

CS2200  
Systems and Networks  
and Architecture  
Spring 2022

## Lecture 1: Introduction

Alexandros (Alex) Daglis  
School of Computer Science  
Georgia Institute of Technology  
[adaglis@gatech.edu](mailto:adaglis@gatech.edu)

# Introductions

---



## Background

- PhD from EPFL (2018)
- Georgia Tech (since 2019)

Office: Klaus 2322

Office Hours: 30 min after each lecture



Alexandros (Alex) Daglis  
Assistant Professor

School of Computer Science  
Georgia Institute of Technology  
Atlanta, GA 30332, USA

Email: [adaglis@gatech.edu](mailto:adaglis@gatech.edu)  
Web: [cc.gatech.edu/~adaglis/](http://cc.gatech.edu/~adaglis/)

## Research Interests

- Computer Architecture
- Datacenters
- Network-Compute Co-design

# Three CS-2200 Sections

---

- Prof Ramachandran, Prof Lively, Prof Daglis
- **Same** syllabus, resources, exams, assignments, **different** lectures
- Shared Canvas page

# Teaching Assistants

---

- 20 TAs

Bozhidar "Bo" Manev (Head TA)

John Lu

Charles Snider

Hemang Dash

Shuyan Lin

Aidan Donelan

Tristan Rogers

Yesha Jigneshkumar Thakkar

Malav Patel

Borun Song

Carsen Miller

Shih-Huan Chou

Ishaan Guha

Aditya Kaushik

Andrej Vrtanoski

Yuqing Huang

Prit Patel

Maksim Tochilkin

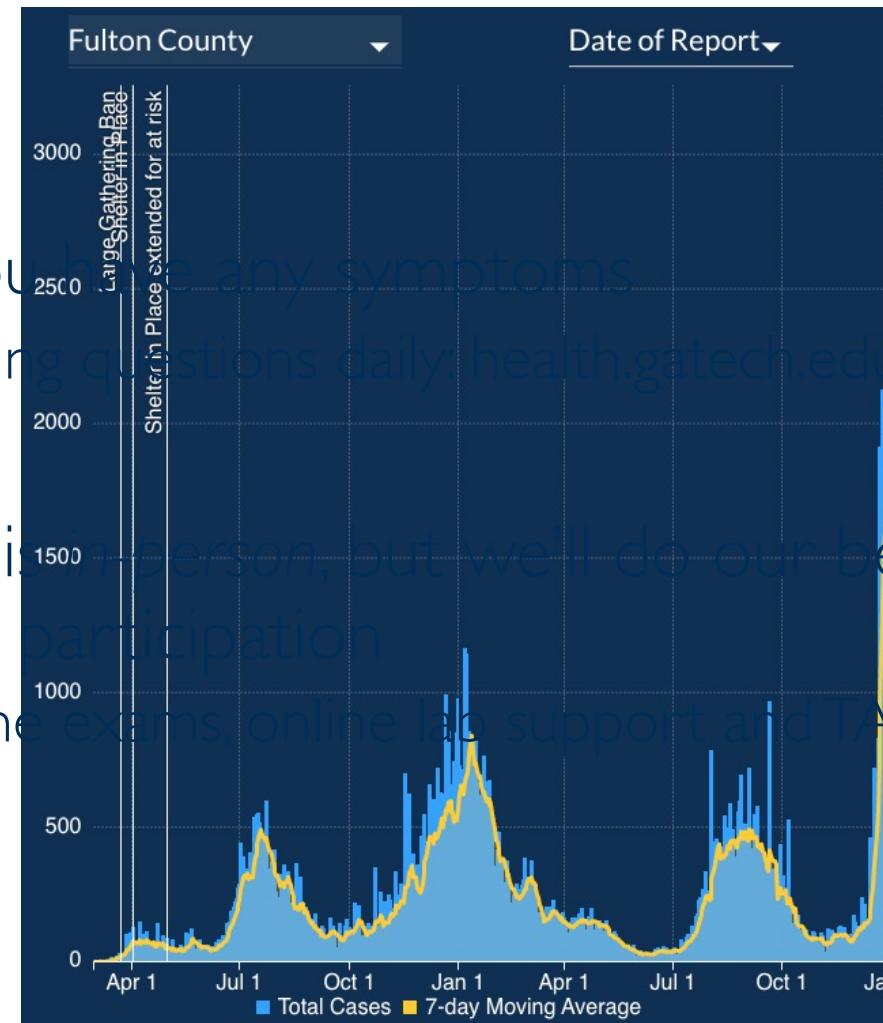
Shunzhi Wen

Reza Reimoo



# A few words on COVID-19

- Protect yourself and your peers – please mask up!



- Please stay home if you have any symptoms
  - Answer the self-screening questions daily: [health.gatech.edu/coronavirus/daily-checklist](http://health.gatech.edu/coronavirus/daily-checklist)
- Official course mode is in person, but we'll do our best to accommodate absences and remote participation
  - Lecture streaming, online exams, online lab support and TA office hours



**Masks recommended**  
for everyone inside  
campus facilities.



# Motivation

---

- How many of you are taking this because you're interested in the subject?
- How many of you are taking this because it's required?

# Objectives

---

What's inside the box?



Learn a lot about systems

Have fun!

# Prerequisites

---

## CS 2110

- Sequential and combination logic
- Memory
- CPU/Datapath
- Instruction-set
- C programming
- Translation Process
  - Preprocessing, Compilation, Linking, Loading
- Linux
- etc.

# Course Outline

---

Broad exposure to computer systems

- Organization of the processor
- Memory hierarchy
- Storage devices
- Parallel processors
- Networking hardware
- Software abstractions in the operating systems for orchestrating their usage
- Networking protocols to connect the computer system to its environment

# Major Topics - Semester Schedule

---

Weeks 1-6:

- Processor: datapath, interrupts, pipelining

Weeks 7-8:

- Process Abstraction and process scheduling

Weeks 9-11:

- Memory Management and Memory Hierarchy

Weeks 12-14:

- Parallel Systems
- Networking Protocols and Distributed Systems
- IO and Disk scheduling

Week 15:

- File Systems

Date	Day	Spring 2022	Wk	Lecture A/B/C			Dates		Recitation Help
		Comments		# Lecture	Reading (Chapter)	Release	Due		
1/10/2022	Mon		1						
1/11/2022	Tue	First Day		1   Introduction/Preassessment		1			
	Wed								
1/13/2022	Thu			2   Feedback/Processors		2			
	Fri								
1/17/2022	Mon	MLK Day	2						
	Tue			3   Processors		2			
	Wed								Intro & circuitsim install help
	Thu			4   Processors		2			
	Fri						P1		
1/24/2022	Mon		3						
	Tue			5   Datapath		3			
	Wed						HW1		Datapath
	Thu			6   Datapath		3			
	Fri								
1/31/2022	Mon		4						
	Tue			7   Interrupts		4			
	Wed						HW2	HW1	Datapath
	Thu			8   Interrupts/Performance		4, 5			
	Fri								
2/7/2022	Mon		5						
	Tue			9   Interrupts/Performance		4, 5			
	Wed						HW3	HW2	Interrupts
	Thu			10   Pipelining		5			
	Fri						P2	P1: Processor	
2/14/2022	Mon		6						
	Tue			11   Pipelining		5			
	Wed						HW4	HW3	Pipelining
	Thu			12   Pipelining		5			
	Fri			Test 1					

# Details

---

- Exams
- Class Website
  - On Canvas
- Piazza

# Homework/Projects

---

- Significant homework and project component to go with each topic
- 3 lab hours in the course credit
- An **excellent** knowledge of C programming is essential for completing the projects

# Assignments

---

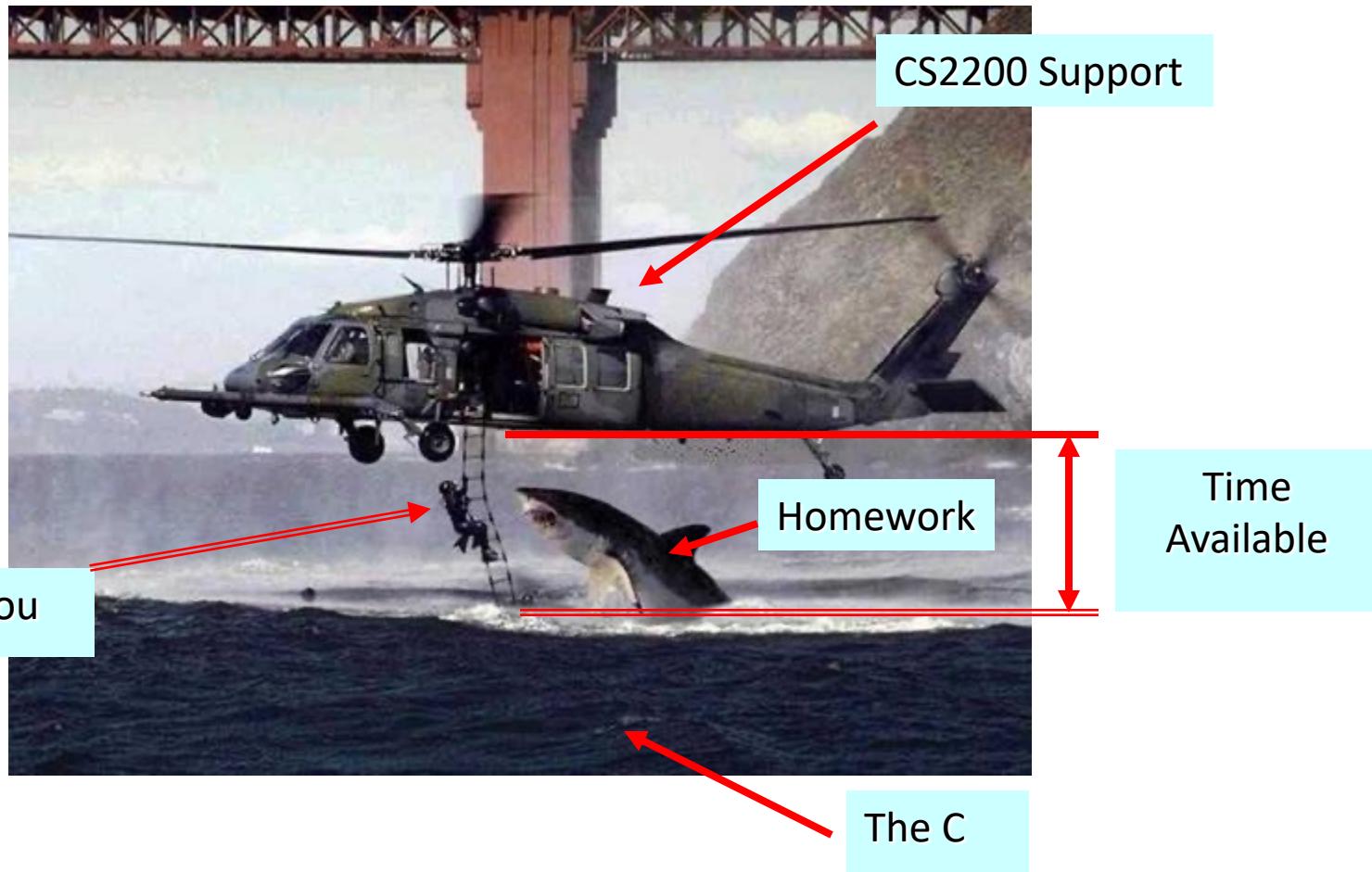
10 Homeworks

5 Projects

1. LC2200 Datapath/Control
  2. I/O Interrupt handler
  3. Virtual Memory
  4. Process Scheduling
  5. Networking
- + optional 6<sup>th</sup> project for extra credit

# A Word About Assignments

---



# Grading Scale

---

4 tests	60%
5 projects	25%
10 homeworks	10%
Participation (TP & Piazza)	5%

# Canvas/Gradescope

---

- Homework Assignments
- Assignment turnin
- Assignment retrieval
- Grades

# Collaboration

---

- Collaboration is allowed and encouraged in this class
- Collaboration means that you can discuss assignments, look at another student's code and help one another out.
- This is not a group project class unless an assignment is specifically designated as such.
- Each student is required to turn in their own homework or project.
- Each student will be required to demo selected assignments and will be expected to be able to understand and explain every part of their submissions (project "interview").

# Project Collaboration Exceptions

---

- Best option A: Submitted project is individual effort
  - max credit 100%
- Acceptable option B: Submitted project is combined effort with peer
  - max credit 50%
  - Collaborative submission must be declared in advance
- Acceptable option C: Submitted project is a peer's, but well-understood
  - max credit 25%
  - Must be able to demo and fully explain solution to TAs during “interview”

# Collaboration

---

- You are expressly forbidden to supply a copy of your homework to another student via electronic means. If you supply an electronic copy of your homework to another student and they are charged with copying you will also be charged. This includes public repositories. That is, you must not store your code in a location where others may access it.
- Homework and projects that are obviously nearly identical will receive a zero and will be sent to the Dean of Students Office of Student Integrity for investigation and possible prosecution. This includes assignments where nearly identical assignments are altered to make them appear different.

# CS 2200 Rules and Regulations

---

- I. Piazza messages are your best communication option. Please use email only if absolutely necessary. If you email the instructor and TA's please make sure to include "cs2200" in the subject. Also, please sign with your real name.

# CS 2200 Rules and Regulations

---

2. You are responsible for turning in assignments on time. This includes allowing for unforeseen circumstances. In other words, plan ahead, start early, finish early.
3. In general, programming assignments should be turned in with a Makefile and all files needed to compile and run the program. The TA grading your submission should be able to make and run your program without adding files, repairing things, etc.
4. Tests must be taken at the scheduled date and time.

# CS 2200 Rules and Regulations

---

5. If you need a certain grade in order to stay in school, maintain a scholarship, etc. the time to worry about this is right from the beginning of the course not during the week before finals. Grades are based on demonstrated performance not individual need based on factors external to the course. Please do not request special consideration based on this type of situation.
  
6. Final grades will be available from OSCAR normally sometime the week after finals. You may review your final and discuss your grades during the following semester in which you are attending GT. Grades will not be discussed over break.

# CS 2200 Rules and Regulations

---

7. If you have any personal problems (family/illness/etc.) please email the Dean of Students' office. They are equipped and authorized to verify the problems and they will issue a note to all your instructors making them aware of the problem and requesting whatever extension, etc. is necessary.
  
8. Canvas announcements should be checked daily. Official announcements about course matters will be posted there. Complaints, questions about your personal problems, etc. should be discussed with your instructor in person or via private Piazza notes.

# CS 2200 Rules and Regulations

---

9. Out of consideration to your fellow students please turn off cell phones, beepers, wristwatch alarms, etc. Also, make every effort to be on time for class. If you are late, please sit near the back and try to avoid as much disruption to the class as possible.
10. If you are graduating and need this course to do so please inform your instructor as soon as possible.
11. Complaints about any aspect of the course should be directed to the course instructor during office hours or via private Piazza notes.

# CS 2200 Rules and Regulations

---

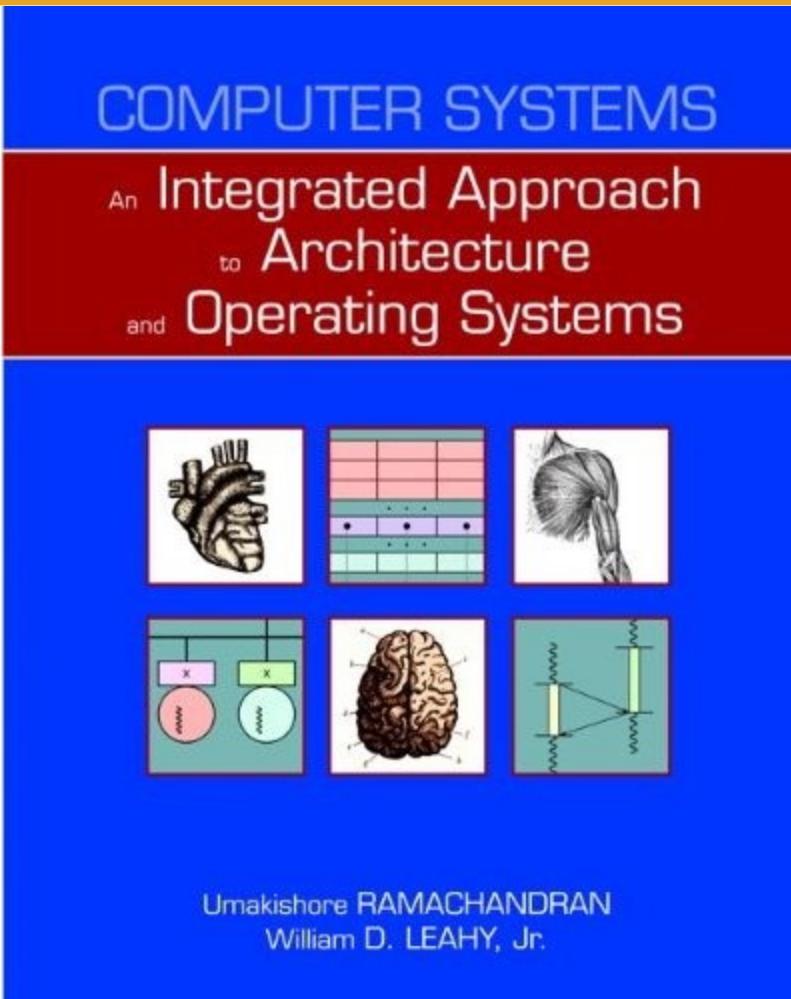
12. The deadline for regrades is 2 weeks after an assignment grade is posted or returned to you. This deadline also applies to picking up items which are returned in class. After this deadline no grade changes will be made and tests not picked up will be destroyed.

# CircuitSim

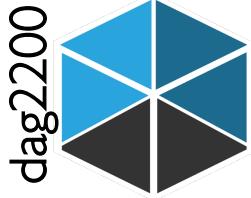
---

- We will be using CircuitSim for the Projects
- Google: CircuitSim

# Textbooks



- Paperback: 784 pages
- Publisher: Addison Wesley; 1 edition (August 9, 2010)
- Language: English
- ISBN-10: 0321486137
- ISBN-13: 978-0321486134
- Average Customer Review: 4.6 out of 5 stars

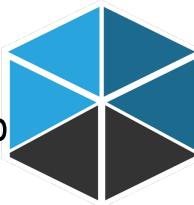


# Turning Point

---

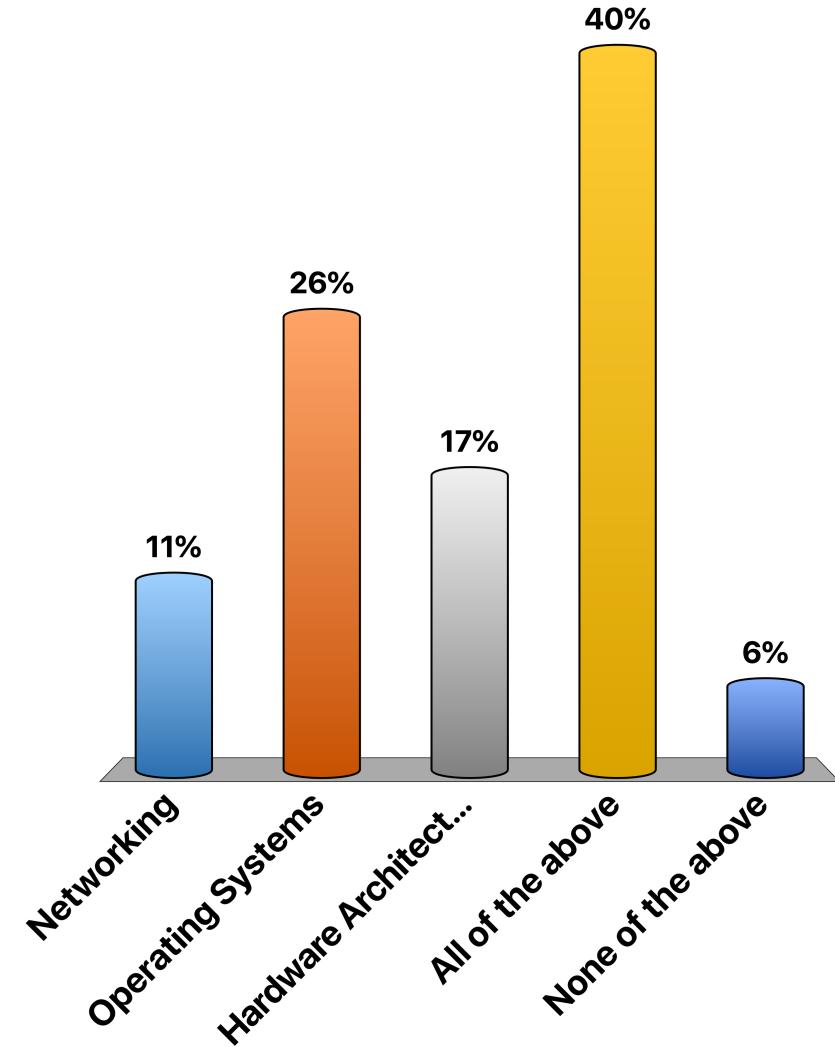
- Used for in-lecture questions – will be used to grade participation!
  - Download “TurningPoint” app on your phone
  - Sign in with your GT credentials
  - Use session ID “dag2200”
  - Question will pop up on your screen!
- 
- You will see this  on the top left corner of each TP slide





# What topic are you interested in the most?

- A. Networking
- B. Operating Systems
- C. Hardware Architecture
- D. All of the above
- E. None of the above



---

---

Questions?

---

---

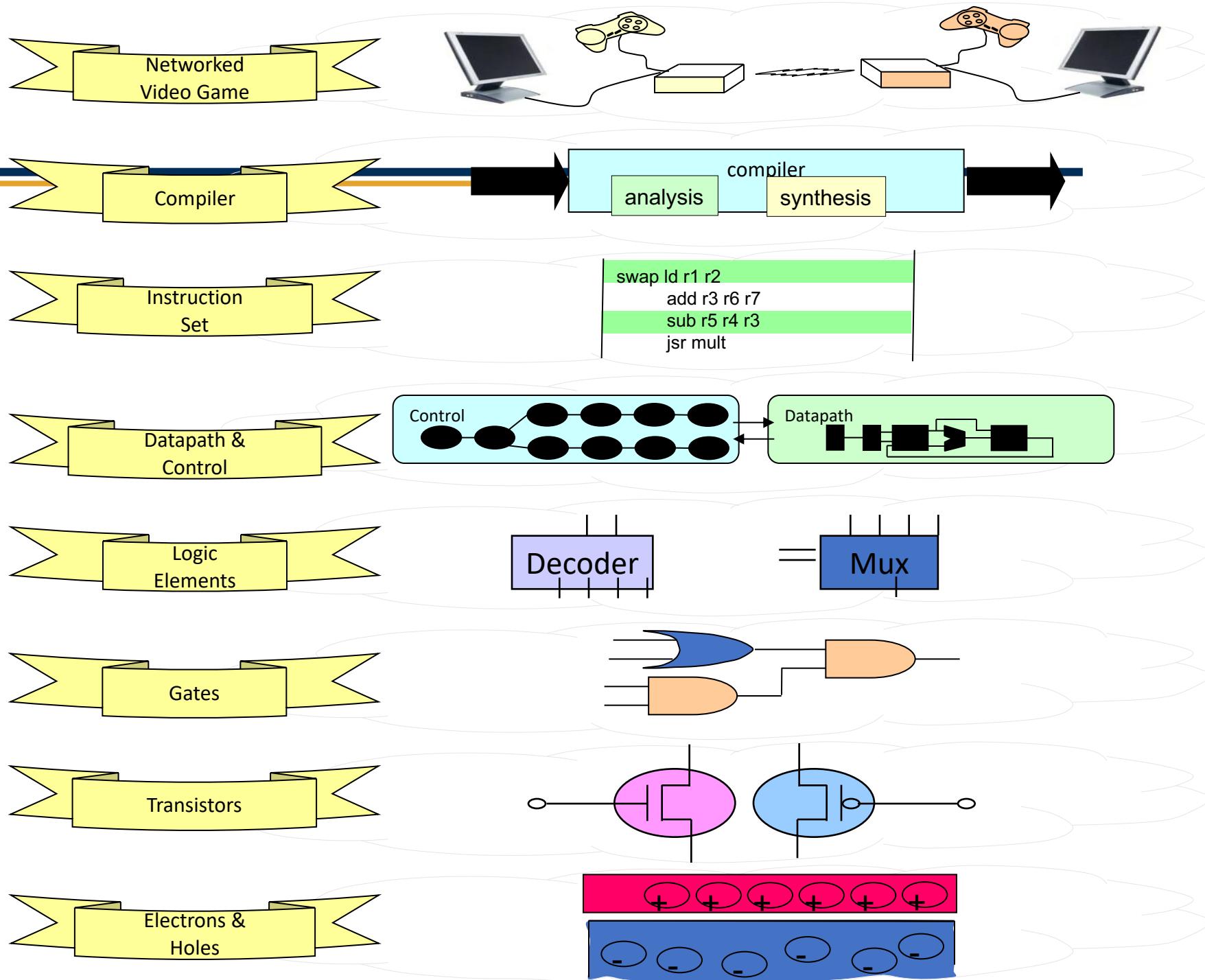
**Looking at the  
big picture!**

Abstraction

# Levels of Abstraction

---





# What will we talk about in this course?

---



Application (Algorithms expressed in High Level Language)

System software (Compiler, OS, etc.)

Computer Architecture

Machine Organization (Datapath and Control)

Sequential and Combinational Logic Elements

Logic Gates

Transistors

Solid-State Physics (Electrons and Holes)

# The Role of the Operating System?

---

- Resource manager—sharing
- Provide consistent interface to resources
- Job scheduler

Client Application  
(Halo 3)



*Player clicks mouse  
cursor on target*

Client Application  
(Halo 3)



*Player clicks mouse cursor on target*

OS: Recognizes interrupt  
Sends it to client application

*It's a mouse interrupt!*

Client Application  
(Halo 3)



Player clicks mouse cursor on target

OS: Recognizes interrupt  
Sends it to client application

*It's a mouse interrupt!*

**CLIENT**

Client Application  
(Halo 3)



Player clicks mouse cursor on target

OS: Recognizes interrupt  
Sends it to client application

*It's a mouse interrupt!*

CLIENT

Client Application creates  
message to send to server  
application



Client Application  
(Halo 3)



Player clicks mouse cursor on target

OS: Recognizes interrupt  
Sends it to client application

*It's a mouse interrupt!*

**CLIENT**

Client Application creates message to send to server application



OS: Sends Message to server



Client Application  
(Halo 3)



OS: Recognizes interrupt  
Sends it to client application

*It's a mouse  
interrupt!*

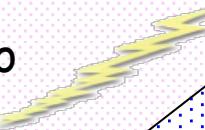
**CLIENT**

Client Application creates  
message to send to server  
application



**SERVER**

OS: Sends  
Message to  
server



OS: Receives  
Message sends  
to server application

*Got a message!*

Client Application  
(Halo 3)



Player clicks mouse cursor on target

OS: Recognizes interrupt  
Sends it to client application

*It's a mouse interrupt!*

**CLIENT**

Client Application creates  
message to send to server  
application



**SERVER**

OS: Sends  
Message to  
server

*Application examines  
message and state of  
game and determines  
Master Chief dies! Sends  
message back to client.*

*Got a message!*

OS: Receives  
Message sends  
to server application

Client Application  
(Halo 3)



Player clicks mouse cursor on target

OS: Recognizes interrupt  
Sends it to client application

*It's a mouse interrupt!*

**CLIENT**

Client Application creates message to send to server application



**SERVER**

OS: Sends Message to server

*Application examines message and state of game and determines Master Chief dies! Sends message back to client.*

OS: Receives Message sends to server application

*Got a message!*

OS: Sends Message to client

Client Application  
(Halo 3)



Player clicks mouse cursor on target

OS: Recognizes interrupt  
Sends it to client application

*It's a mouse interrupt!*

**CLIENT**

Client Application creates  
message to send to server  
application



OS: Sends  
Message to  
server

*Application examines  
message and state of  
game and determines  
Master Chief dies! Sends  
message back to client.*

*Got a message!*

**SERVER**

OS: Receives  
message and  
sends it to  
application

OS: Receives  
Message sends  
to server application

OS: Sends  
Message to client

Client Application  
(Halo 3)



Player clicks mouse cursor on target

OS: Recognizes interrupt  
Sends it to client application

*It's a mouse interrupt!*

**CLIENT**

Client Application creates message to send to server application



Client Application generates required images, etc.  
Sends I/O requests to OS

OS: Sends Message to server

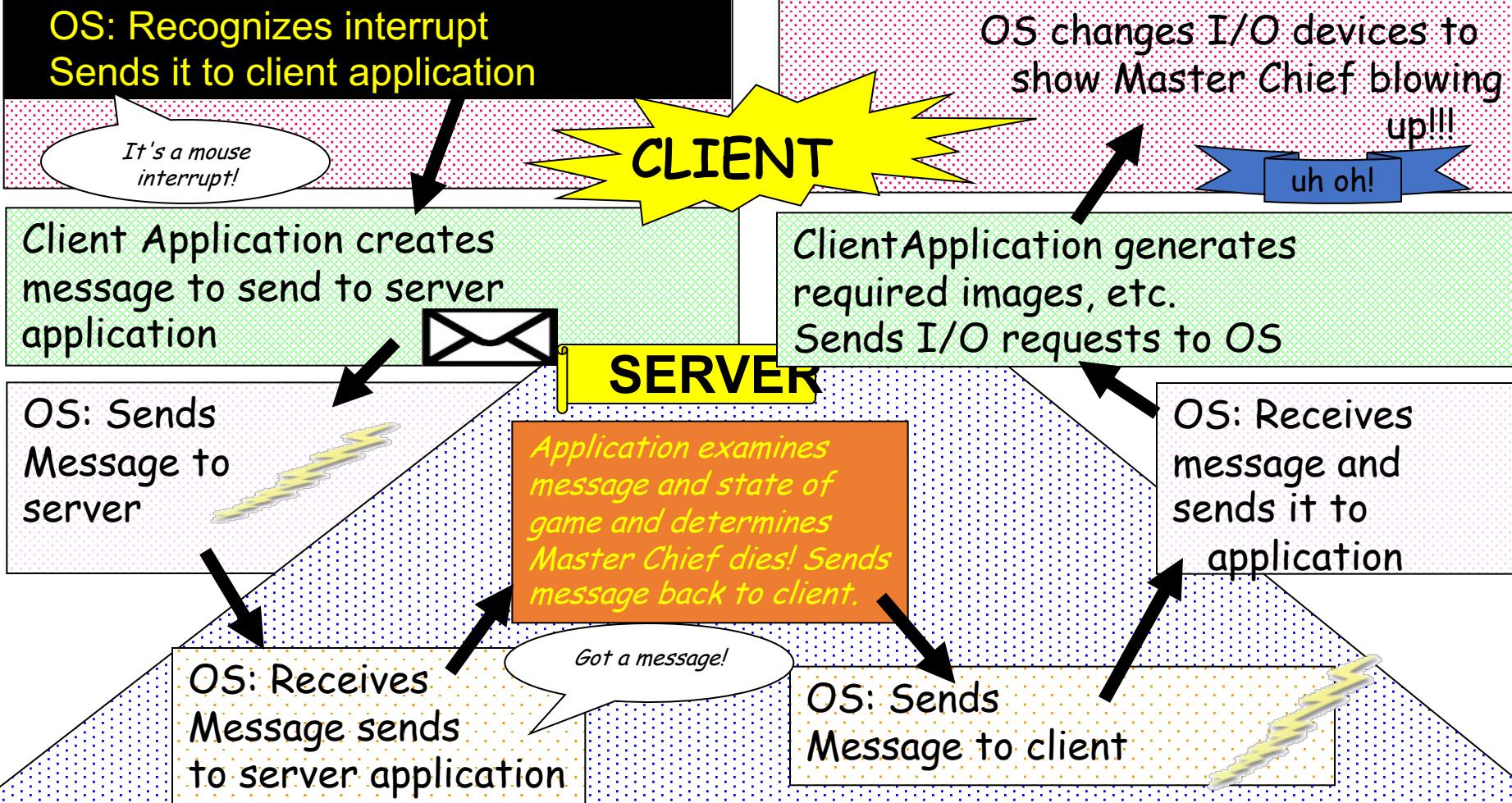
*Application examines message and state of game and determines Master Chief dies! Sends message back to client.*

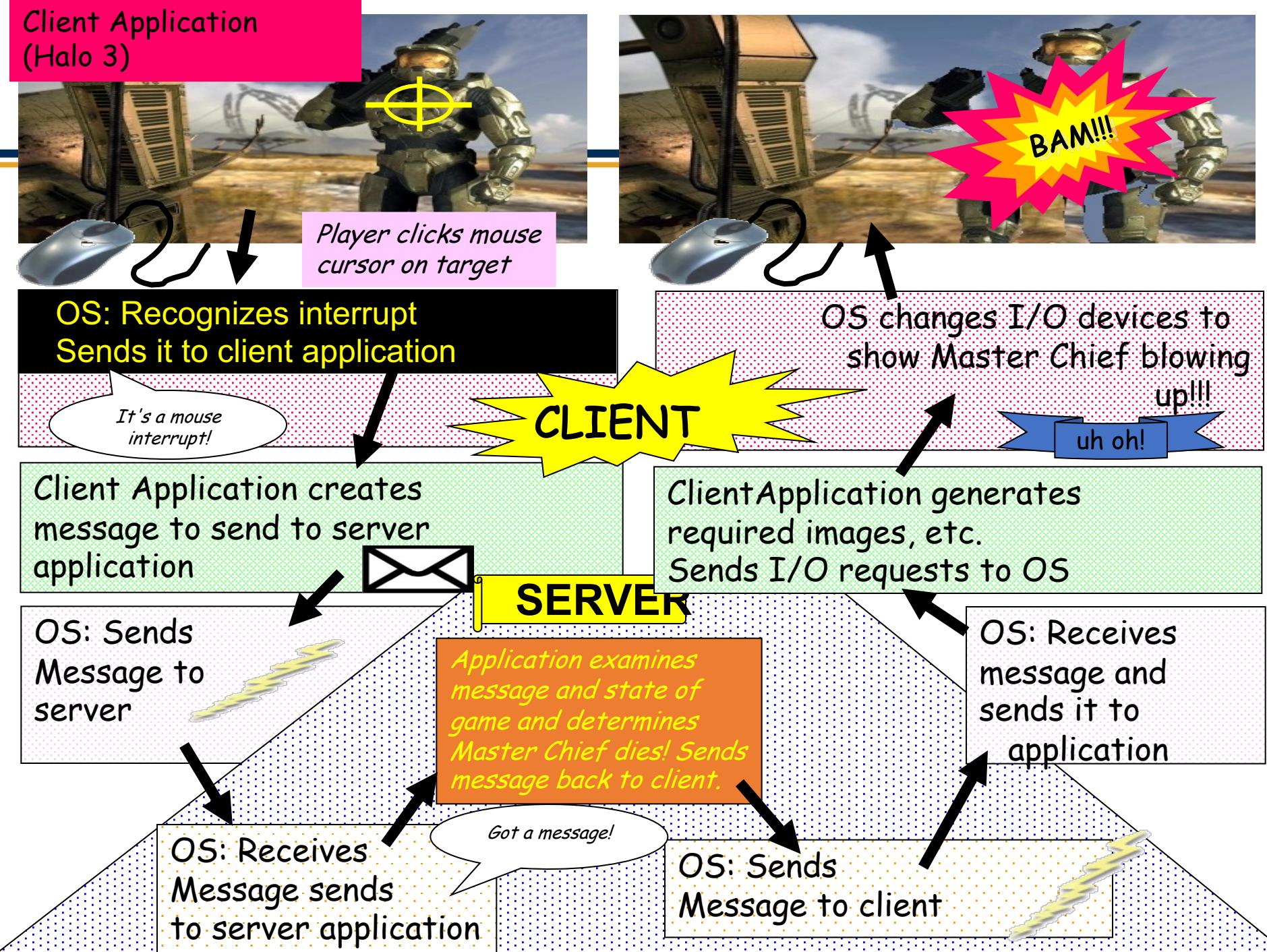
OS: Receives message and sends it to application

OS: Receives Message sends to server application

OS: Sends Message to client

Client Application  
(Halo 3)

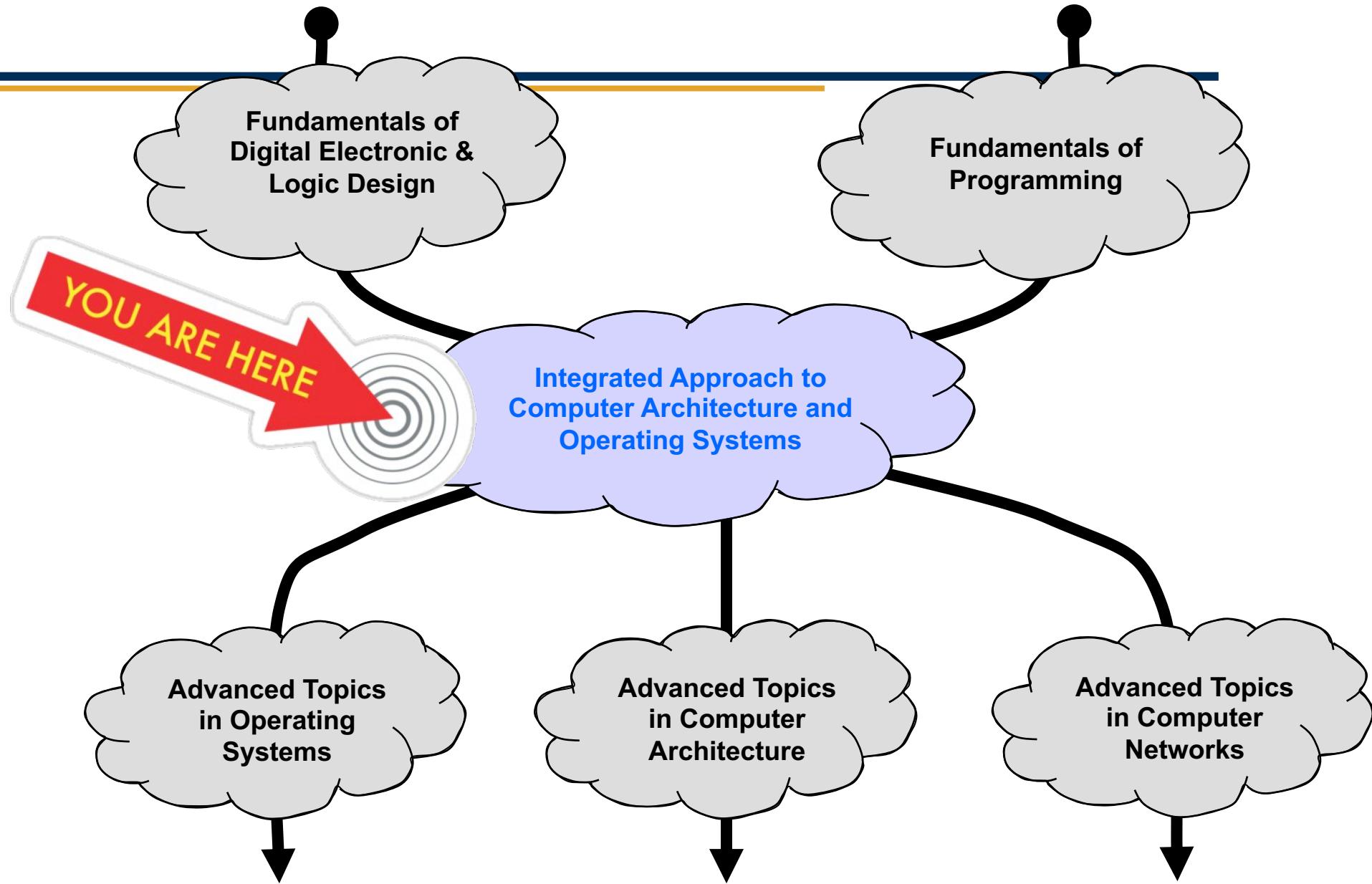




# Why are we studying this?

---

- Very cool!
- Need to have working knowledge one level down
- Somebody has to understand how the darn things work
- Somebody has to tell CmpE's what we need
- Be able to predict GameStop stock movement



---

---

Questions?