

CS 202-(001): Operating Systems

Instructor: Michael Walfish

TAs: Yash Pazhianur (head)

Kevin Ding

Andrew Hua

Joseph Liao

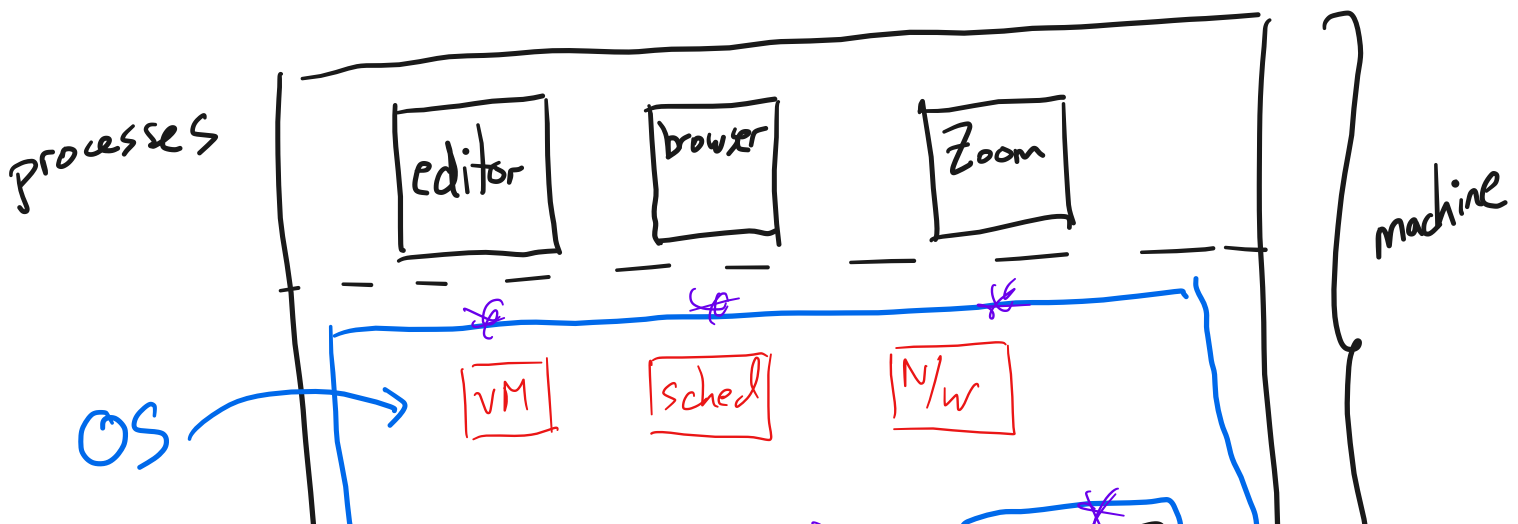
Peter Lin

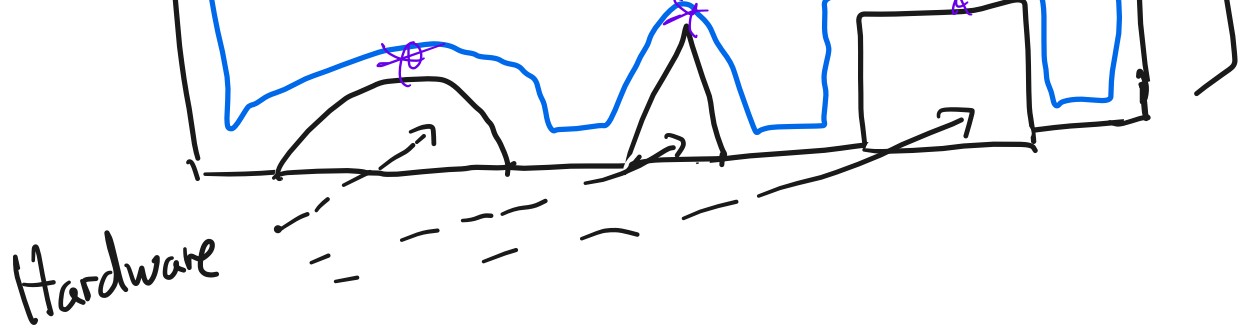
Amy Tao

<http://www.cs.nyu.edu/~mwalfish/classes/25sp>

- 1. Intro & goals
- 2. What is an operating system?
- 3. Why study systems?
- 4. How will we study (operating) systems?
- 5. Mechanics + admin
- 6. History
- 7. Processes (next time)

2. What is an operating system?





Classical description of OS:

- I. Managing the resources of the machine
- II. Abstracting the hardware

Examples?

- file system

Abstraction: seq. of bytes

Isolation: hiding users' files

- Text input

Abstraction: linear stream

Isolation: chars go to the intended application

- Memory

Abstraction: very large conceivable quantity of mem

Isolation: processes cannot access each other's mem

- Scheduling
 - Abstraction : continuous execution
 - Isolation : one heavy consumer cannot monopolize the CPU
-

3. Why study systems?
- a. "how things work"
 - b. ideas are everywhere
 - c. fundamental design trade-offs
 - d. unsolved problems
 - e. skills building
-

4. How will we study?

5. Mechanics + admin

□ Comms

□ Components

□ class

□ labs

□ exams

03/12/25

□ reading

□ practice questions

□ recitation / review; mostly Thurs, some Tuesdays

□ quizzes

□ grading

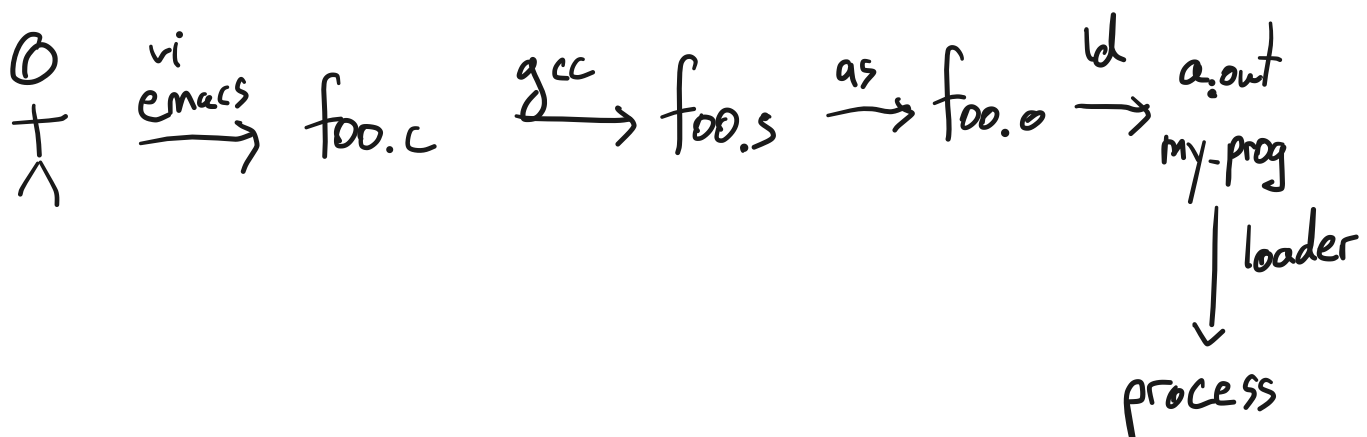
□ policies

6. History (abridged)

Unix

7. Processes

Key abstraction

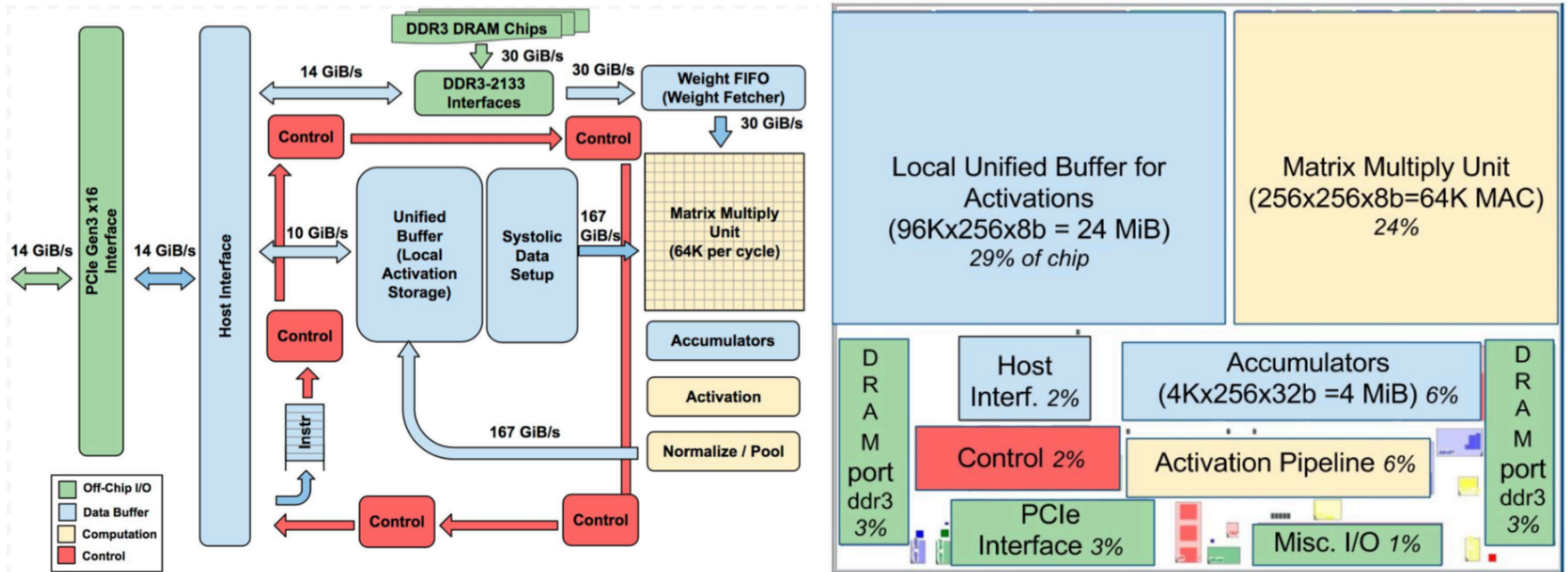


processes can be understood in two ways:

- from the process's point of view

- from the OS's point of view

C? x86-64? My Future Is In Machine Learning!



- Cutting-edge ML backed by custom TPU, unique system software and OS support...



(see class notes for source)



(See class notes for source)