, Nina <nina.bhatti@hp.com>	RE:	Yesterday 5:07 PM
Roly <rperez@fas.harvard.edu>	thesis proposal	Yesterday 5:31 PM
Jeffrey B Johnson <jeff.johnson@unh.edu>	Re: NSF budget and subcontract	Yesterday 5:51 PM
Lukasz Strozek <strozek@fas.harvard.edu>	assignment 5 ideas	Yesterday 8:14 PM
Jonathan M. Lees <jonathan.lees@unc.e...	Re: Phone meeting?	Yesterday 11:38 PM
Robert Poor <robert.poor@ember.com>	references on wireless localization	Yesterday 11:42 PM
Joe Polastre <joe@moteiv.com>	The Moteiv "2" Party - Save The Date	12:12 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information...	12:28 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information Pr...	12:34 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information...	12:34 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information...	12:34 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information...	12:43 AM
Frank Vernon <flvernon@ucsd.edu>	The Fourth International Conference on Information...	12:52 AM
William J. Kaiser <kaiser@sensoria.com>	RE: The Fourth International Conference on Informa...	1:09 AM
Jonathan Ben-Tovim <bentovim@ee.bgu....	Re: EKG sensor	Today 8:51 AM
Jonathan Ben-Tovim <bentovim@ee.bgu....	Java on IPAQ	Today 9:43 AM
Emre Kiciman <emrek@cs.stanford.edu>	Re: q: who to contact re: status of app to harvard	3:11 AM
Joe Paradiso <joep@media.mit.edu>	RE: The Fourth International Conference on Informa...	4:31 AM
Brent N. Chun <bnc@theether.org>	New email address (bnc@theether.org)	5:42 AM

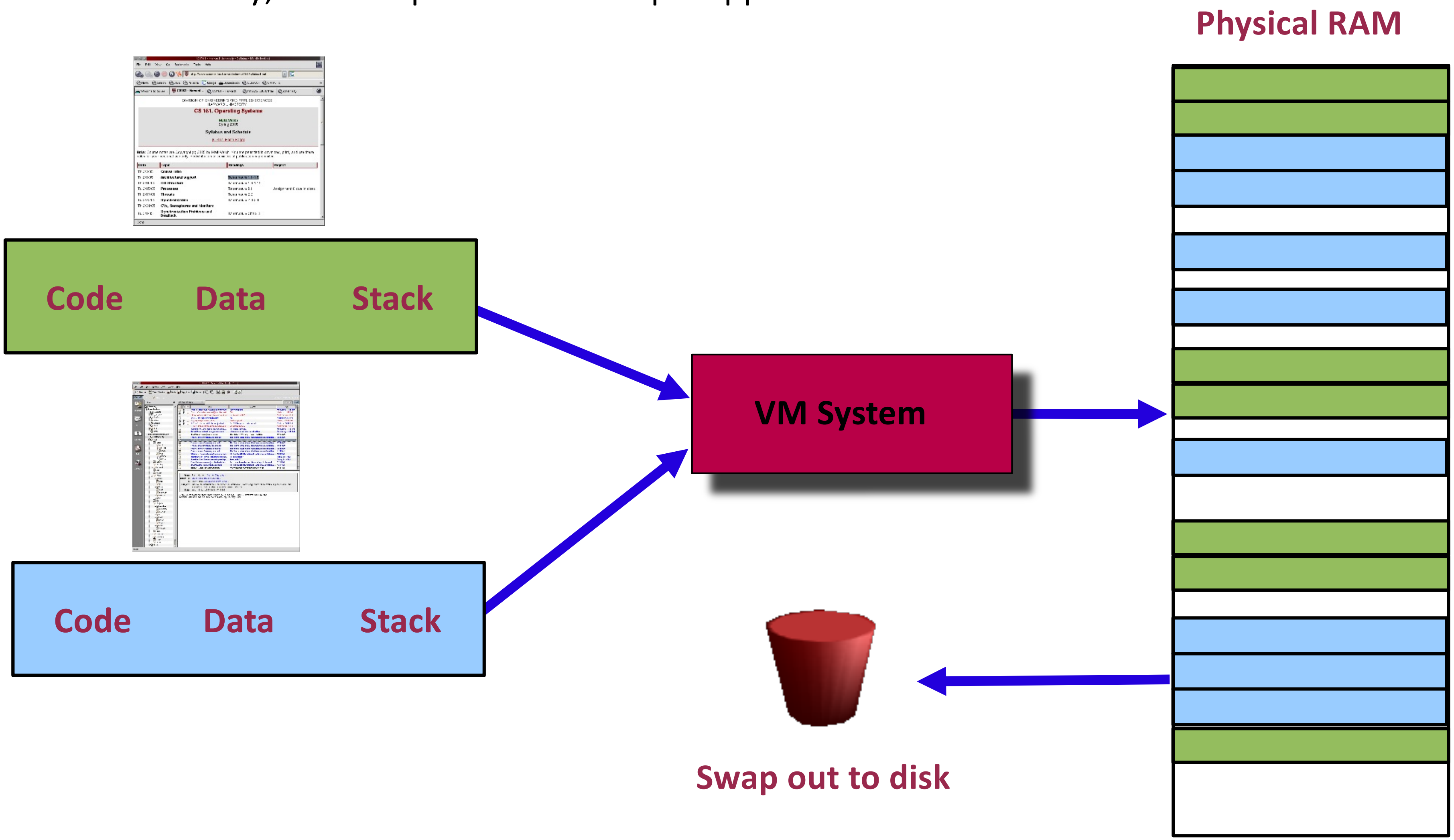
From: Frank Vernon <flvernon@ucsd.edu>
Reply-To: edas+1568950206@edas.info
To: Matt Welsh <mdw@eecs.harvard.edu>
Subject: The Fourth International Conference on Information Processing in Sensor Networks: Implementation and evaluation of miniaturised wearable sensor networks
Date: Wed, 19 Jan 2005 00:34:17 -0500

```
I would recommend that this paper be a reject.  Now 3 reviews are in the
bottom 50% and one is the next level up in top 50%
```

Timeslice on single CPU system

Another OS Function: Virtual Memory

- **Give every application the illusion of having infinite memory**
  - And, that it can access any memory address it likes!
  - In reality, RAM is split across multiple applications



More OS Functions

- **Multiprocessor support**
  - Modern systems have multiple CPUs
  - Can run multiple applications (or *threads* within applications) in parallel
  - OS must ensure that memory and cache contents are consistent across CPUs
- **Filesystems**
  - Real disks have a hairy, sector-based access model
  - User applications see flat files arranged in a hierarchical namespace
- **Network protocols**
  - Network interface hardware operates on the level of unreliable packets
  - User apps see a (potentially reliable) byte-stream *socket*
- **Security and protection**
  - Prevent multiple apps from interfering with each other and with normal system operation

Why bother with an OS?

- **Not just to give Slashdot readers something to argue about...**
- **Abstract away messy details of hardware**
  - Give apps a nice clean view of the system
  - Save programmers a lot of trouble when building applications
  - Allow apps to be ported across a wide range of hardware platforms
- **Safety!**
  - Don't let applications run amok – keep them in a “sandbox”
  - e.g., Access to unallocated memory address crashes only the program, not the whole system
    - `Segmentation fault – core dumped`
- **Efficiency**
  - Share one machine across many different apps: concurrent execution
  - You would be surprised how much slack there is in a typical computer system

Why study operating systems?

- **Most people will never write one from scratch...**
    - (Though if you do, you stand to get incredibly rich)
  - Although more and more people are hacking them (e.g., Linux and BSD)
  - You need to understand the “big picture” in order to hack the details
- **This class is about much more than the kernel!**
  - Data structures, concurrency, performance, resource management, synchronization, networks, distributed systems, databases ...
  - The ideas and skills you pick up in this class have broad applications
    - And it doesn't hurt for those Microsoft interviews either
- **This course is the basis for future work in other areas of systems**
  - Distributed systems, P2P, sensor nets, etc.

Major OS Design Issues

- **Structure**
  - How is the OS itself organized? Lots of modules? One big blob of code?
- **Sharing**
  - How are limited resources multiplexed across users?
- **Naming**
  - How to programs and users name and access resources?
- **Security**
  - How to prevent malicious users from compromising the system?
- **Performance**
  - How to keep it all fast?
- **Reliability**
  - What happens when a program (or the OS itself) has a bug or failure?

More OS Design Issues

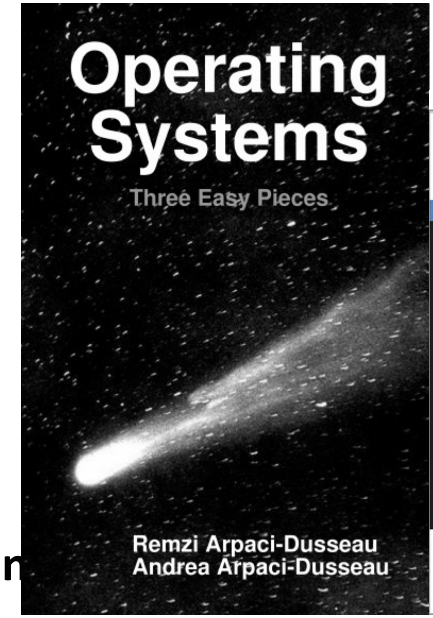
- **Extensibility**
  - How do users and developers add new features?
- **Communication**
  - How do programs exchange information?
- **Concurrency**
  - How are multiple concurrent activities created and controlled?
- **Scale**
  - What happens when demands on resources increase?
- **Distribution**
  - How do many computers interact with each other, e.g., to pool resources?
- **Accounting**
  - How do you track (and maybe charge for) resource usage?

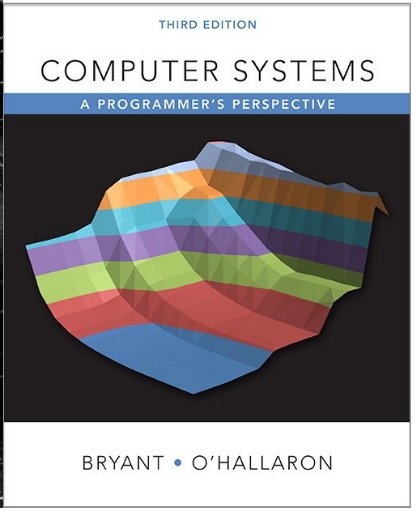
Teaching staff

- **Instructors:**
  - Section 1: Dr. Erol Şahin (A-207)
    - E-mail: erol@ceng
  - Section 2: Dr. Onur Tolga Şehitoğlu (B-209)
    - E-mail: onur@ceng
- **Section 1 lectures (will be at YPA (Yüksel Proje Amfisi))**
  - Thursdays: 11:40-13:00 @ **YPA-2**
  - Fridays: 13:40-15:00 @ **YPA-3**
  - 2+2 lecture time allocated
- **Section 2 lectures:**
  - Mondays: 15:40-17:20 @ **BMB1**
  - Wednesdays: 12:40-13:20 @ **BMB1**
- **Teaching assistants:**
  - Cagri Utku Apak (A-205)
    - E-mail: cakpak@ceng
  - Deniz Sayin (A-409) ****Will be unreachable till mid April****
    - E-mail: sayin@ceng

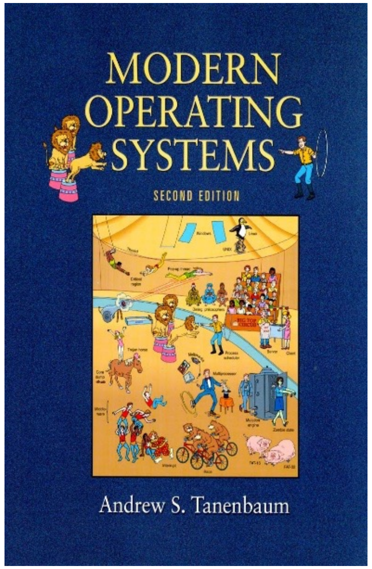
Textbook

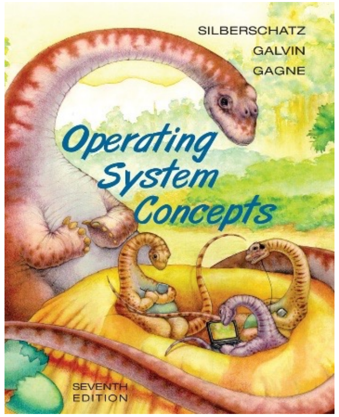
- Operating Systems: Three Easy Pieces by Arpaci-Dusseau and Arpaci-Dusseau
  - https://pages.cs.wisc.edu/~remzi/OSTEP/





- The Little Book of Semaphores by Downey (Green Tea Press)
  - https://greenteapress.com/wp/semaphores/





- Computer Systems: A Programmer's Perspective by Bryant and O'Hallaron.

- Operating System Concepts by Silberschatz, Galvin & Gagne (Wiley)
- Modern Operating Systems by Tanenbaum (Prentice Hall)

The content of the lectures (slides) will define the coverage.

Contents of the books are similar, but the presentation may differ.

Pick at your own taste, and follow the lectures.

Videos and other resources

- **Recorded lectures from 2020 [starting from 3rd week]**
  - http://tiny.cc/o86puz
- **Recorded lectures from 2021**
  - Needs editing – to remove student participations due to KVKK
    - Any volunteers?

Grading

- 21%: Midterm I
- 21%: Midterm II
- 23%: Final Exam.
- 30%: Assignments.
- 5%: Pop quiz

Dates of the exams and assignments will be decided and announced later together with the instructors of other 3rd year courses to minimize overlap and conflict.

- Late submission penalty = 5d^2 , where d is number of days of late submission.
- Students need to collect 25/100 in average from programming assignments announced before final exam (typically first two assignments). Otherwise student will not be qualified for final and graded as NA.

Assignments - To be decided

- Assignment 1: Pipes ?
- Assignment 2: Basic Threading?
- Assignment 3: File System?

Policies

- **Late submission penalty : $5d^2$**
  - d: the number of days of late submission
  - d=1 => 5% penalty
  - d=2 => 20% penalty
  - d=3 => 45% penalty
  - d=4 => 80% penalty
  - d>=5 => 100% penalty
- **Academic dishonesty:**
  - **All assignments submitted should be fully your own.**
  - Your work will be regularly checked for such misconduct and any such attempts will be prosecuted:
  - **Zero tolerance policy on cheating and plagiarism**.
  - At all times you have the right to challenge our decisions on cheating, upon which the case will be processed through the disciplinary channels of the university. However, we would like to remind you that, if found guilty, the legal code of the university proposes a minimum of six month expulsion from the university.

Cheating

- **What is cheating?**
  - Sharing code: either by copying, retyping, looking at, or supplying a copy of a file.
- **What is NOT cheating?**
  - Helping others use systems or tools.
  - Helping others with high-level design issues.
  - Helping others debug their code.

Cheating: Caught and punished

- **Friend**
  - Whole code
    - Easy to detect!
  - Partial code/ “close collaboration”
    - Walk on a thin line
- **Friend who submitted a similar homework in the previous years**
  - Caught, since all previous homework submissions are archived
- **Internet**
  - Whole code
  - Partial code

Web page/communication

- **Course web page:**
  - https://odtuclass.metu.edu.tr
    - All Sections: General announcements and resources for both sections
    - Section 1: Specific announcements for Section 1
- **Official announcements and discussions:**
  - https://cow.ceng.metu.edu.tr/c/courses-undergrad/ceng334/
- **Questions that are general or related to content should be posted to the newsgroup.**
- **Course conduct related questions should be posted to the instructor and/or teaching assistants in private. Make sure you include "CENG334" in the subject line**

Course conduct - Sections

- **Same midterms**
- **Same assignments**
- **Different quizzes**
  - Pick a section and attend it
- **No need to change sections**
  - Registering to one section, but attending to another is fine..
  - Just be consistent..

Web surfing homework for next lecture..

- Learn
- More about XEROX PARC
  - What have they invented?
- More about Ken Thomson and Dennis Ritchie
  - What are they known for?
- More about Microsoft
  - How did MS-DOS become so successful?
- More about Apple
  - What’s the relation between XEROX PARC GUI and Apple GUI?
- What’s Slashdot?
- Use Wikipedia, and google the web..

Movie suggestion

Quiz policy – Section 1

- **Questions about the current lecture**
    - To check that you’ve understood the course content in that lecture
- **First quiz will remain open until 13:50!**