ECE566
Enterprise Storage Architecture

Fall 2019

Introduction

Tyler Bletsch
Duke University

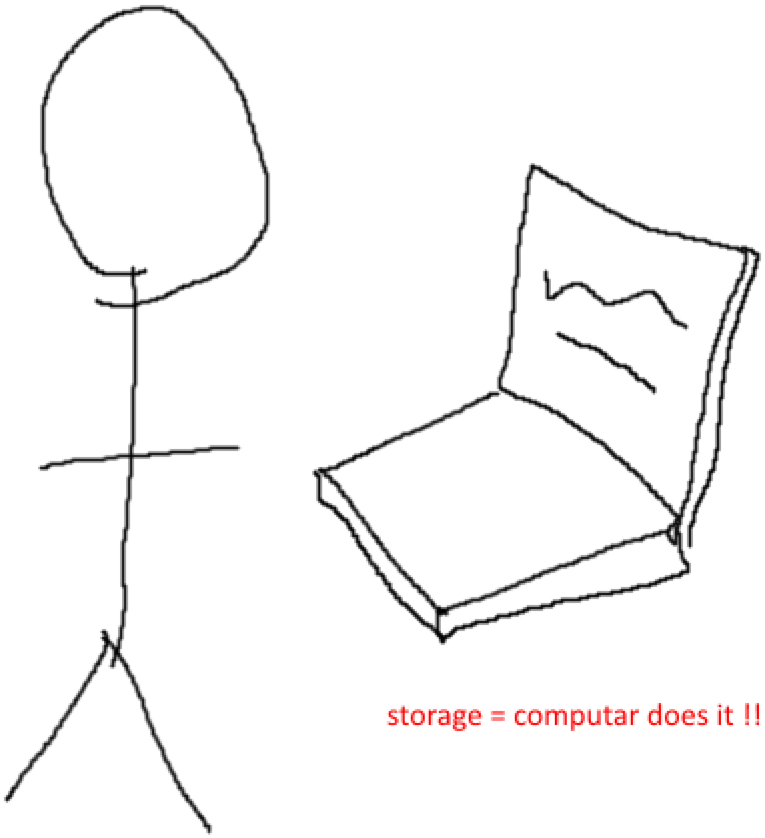
Slides include material from Vince Freeh (NCSU)

Instructor and TAs

- Professor: Tyler Bletsch
  - Office: Hudson Hall 106
  - Email: Tyler.Bletsch@duke.edu
  - Office Hours: See course site
- TA:
  - Bonan Yan (bonan.yan@duke.edu)

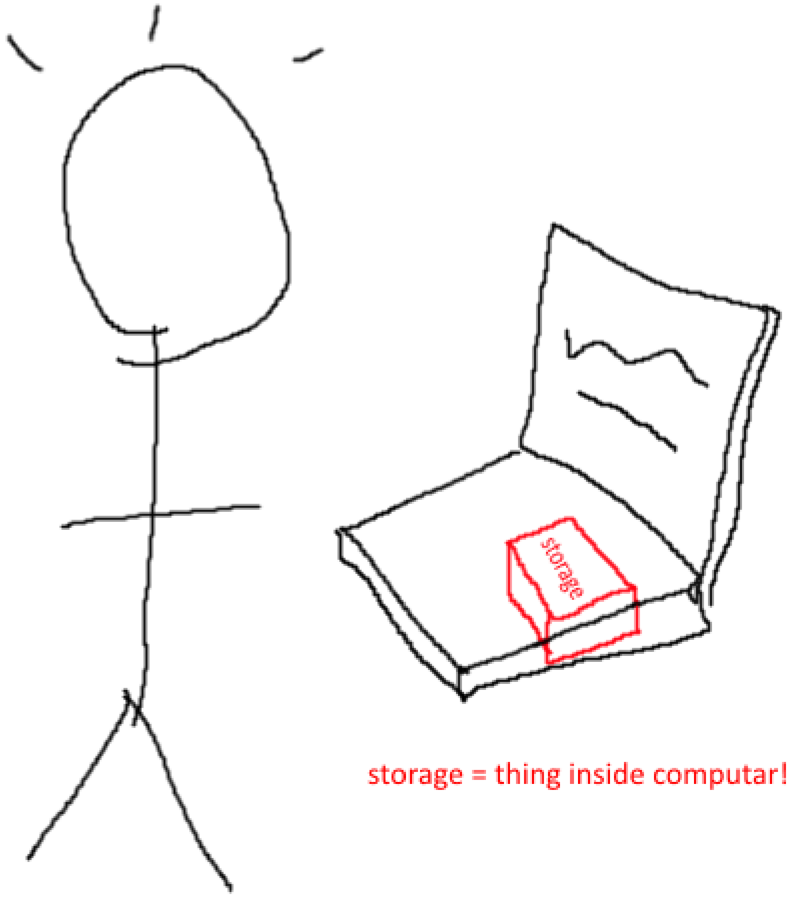
MOTIVATION

Average person’s view of storage



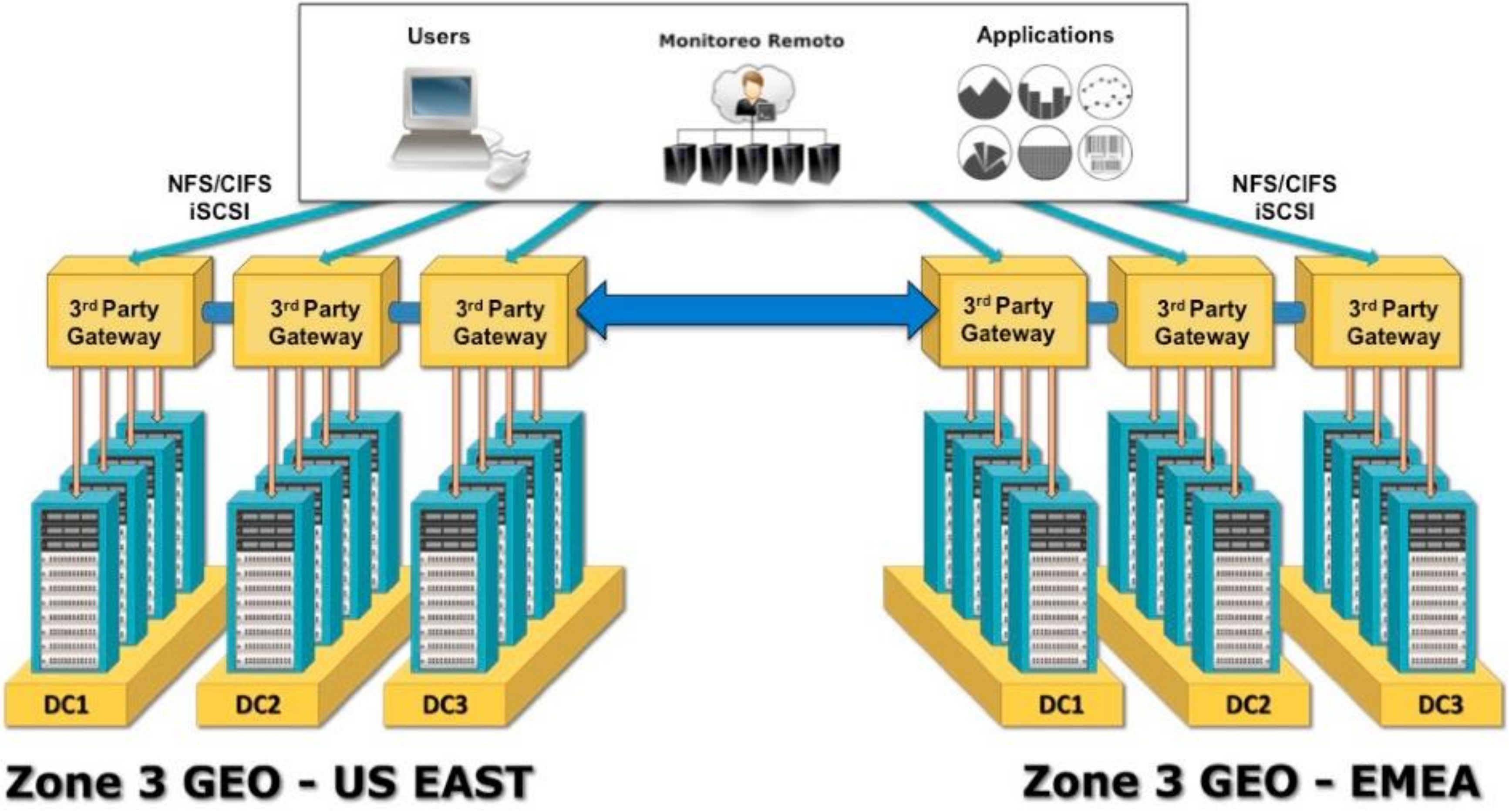
storage = computar does it !!

Average engineer’s view of storage



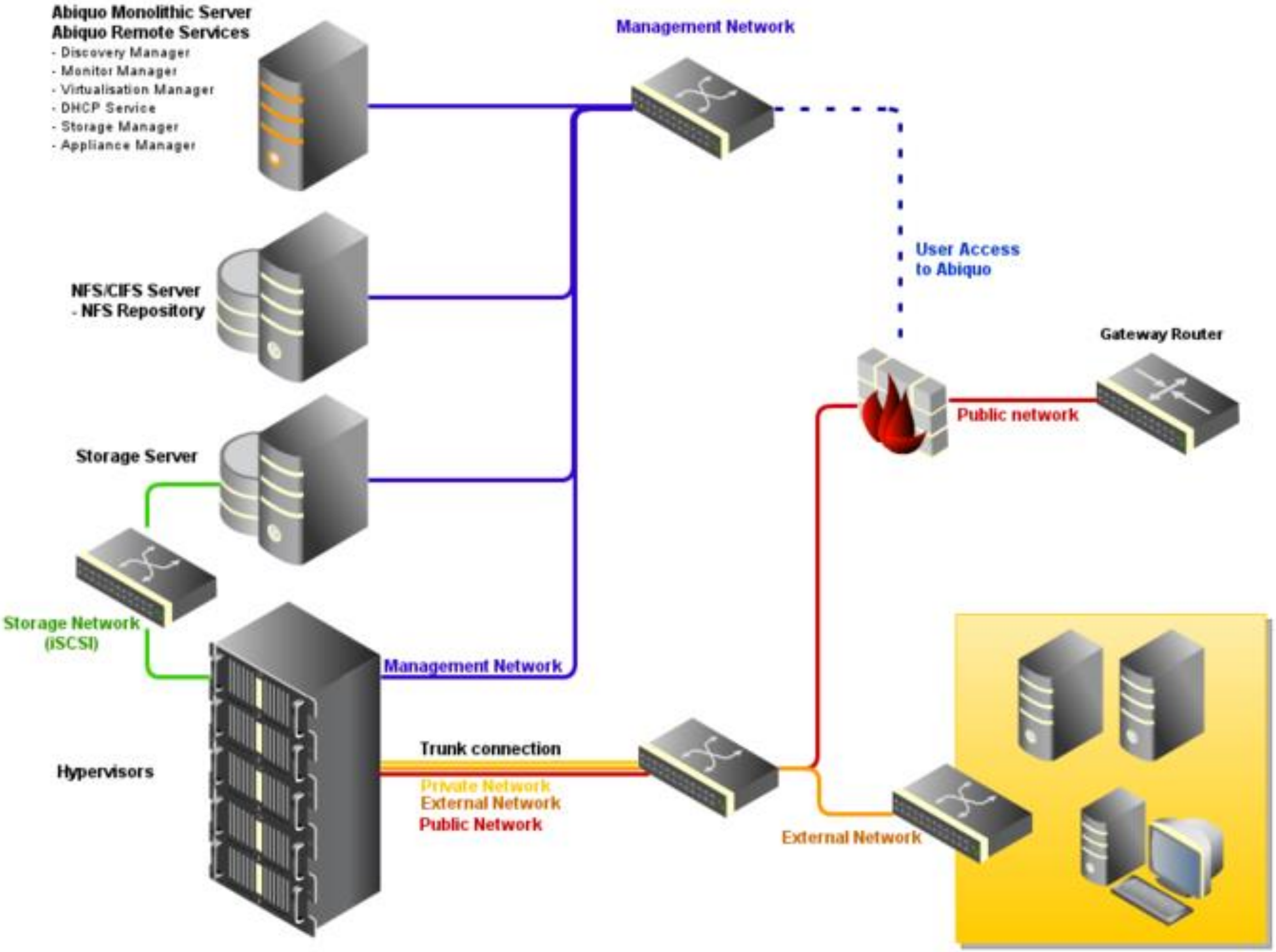
storage = thing inside computar!!

A few enterprise storage architectures (1)



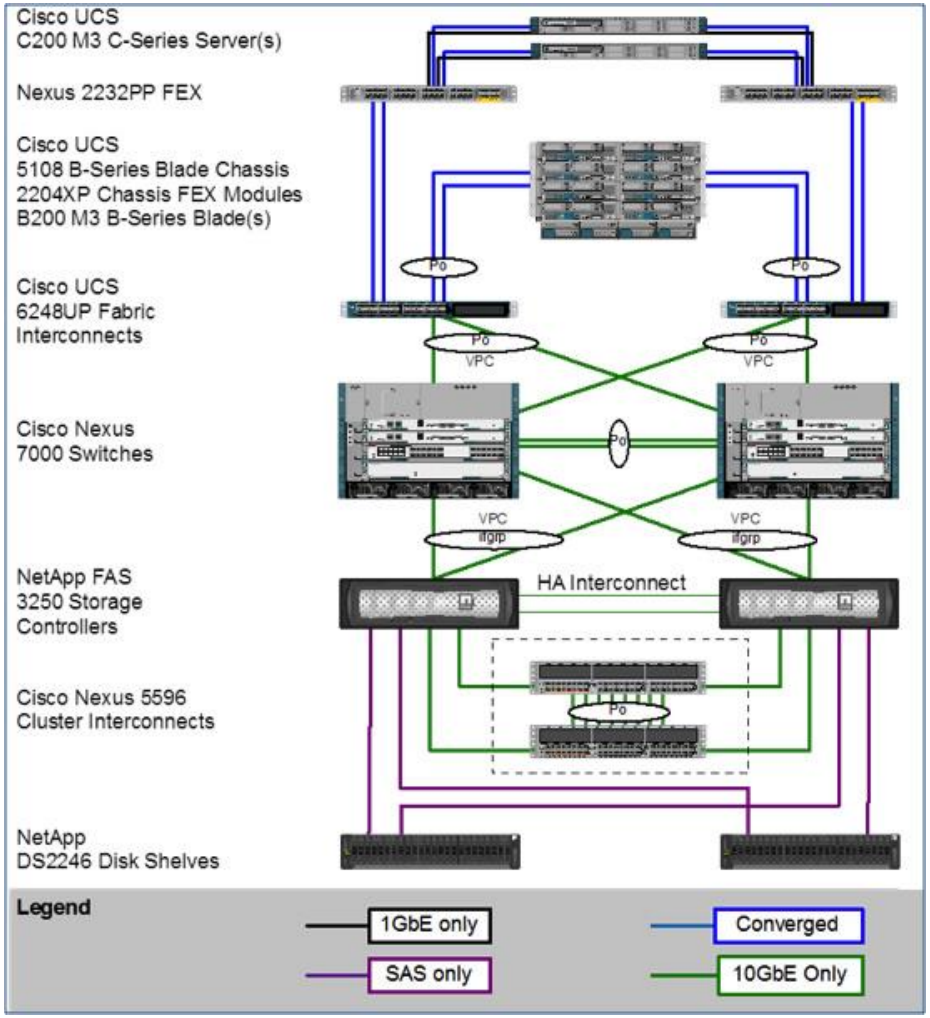
- From: http://www.storagenewsletter.com/rubriques/software/massively-scalable-himalaya-architecture-by-amplidata/

A few enterprise storage architectures (2)



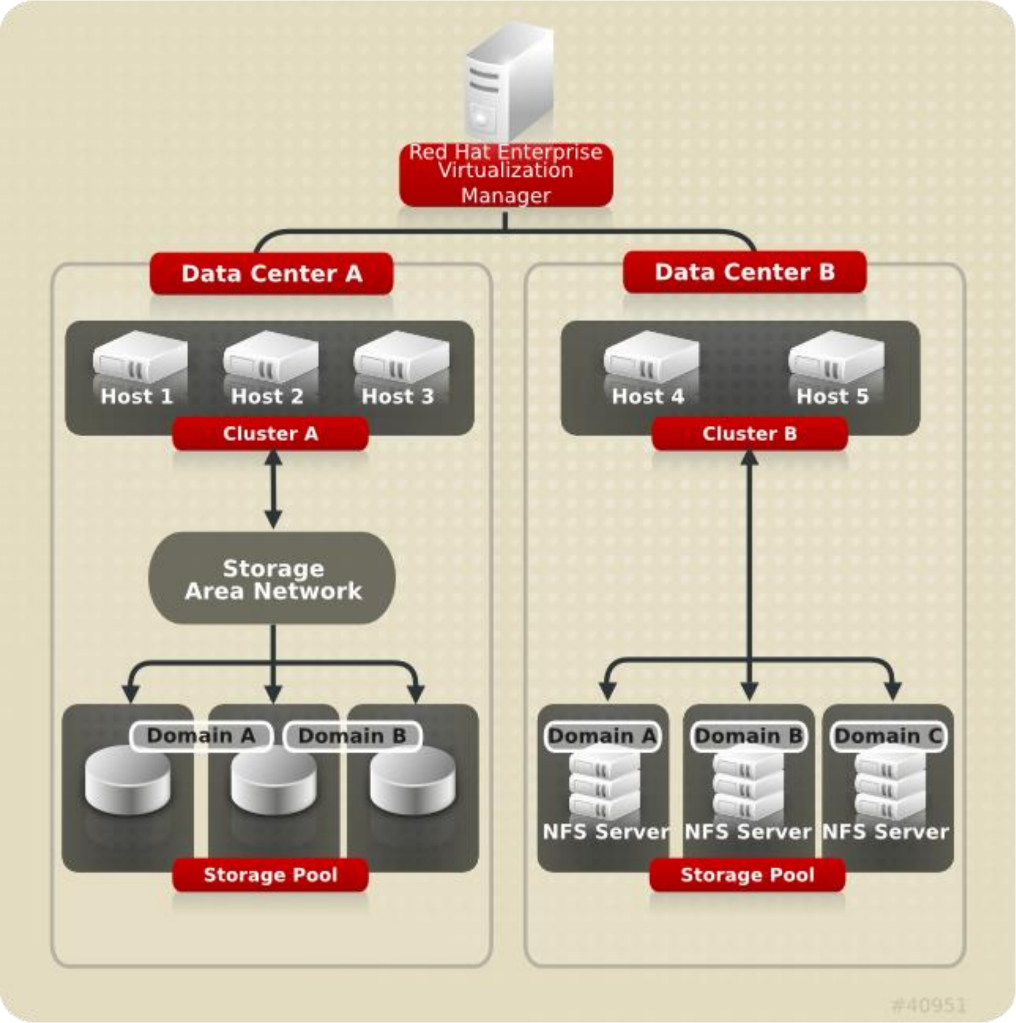
- From: http://wiki.abiquo.com/display/ABI20/Monolithic+Architecture

A few enterprise storage architectures (3)



- From: http://community.netapp.com/t5/Tech-OnTap-Articles/FlexPod-Innovation-and-Evolution/ta-p/85156

A few enterprise storage architectures (4)



- From: https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Virtualization/3.0/html/Technical_Reference_Guide/chap-Technical_Reference_Guide-Storage_Architecture.html

Why do all this? What problems are we solving?

- **Cost**: Is it cheap enough?
- **Capacity**: Can it hold enough?
- **Performance**: Is it fast enough?
- **Accessibility**: Can the data be accessed by everyone who needs it?
- **Security**: Is data protected from unauthorized access?
- **Reliability**: Is the downtime probability low enough?
- **Integrity**: Is data protected from hardware failures, disasters, and malicious attacks?
- **Compliance**: Do I keep data long enough safely?
- **Accountability**: Can I track all changes?
- **Space efficiency**: How much floor space do I need?
- **Power efficiency**: How many watts do I burn?

Why do all this? What problems are we solving?

Color code: how well can a simple drive in a laptop let you control these variables?

- **Cost**: Is it cheap enough?
- **Capacity**: Can it hold enough?
- **Performance**: Is it fast enough?
- **Accessibility**: Can the data be accessed by everyone who needs it?
- **Security**: Is data protected from unauthorized access?
- **Reliability**: Is the downtime probability low enough?
- **Integrity**: Is data protected from hardware failures, disasters, and malicious attacks?
- **Compliance**: Do I keep data long enough safely?
- **Accountability**: Can I track all changes?
- **Space efficiency**: How much floor space do I need?
- **Power efficiency**: How many watts do I burn?

Online course Info

- Course Web Page: static info

➡ **https://people.duke.edu/~tkb13/courses/ece566/**

  - Syllabus, schedule, slides, assignments, rules/policies, prof/TA info, office hour info
  - Links to useful resources

- **Piazza**: questions/answers
  - Post all of your questions here
  - Questions must be "public" unless good reason otherwise
  - **No code** in public posts!
- **GradeScope**: Submit annotated PDFs for grading
- **Sakai**: just assignment submission and gradebook

Where to get info

- This info is fairly industry-connected, no great textbook
  - Semi-exception: “Evolution of the Storage Brain” by Larry Freeman (not a required text)
- Course material will come from lectures and supplementary readings
  - See course site for resources
- Additional independent research on your part will likely be necessary!

Grading Breakdown

Project: 45%

Assignment	%
Project initial proposal	2%
Project final proposal	3%
Project status reports	5%
Project final report	10%
Project final presentation	5%
Project final demo	20%
Homeworks/programs/labs	45%
Final exam	10%

HOMEWORKS, LABS, AND PROGRAMS

Lab Motivation: What is a computer?

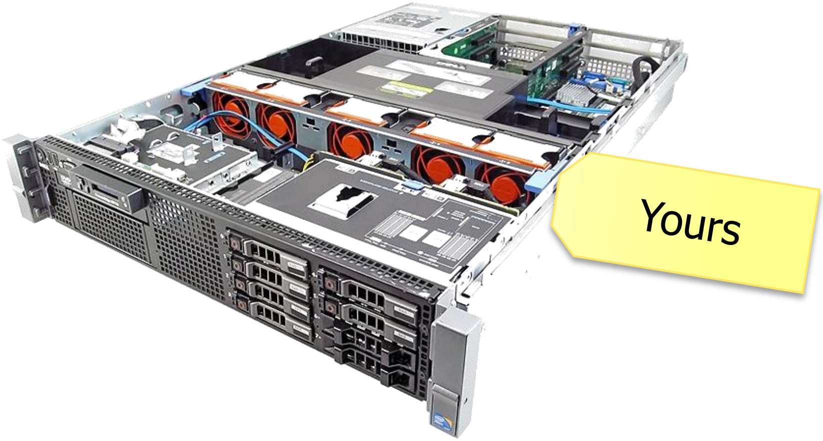
- Computers are:
  - ~~Abstract theoretical math engines that float around on the internet?~~

  - PHYSICAL OBJECTS
  - MADE OF MATERIALS
  - IN THE REAL WORLD
  - AND YOU CAN TOUCH THEM
  - AND PUT THEM PLACES
  - WITH YOUR ARMS/LEGS/FINGERS/BODY
  - AND LIKE A SCREWDRIVER OR WHATEVER!!!!!!!!

Result: this course is HANDS ON

- Historically, the most popular assignments have been the realistic, hands-on ones. So I've added a *lot* of hands-on experience to the course.

- Each student group will be assigned a physical storage server which is upstairs in Hudson 214
- **Lab 0** will have you prepare and deploy this server.
- **Labs 1+** will have you do realistic storage tasks on it.



Labs vs Homeworks

Labs

- Group work
- Hands-on
- Usually on your server
- Submitted via GradeScope (and Sakai for code)
- Can discuss concepts with other groups, but not answers

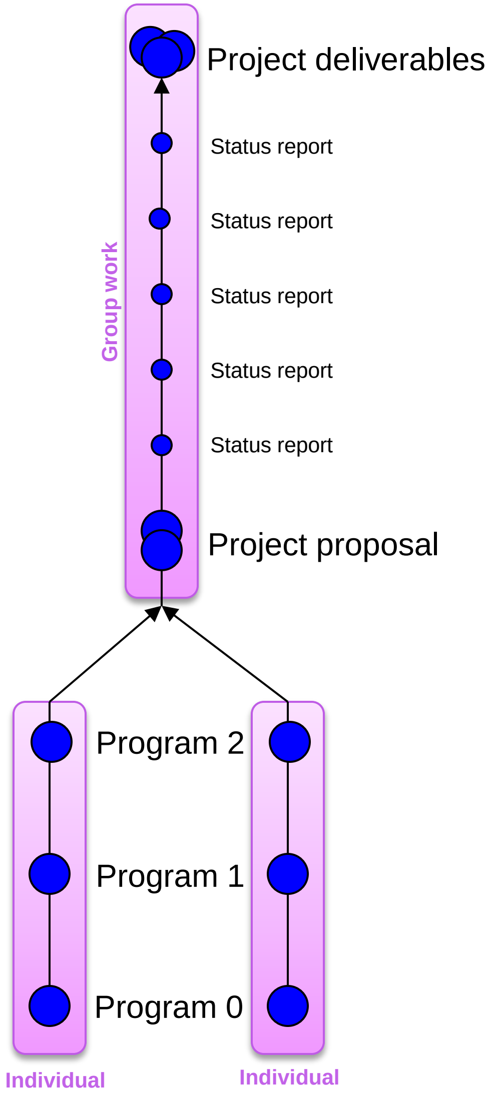
Homeworks

- Individual work
- Pen-and-paper questions
- Submitted via GradeScope (and Sakai for code)
- Can discuss concepts with others, but not answers

Also: a few “Program” assignments

- Project will involve writing filesystem code using FUSE

- Assignments “Program 0”, “Program 1”, “Program 2” are individual
  - Introduce you to FUSE
  - Work you through writing a basic filesystem
  - Prepare you for the project



Late penalties

- Late homework/lab/program incur penalties as follows:
  - Submission is 0-24 hours late: total score is multiplied by 0.9
  - Submission is 24-48 hours late: total score is multiplied by 0.8
  - Submission is more than 48 hours late: total score is multiplied by the Planck constant (in J·s)

- NOTE: If you feel in advance that you may need an extension, contact the instructor.

Labs are group work

- Lab assignments – done ***together*** as a group

- What does “together” mean?

- It means that everyone must understand all of it

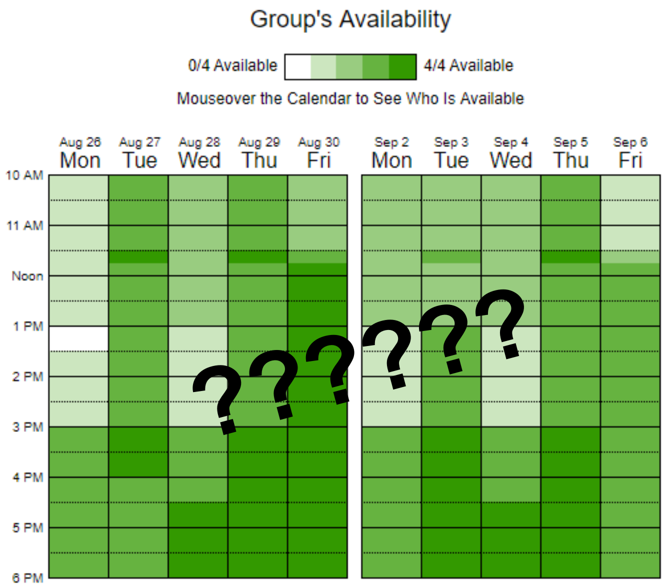
- If I ask “How did this part work?”, you cannot answer “I didn’t work on that part”!

- How do we check? **Lab quizzes**: Quick in-class assignments that are easy to answer if you were involved in the lab work.

Class lab sessions to kickstart homework

- We’re going to schedule a few **class-wide lab sessions** so everyone can start to work on their server with instructor support
  - Why not a separate lab section? We don’t need every week...

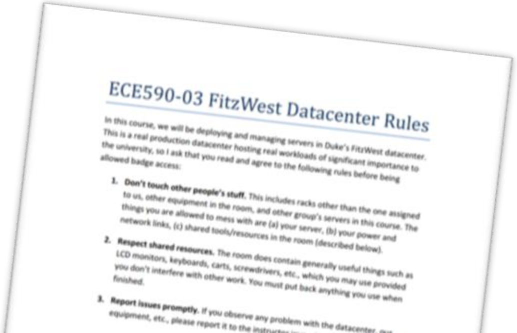
- Be sure to respond to the **scheduling survey** that I sent; deadline is end of today!



FitzWest Datacenter

- You will eventually deploy your server in a real datacenter: the FitzWest server room in the CIEMAS basement
  - This means you'll have **badge access** to a real datacenter

- Datacenter rules (you need to sign this to get access):
  1. **Don't touch other people's stuff.** Includes other racks, other equipment, and other group's servers in this course. You can touch your server, its cables, and shared tools.
  2. **Respect shared resources.** The room has LCD monitors, keyboards, carts, screwdrivers, etc., which you can use. You must not interfere with IT operations and you must put stuff away when done.
  3. **Report issues promptly.** Tell me if anything's wrong.



ECE590-03 FitzWest Datacenter Rules

In this course, we will be deploying and managing servers in Duke's FitzWest datacenter. This is a real production datacenter hosting real workloads of significant importance to the university, so I ask that you read and agree to the following rules before being allowed badge access:

1. **Don't touch other people's stuff.** This includes racks other than the one assigned to us, other equipment in the room, and other group's servers in this course. The things you are allowed to mess with are (a) your server, (b) your power and network links, (c) shared tools/resources in the room (described below).
2. **Respect shared resources.** The room does contain generally useful things such as LCD monitors, keyboards, carts, screwdrivers, etc., which you may use provided you don't interfere with other work. You must put back anything you use when finished.
3. **Report issues promptly.** If you observe any problem with the datacenter, our equipment, etc., please report it to the instructor

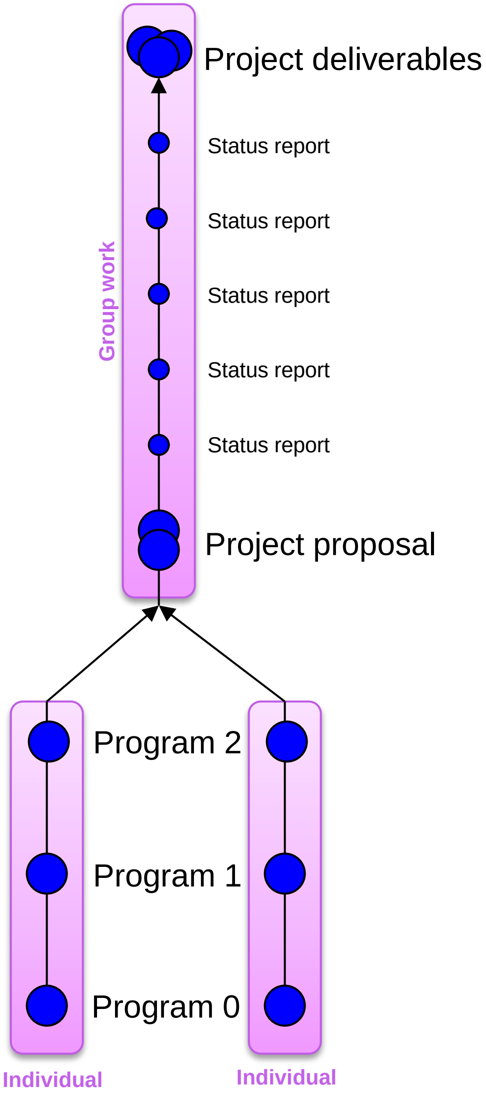
< Print, sign, and turn in to gain access

THE PROJECT

The Project

- **Initial proposal**: Say what you're going to do and how.
  - Write-up plus 60-minute meeting scheduled out of class.
  - Must include weekly schedule!
  - Get feedback
- **Final proposal**: Incorporate feedback from above.
- **Weekly status reports**: Small report that shows progress vs proposed schedule.
- **Workdays:** Time to meet with me in class to steer your project.
- **Final report**: Describe your work (max 8 pages).
- **Final presentation**: Demo your work and explain the implementation process to the class (15 min).
- **Final demo**: Defend your project to the instructor.
  - 60+ minute meeting scheduled out of class.

- ***Read course page for details!***



The project is also group work

- Project work – also done ***together*** as a group

- The word "together" still means that everyone must understand all of it!

- Again, you can't say "I didn't work on that part"!

POLICIES

Grade Appeals

- All regrade requests must be in writing
  - Email the TA who graded the question (we'll indicate who graded what)
- After speaking with the TA, if you still have concerns, contact the instructor
- All regrade requests must be submitted no later than 1 week after the assignment was returned to you.

Academic Misconduct

- Academic Misconduct
  - Refer to Duke Community Standard
  - Labs are groupividual – everyone works on it
  - Common examples of cheating:
    - Running out of time and using someone else's output
    - Borrowing code from someone who took course before
    - Using solutions found on the Web
    - Having a friend help you to debug your program
- I will not tolerate any academic misconduct!
  - Software for detecting cheating is very, very good … and I use it
- “But I didn’t know that was cheating” is not a valid excuse

Our Responsibilities

- The instructor and TA will…
  - Provide lectures/recitations at the stated times
  - Set clear policies on grading
  - Provide timely feedback on assignments
  - Be available out of class to provide reasonable assistance
  - Respond to comments or complaints about the instruction provided

- Students are expected to…
  - Receive lectures/recitations at the stated times
  - Turn in assignments on time
  - Seek out of class assistance in a timely manner if needed
  - Provide frank comments about the instruction or grading as soon as possible if there are issues
  - Assist each other *within the bounds of academic integrity*

Course summary

- We have **hard disks** and **solid-state drives (SSDs)**
- We can use **RAID** to combine performance and capacity while masking effects of drive failure
- The concept of files and directories comes from **File Systems**, a rich field of study.
- We can provide virtual disks to users over **Storage Area Network (SAN)** protocols
- We can provide file access to users using **Network-Attached Storage (NAS)** protocols
- We can provide **storage as a service (SaaS)** via cloud-type protocols.
- Storage efficiency can be improved with **data deduplication** and **compression**.
- We need to preserve **business continuity**:
  avoid downtime and lost data through **backups** and **high availability**
- Storage arrays are deployed based on **workload sizing**.
- Storage is often folded into a complete hardware/software stack: **converged architecture**.
- Storage systems are large enough that **management/monitoring** is its own challenge.
- Storage architects need to understand **basic finance** and **legal/compliance issues**