ECE590-03
Enterprise Storage Architecture

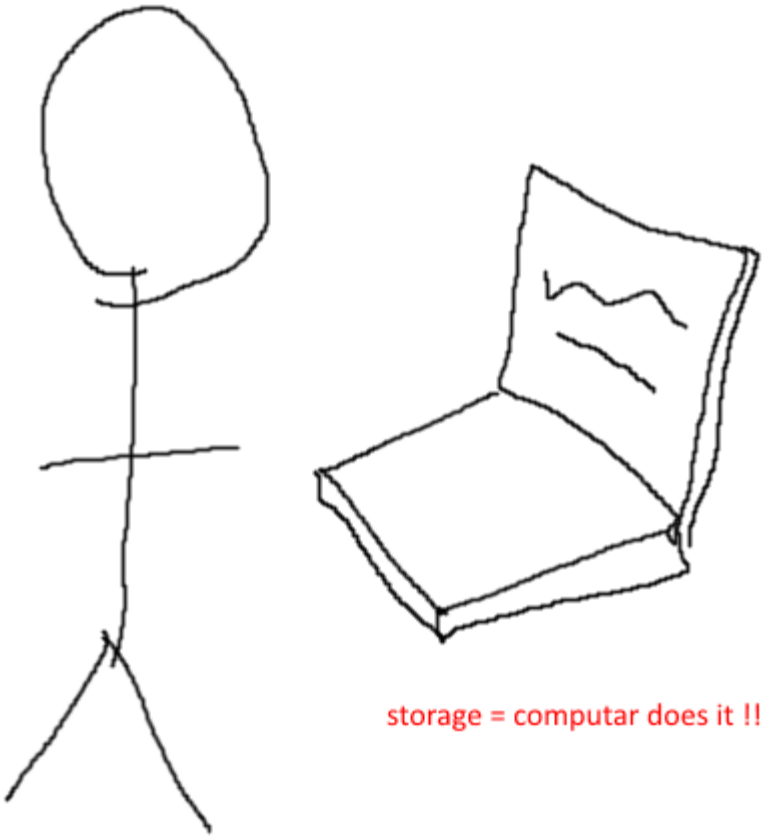
Fall 2016

Introduction

Tyler Bletsch
Duke University

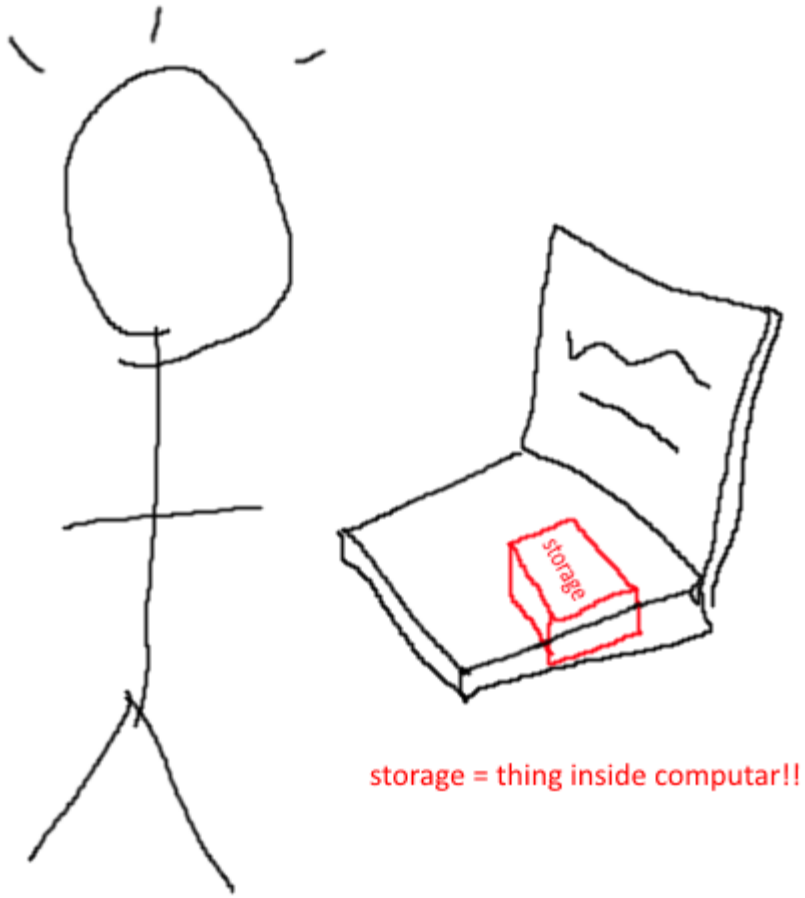
Slides include material from Vince Freeh (NCSU)

Average person's view of storage



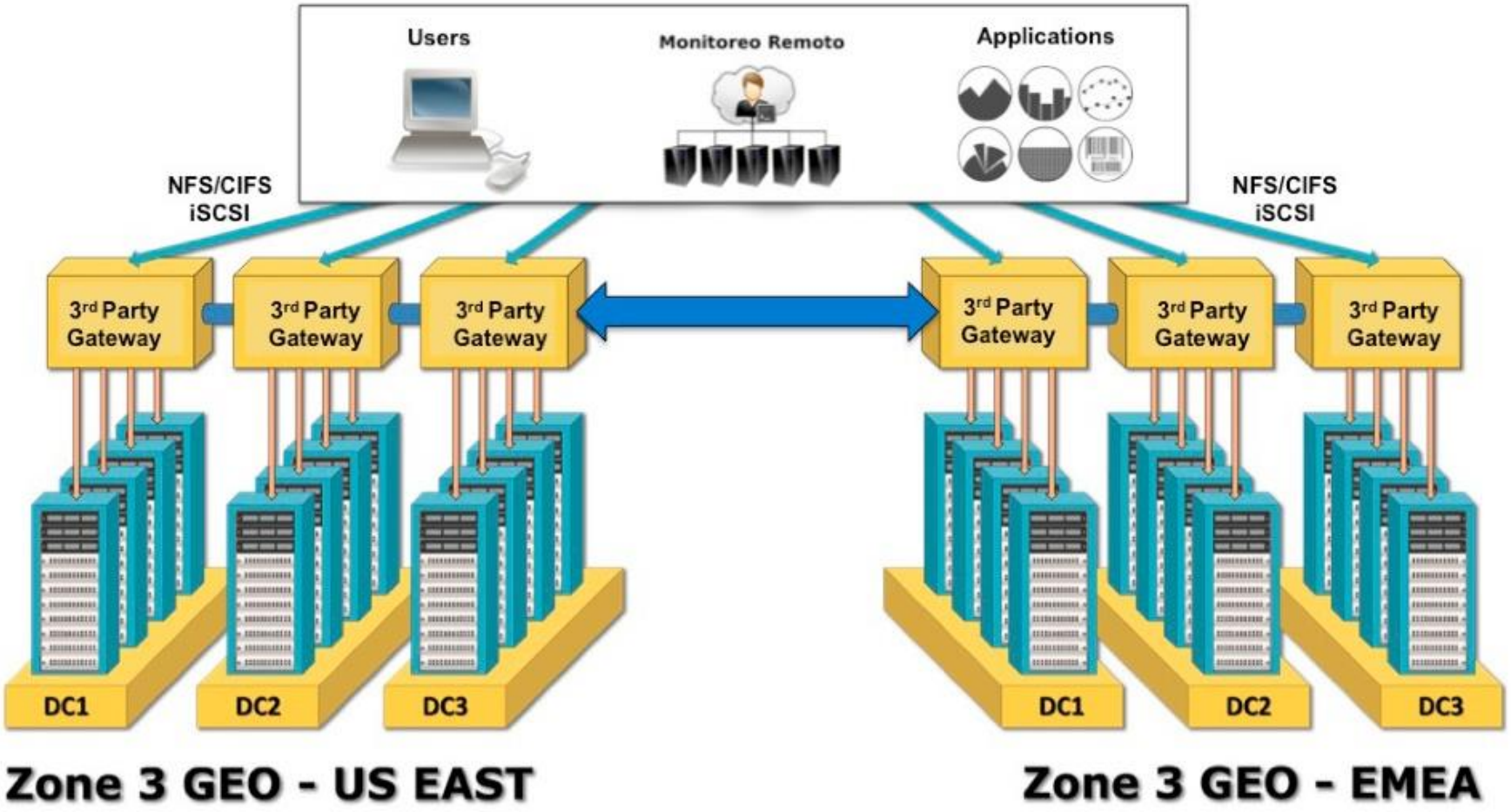
storage = computar does it !!

Average engineer's view of storage



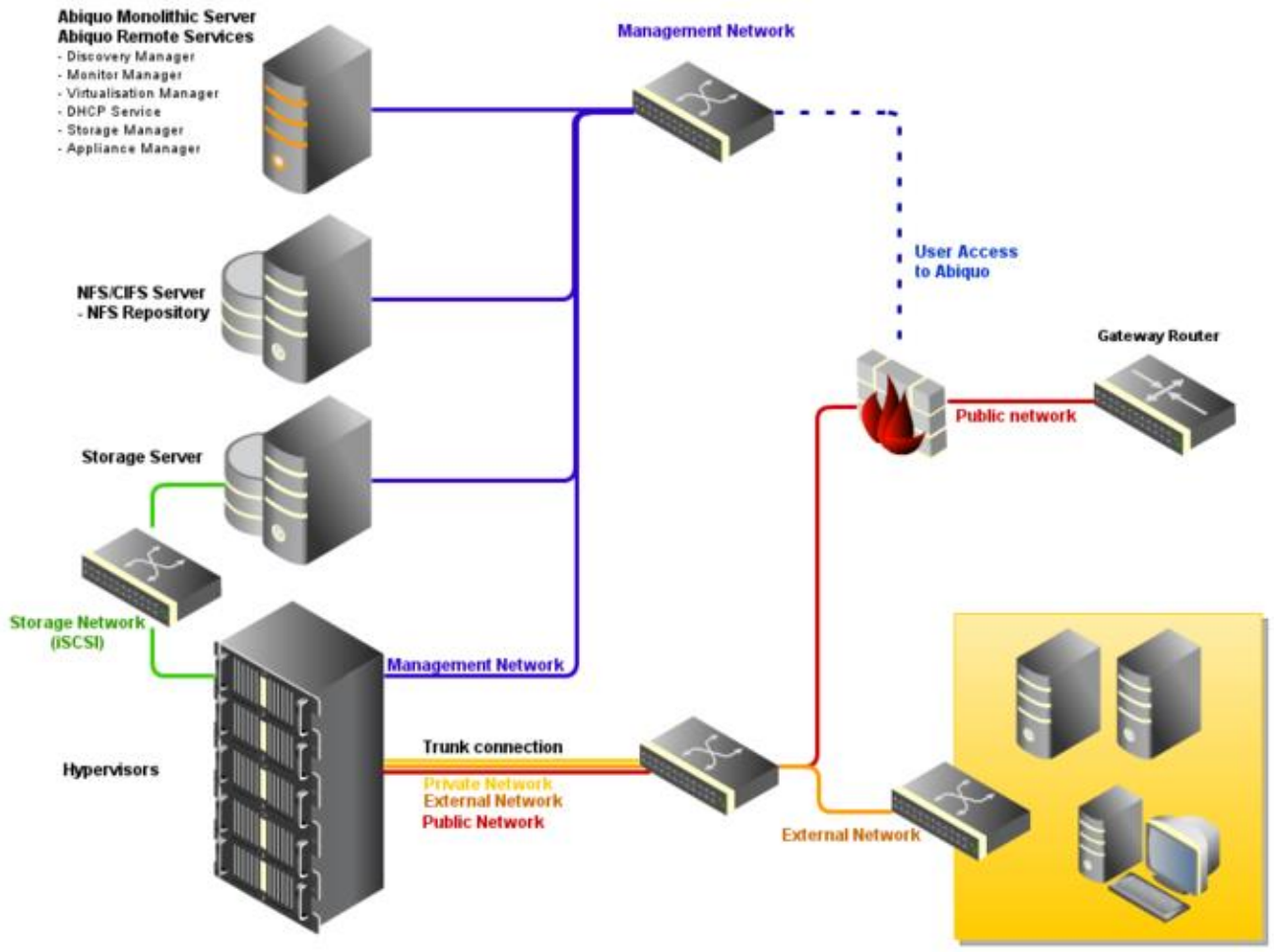
storage = thing inside computar!!

A few enterprise storage architectures (1)



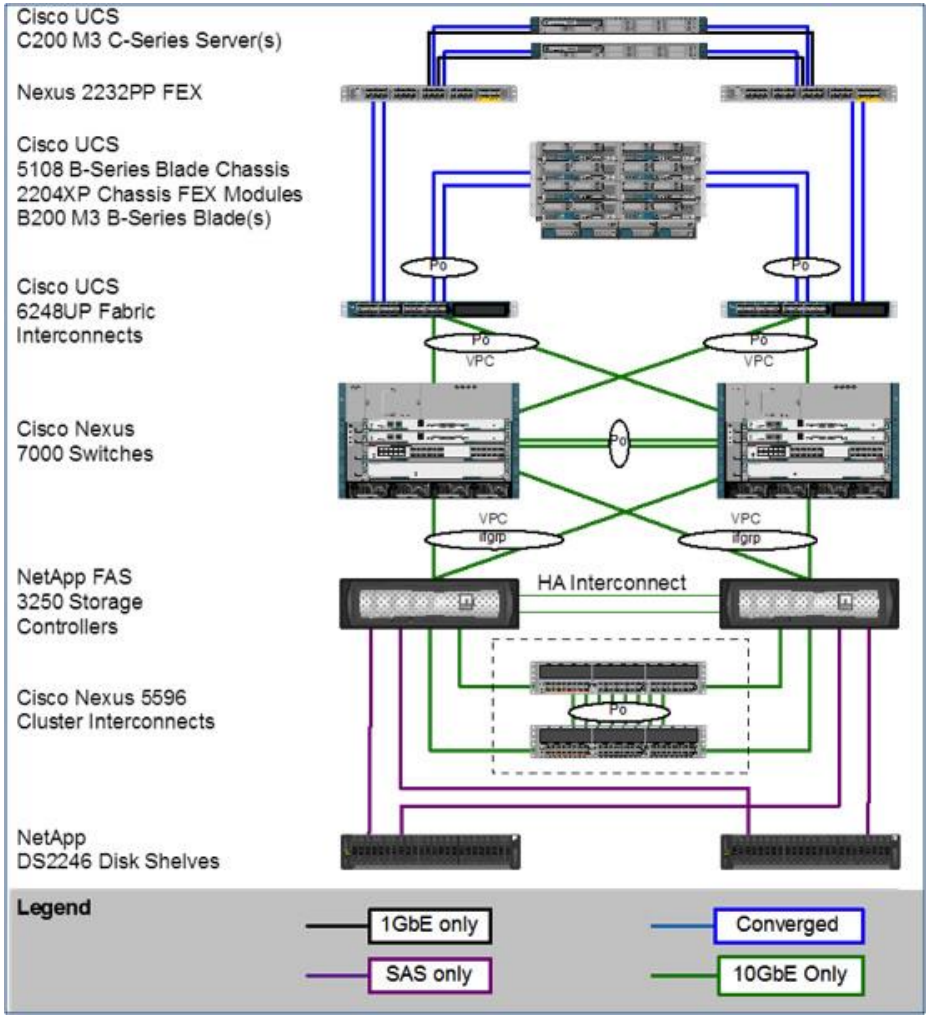
- From: http://www.storagenewsletter.com/rubriques/software/massively-scalable-himalaya-architecture-by-amplidata/

A few enterprise storage architectures (2)



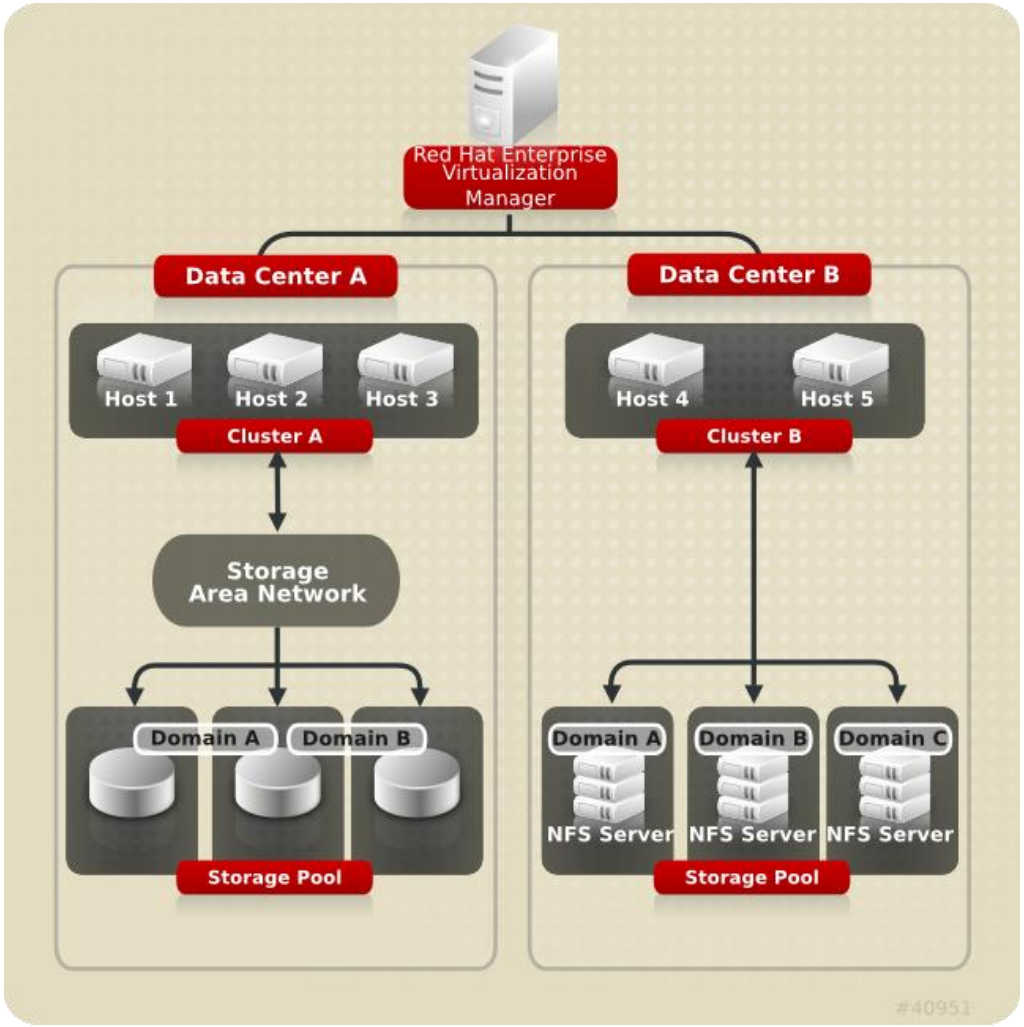
- From: http://wiki.abiquo.com/display/ABI20/Monolithic+Architecture

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?

- **Capacity**: Can it hold enough?
- **Performance**: Is it fast enough?
- **Cost**: Is it cheap 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?

- **Capacity**: Can it hold enough?
- **Performance**: Is it fast enough?
- **Cost**: Is it cheap 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?

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

Instructor and TAs

- Professor: Tyler Bletsch
  - Office: Hudson Hall 106
  - Email: Tyler.Bletsch@duke.edu
  - Office Hours: TBD

- TA:
  - Andrew Stevens (andrew.j.stevens@duke.edu)

Getting Info

- Course Web Page: static info

http://people.duke.edu/~tkb13/courses/ece590/

  - 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!

- 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

	Assignment	%
Project: 50%	Project proposal	5%
	Project outline	5%
	Project milestone presentation	5%
	Project final presentation	15%
	Project demo	20%
	Homework	30%
	Final exam	20%

The Project

- **Proposal**: Group up and say what you're going to do.
  - Write-up plus 30-minute meeting scheduled out of class.
- **Outline**: Add detail. Say how you're going to do it.
  - Write-up plus 60-minute meeting scheduled out of class.
- **Milestone presentation**: Present work done so far to class.
  - 5-minute talk in class.
- **Final presentation**: Present complete project to class.
  - 15-minute talk in class.
- **Final demo**: Defend your project to the instructor.
  - 60+ minute meeting scheduled out of class.

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

Homework

- Homework assignments – **done individually**
- Partial credit is available – provide detail in your answers to seek it!
- Late homework submissions 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.

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
  - Homework is individual – you do your own work
  - 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
  - 8 students were busted on Homework #1 in spring 2013, and 2 of them were referred to the Office of Student Conduct
- "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**