Operating Systems

Isfahan University of Technology
Electrical and Computer Engineering Department

Zeinab Zali

Session1: Meet OS class

Instructor info

Zeinab Zali

zali@iut.ac.ir

RESEARCH INTERESTS

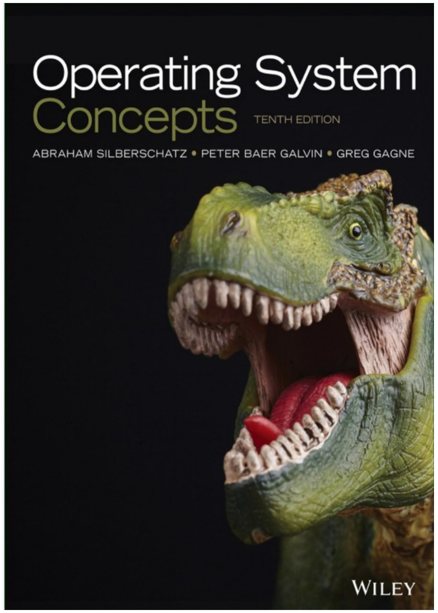
- Computer Networks (NFV, SDN, 5G)
- Cloud Computing Infrastructures and Big Data processing
- IoT, Smart home/city/factory
- Operating Systems, Distributed Systems

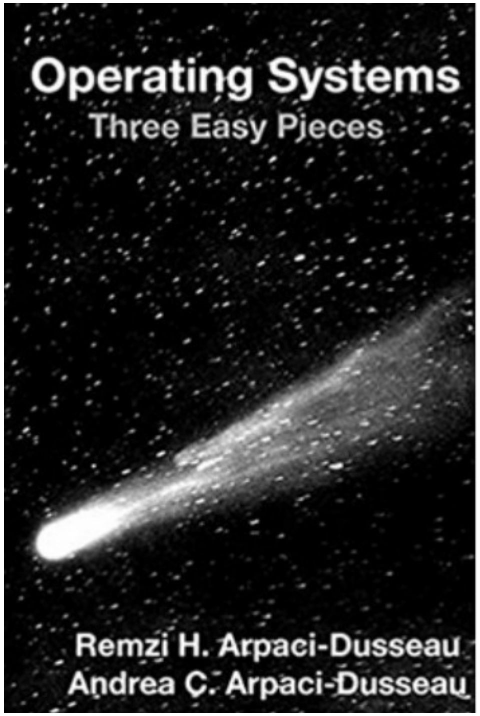
Text books

(Main) Operating Systems Concepts,

A. Silberschatz, P.B. Galvin, G. Gange,

10th edition, 2018





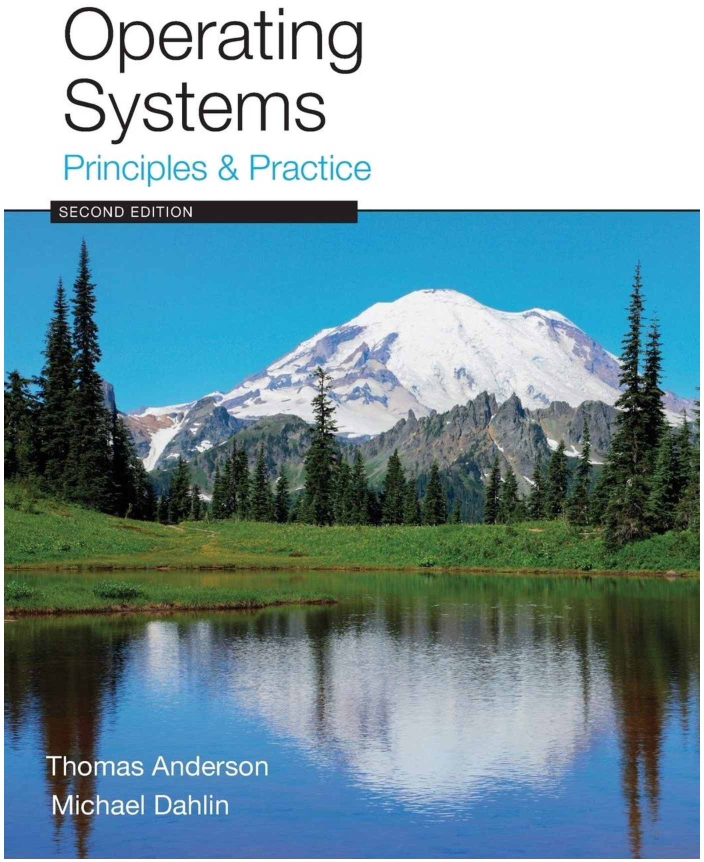
Operating systems: Three easy pieces,

Arpaci-Dusseau, Remzi H and

ArpaciDusseau, Andrea C, 2018

Text books

Operating systems: Principles and practice, volume 1: kernels and processes, second edition, 2015



Plan

	Subject	**References**	**HW/ Project**
1	OS Introduction	Easy ch2 Dinosaur ch1	
2	OS Structure	Easy ch2 Dinosaur ch2	HW1
3	Process	Dinosaur ch3: concepts Dinosaur ch3: API	
4	process	Dinosaur ch3: API, IPC Dinosaur ch3: IPC	
5	thread	Dinosaur ch4	HW2
6	CPU scheduling	Dinosaur ch5: concepts Dinosaur ch5: basic algorithms	
7	CPU scheduling synchronization	Dinosaur ch5: basic algorithms Dinosaur ch5: Linux scheduling Dinosaur ch6: problem statement	HW3
8	synchronization	Dinosaur ch6, Easy ch28: basic methods Dinosaur ch6, Easy ch28-29: locks	

Plan

	Subject	References	HW/ projetcs
9	synchronization	Dinosaur ch6, Easy ch31: semaphore Dinosaur ch6,Easy ch30: condition variables, monitors	
10	synchronization	Dinosaur ch7,Easy ch30: Bounded Buffer Dinosaur ch7,Easy ch31: reader/writer, dining philosopher	HW4
11	Deadlocks	Dinosaur ch8: problem statement and concepts Dinosaur ch8: prevention	
12	Deadlocks Main Memory	Dinosaur ch8: Avoidance and detection Dinosaur ch9: concepts and Address space	HW5
13	Main Memory	Dinosaur ch9: paging, segmentation Dinosaur ch9: swapping, Examples	
14	Virtual Memory	Dinosaur ch10: basics, demand paging Dinosaur ch10: Page replacement	HW6
15	Storage Management		

Grading

😊	Class appearances Open questions	1
😃	Homework	3
😟	Quiz	2
😞	Midterm	7 or 8
	Final	8 or 7

Deadlines headache

BALDWIN
2 X 2 =
3 X 2 =
4 X 2 =

"I'm not late. Everyone learns at their own speed."

How do we enjoy this course

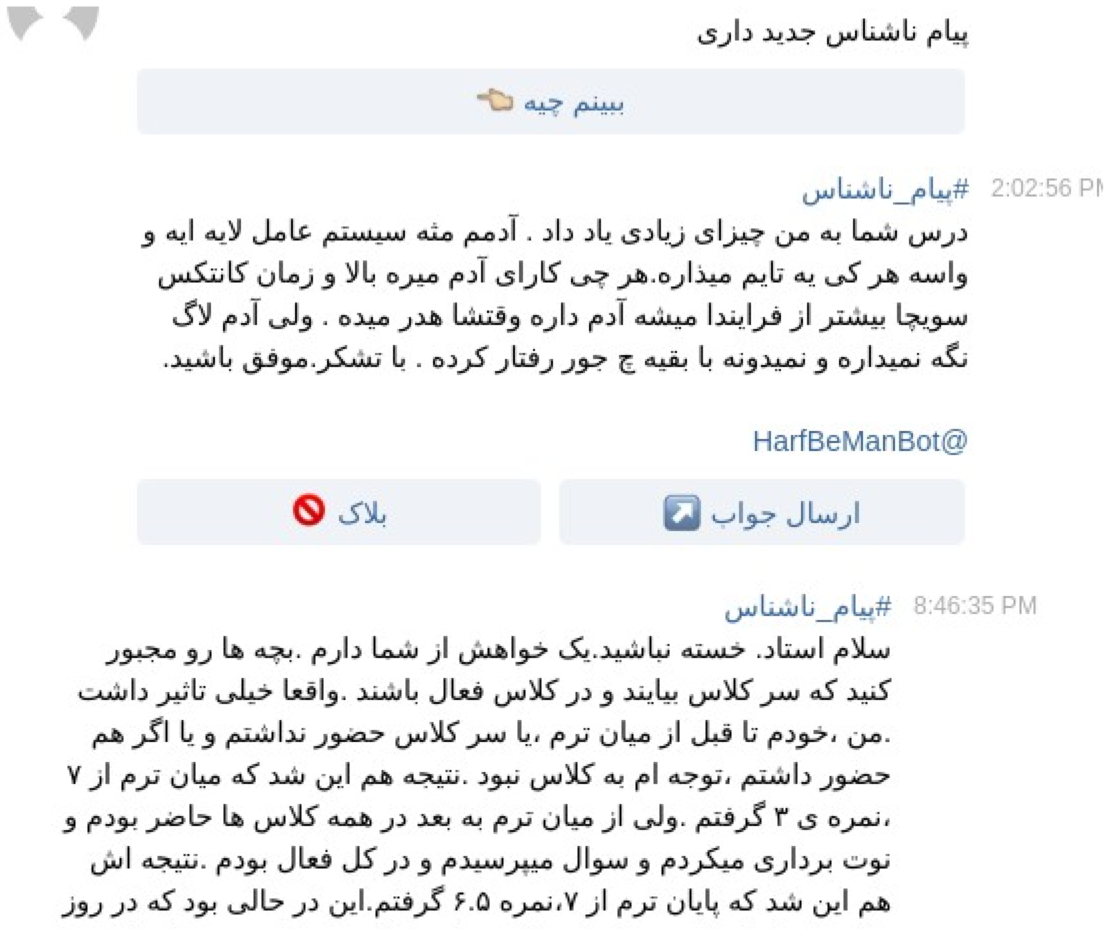
Deadlines

- ✓ For all of homeworks, totally late submission of 10 days is accepted without penalty.
- ✓ Please keep in touch with the course concepts and instructor!
  - ✓ Listening , Studying, programming, Searching, visiting my office, visiting the TAs

How do we enjoy this course

- **Interactive classes:**
  - OS is itself nice, delightful and pleasant :)
  - You can talk about the course concepts in new applications and OSs (sometimes proposed by myself)
  - some breaks during session time for your talk (about anything related to university!)

What your friends says about the course!:)



پیام ناشناس جدید داری

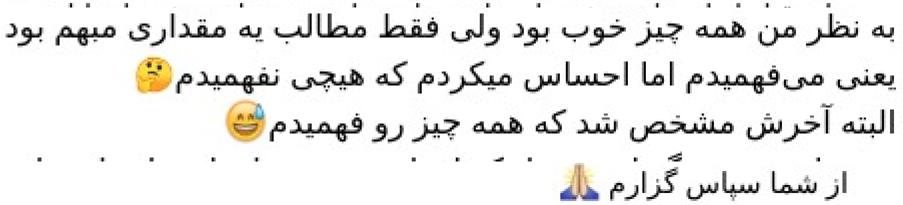
#پیام_ناشناس 2:02:56 PM

درس شما به من چیزای زیادی یاد داد . آدمم مثه سیستم عامل لایه ایه و
واسه هر کی یه تایم میذاره.هر چی کارای آدم میره بالا و زمان کانتکس
سویچا بیشتر از فرایندا میشه آدم داره وقتشا هدر میده . ولی آدم لاگ
نگه نمیداره و نمیدونه با بقیه چ جور رفتار کرده . با تشکر.موفق باشید.

@HarfBeManBot

#پیام_ناشناس 8:46:35 PM

سلام استاد. خسته نباشید.یک خواهش از شما دارم .بچه ها رو مجبور
کنید که سر کلاس بیایند و در کلاس فعال باشند .واقعا خیلی تاثیر داشت
.من ،خودم تا قبل از میان ترم ،یا سر کلاس حضور نداشتم و یا اگر هم
حضور داشتم ،توجه ام به کلاس نبود .نتیجه هم این شد که میان ترم از ۷
،نمره ی ۳ گرفتم .ولی از میان ترم به بعد در همه کلاس ها حاضر بودم و
نوت برداری میکردم و سوال میپرسیدم و در کل فعال بودم .نتیجه اش
هم این شد که پایان ترم از ۷،نمره ۶.۵ گرفتم.این در حالی بود که در روز



به نظر من همه چیز خوب بود ولی فقط مطالب یه مقداری مبهم بود
یعنی می‌فهمیدم اما احساس میکردم که هیچی نفهمیدم🤔
البته آخرش مشخص شد که همه چیز رو فهمیدم😅

از شما سپاس گزارم 🙏

What do you expect from this course?

© CanStockPhoto.com - csp54928712

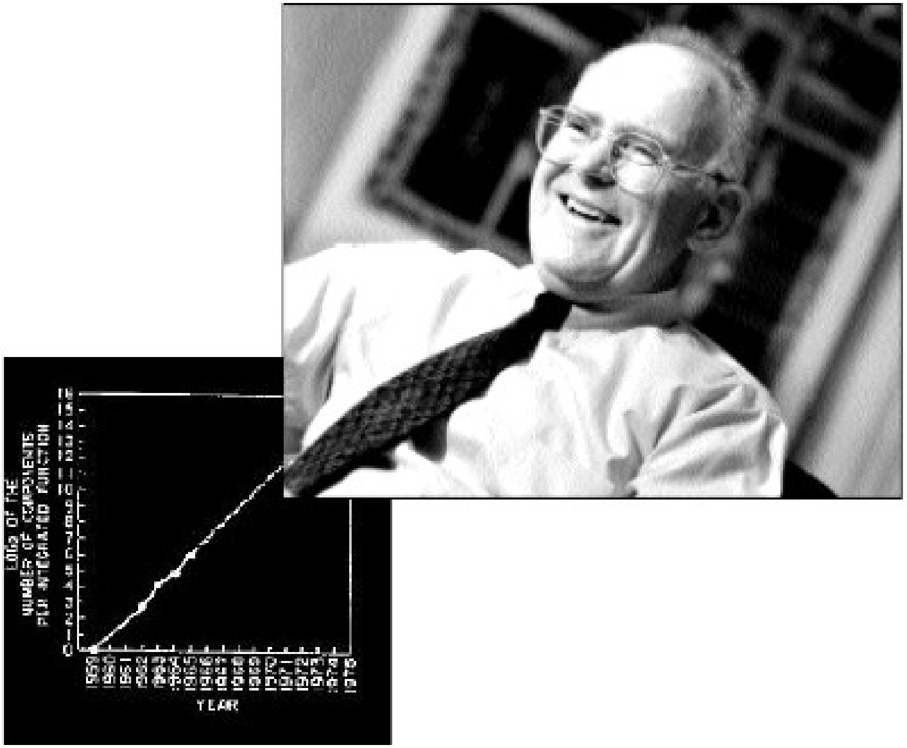
- operating systems concepts are among the most complex in computer science.
  - A modern general purpose OS can exceed 50 million lines of code
  - New operating systems are being written all the time: if you use an e-book reader, tablet, or smartphone, an operating system is managing your device.

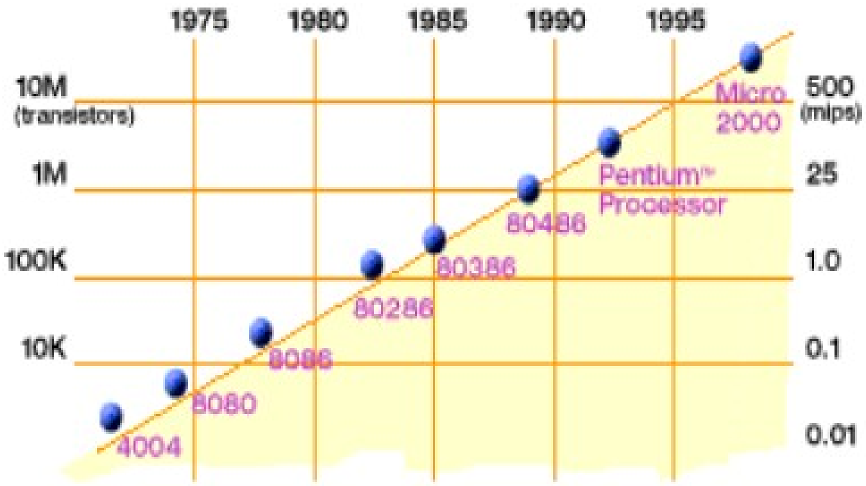
- operating systems concepts are also among the most accessible in computer science
  - Many topics in this course will seem familiar to you

    (Peaceful Co-Existence, Fairness , Traffic rules, ...)
  - Software engineers use many of the same technologies and design patterns as those used in operating systems to build other complex systems

What makes OS exciting and challenging

Technology Trends: Moore's Law





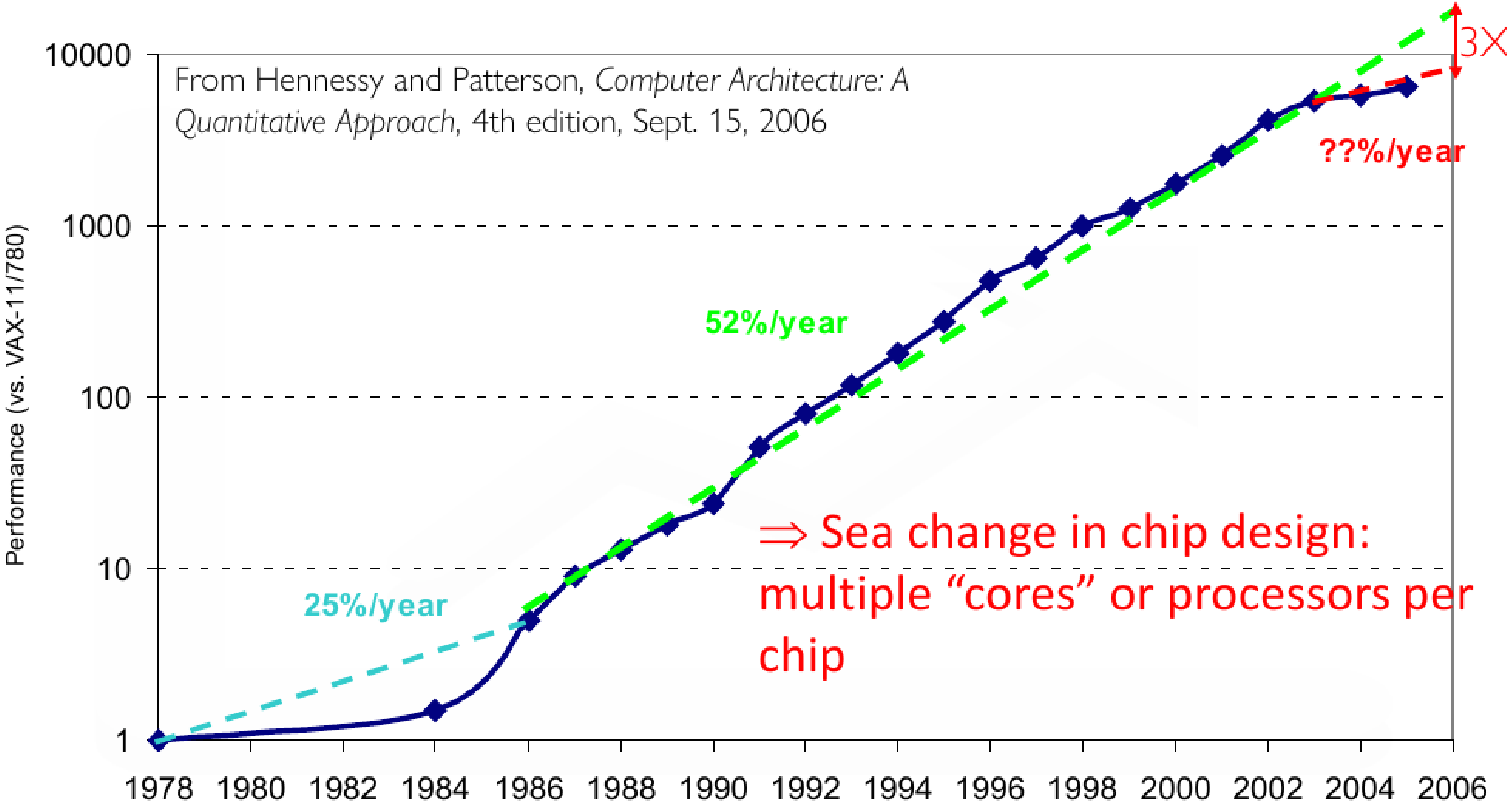
2X transistors/Chip Every 1.5 years

Called "Moore's Law"

Gordon Moore (co-founder of Intel) predicted in 1965 that the transistor density of semiconductor chips would double roughly every 18 months

Microprocessors have become smaller, denser, and more powerful

The End of moore's law

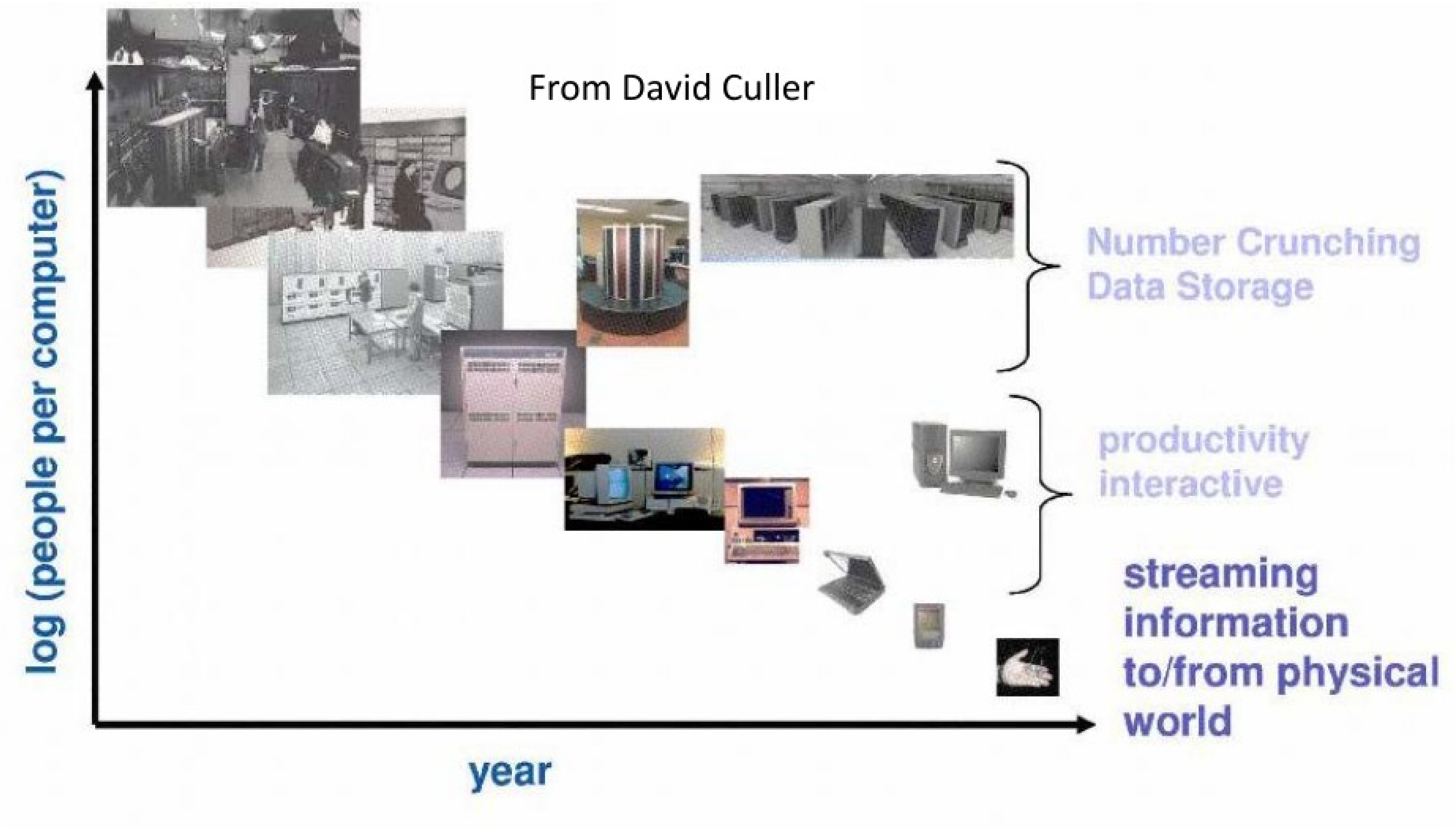


From Hennessy and Patterson, *Computer Architecture: A Quantitative Approach*, 4th edition, Sept. 15, 2006

⇒ Sea change in chip design: multiple "cores" or processors per chip

- VAX : 25%/year 1978 to 1986
- RISC + x86 : 52%/year 1986 to 2002
- RISC + x86 : ??%/year 2002 to present

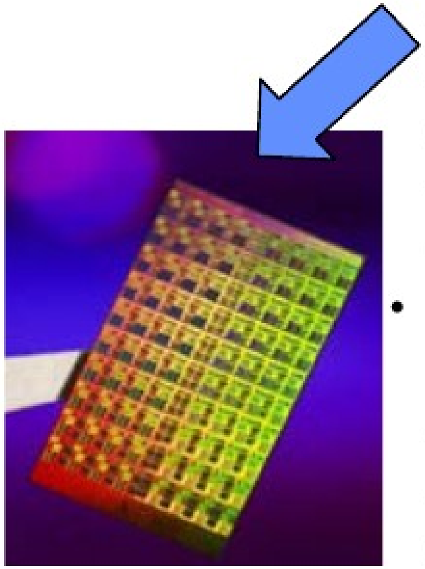
People-to-Computer Ratio Over Time


From David Culler

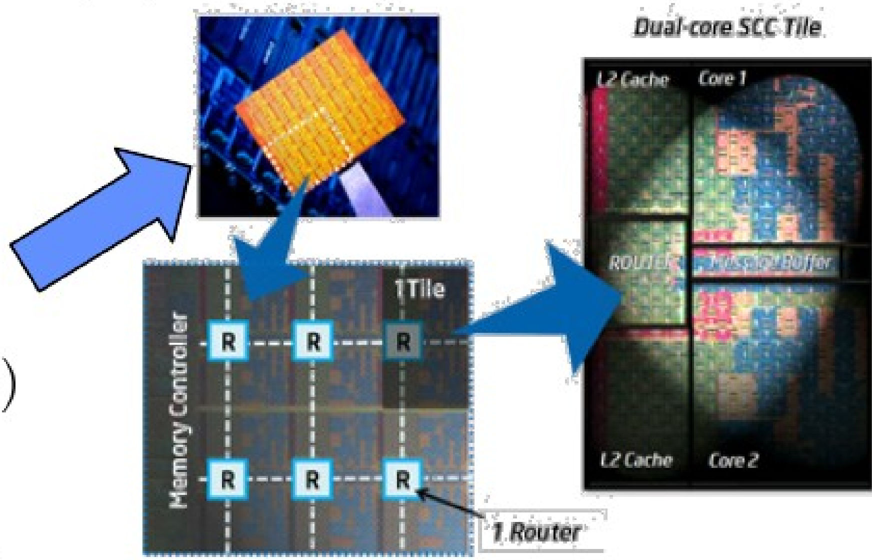
- Today: multiple CPUs/person!
  - Approaching 100s?

ManyCore Chips: The future is here

- Intel 80-core multicore chip (Feb 2007)
  - 80 simple cores
  - Two FP-engines / core
  - Mesh-like network
  - 100 million transistors
  - 65nm feature size



- Intel Single-Chip Cloud Computer (August 2010)
  - 24 "tiles" with two cores/tile
  - 24-router mesh network
  - 4 DDR3 memory controllers
  - Hardware support for message-passing



- Amazon X1 instances (2016)
  - 128 virtual cores, 2 TB RAM
- How to program these?
  - Use 2 CPUs for video/audio
  - Use 1 for word processor, 1 for browser
  - 76 for virus checking???
- Parallelism must be exploited at all levels

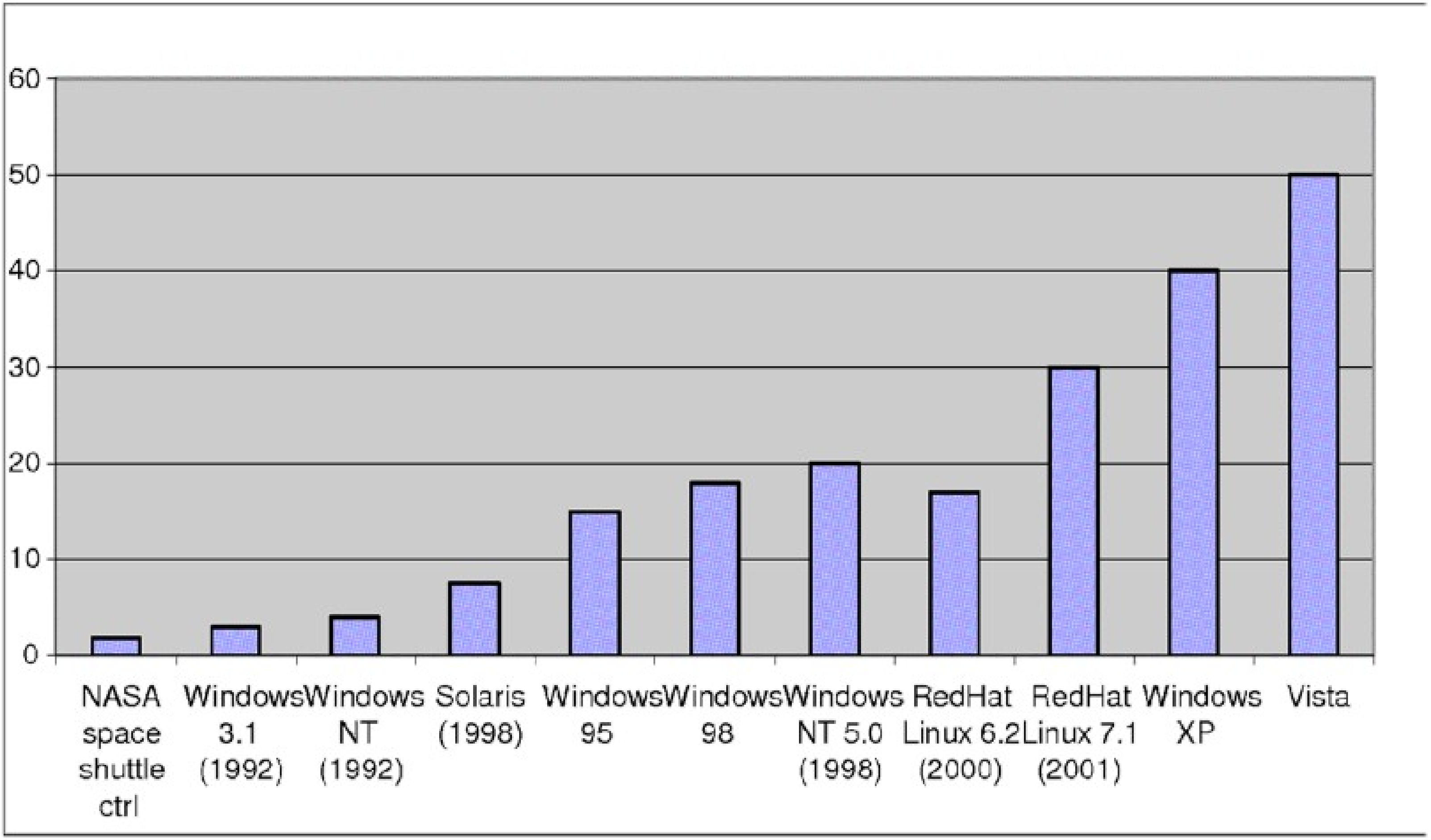
Challenge: Complexity

- Applications consisting of…
  - … a variety of software modules that …
  - … run on a variety of devices (machines) that
    - » … implement different hardware architectures
    - » … run competing applications
    - » … fail in unexpected ways
    - » … can be under a variety of attacks

- Not feasible to test software for all possible environments and combinations of components and devices
  - The question is not whether there are bugs but how serious are the bugs!

Increasing Software Complexity

Millions of lines of source code

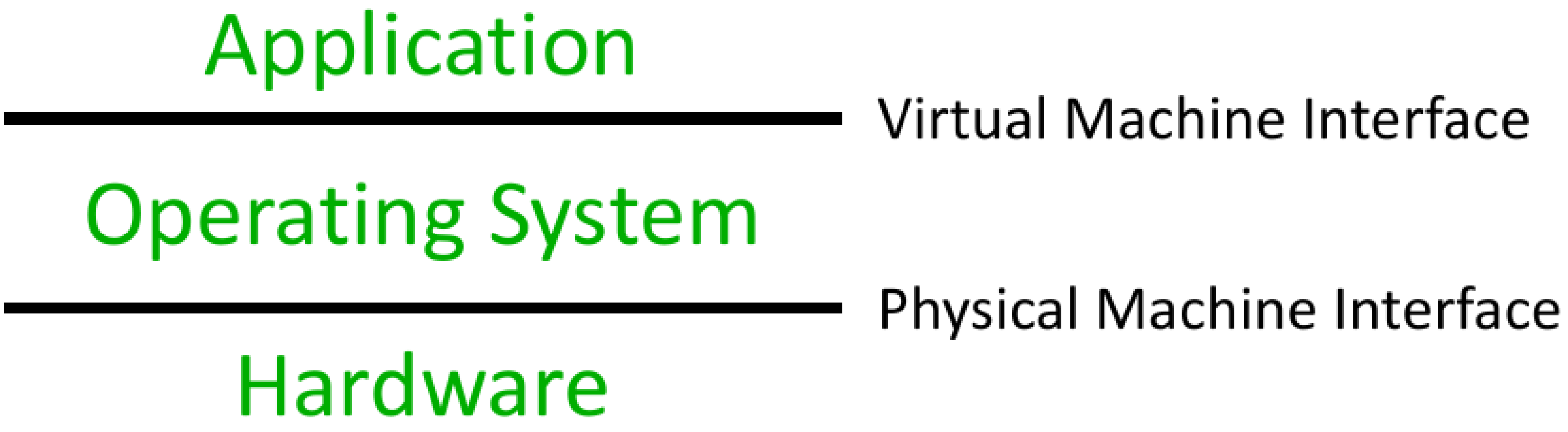


From MIT's 6.033 course

How do we tame complexity?

- Every piece of computer hardware different
  - Different CPU
    - » Pentium, PowerPC, ColdFire, ARM, MIPS
  - Different amounts of memory, disk, …
  - Different types of devices
    - » Mice, Keyboards, Sensors, Cameras, Fingerprint readers
  - Different networking environment
    - » Cable, DSL, Wireless, Firewalls,…
- Questions:
  - Does the programmer need to write a single program that performs many independent activities?
  - Does every program have to be altered for every piece of hardware?
  - Does a faulty program crash everything?
  - Does every program have access to all hardware?

OS Tool: Virtual Machine Abstraction



Application

Virtual Machine Interface

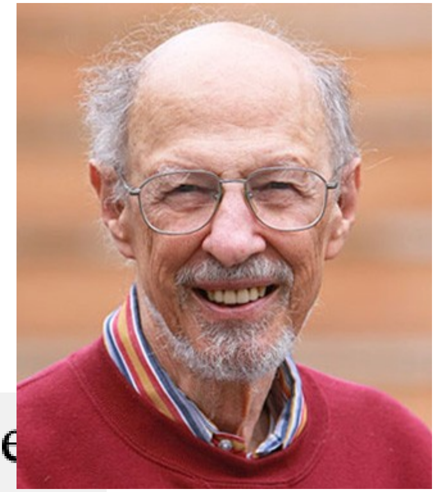
Operating System

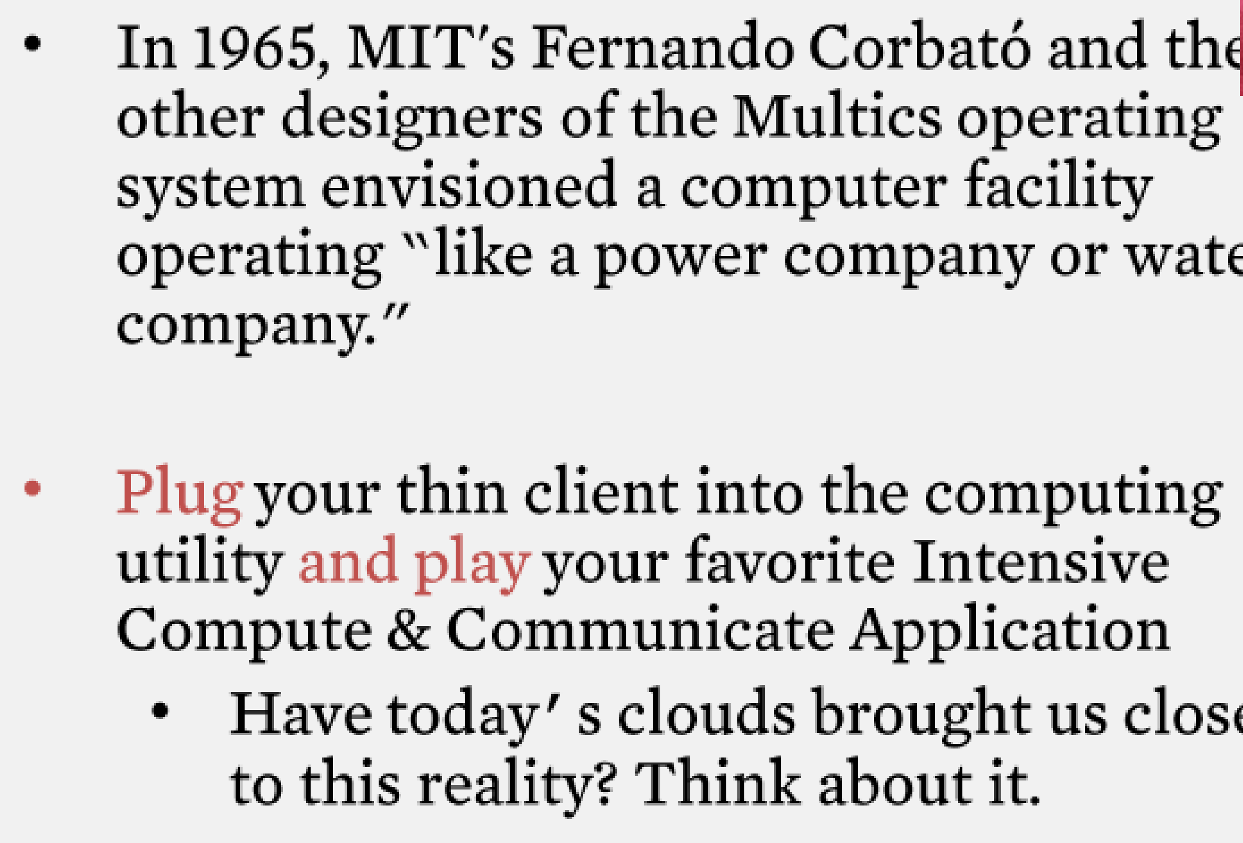
Physical Machine Interface

Hardware

- Software Engineering Problem:
  - Turn hardware/software quirks ⇒ what programmers want/need
  - Optimize for convenience, utilization, security, reliability, etc…
- For any OS area (e.g. file systems, virtual memory, networking, scheduling):
  - What's the hardware interface? (physical reality)
  - What's the application interface? (nicer abstraction)

What do you think about future?
Like people 50 years before





- In 1965, MIT's Fernando Corbató and the other designers of the Multics operating system envisioned a computer facility operating "like a power company or water company."

- Plug your thin client into the computing utility and play your favorite Intensive Compute & Communicate Application
    - Have today’ s clouds brought us closer to this reality? Think about it.