THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

The End

CSCI 2541 Database Systems & Team Projects

Wood & Chaufournier

Today

Demo / Report details

A few more "hot" topics

Course summary

Demo

We will release blocks of appointments over the next few days

Each team can sign up by entering their team number

If you want to change your time after signup, you must get my approval (slack or email)

Report

Project Pitch: ~2 paragraphs describing your overall project to someone who knows nothing about it. Include a representative screenshot

Special Features: ~2 sentences describing each extra feature you added beyond the spec

DB Schema: Show schema for 2-3 tables and justify your design. You should focus on the most important/interesting aspects (2-3 paragraphs)

Work Breakdown: List teammates and specify the aspects of the project they worked on

Submit as a file: **report.md** (make Issue if you already demoed)

Due by 3pm Tuesday May 11th (earlier if possible!)

or demo time

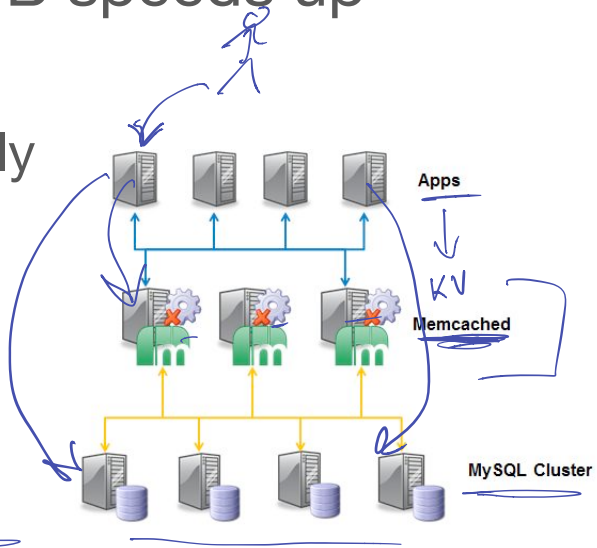
Hot Topics:
Caches, Clouds, Containers

Caches

Previously we said that RDBMS don't scale well and can get slow when size is very large

Adding a **cache** in front of the DB speeds up accesses

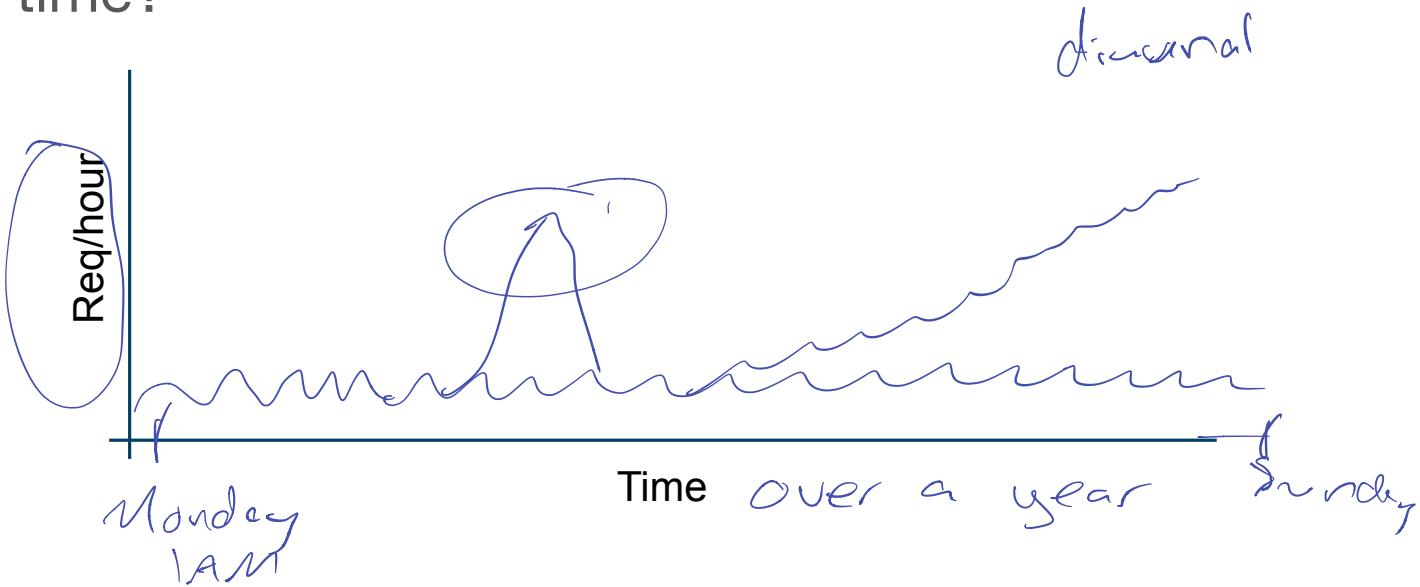
- Smaller, faster source of frequently accessed data



What challenges does this add?

Web traffic behavior

What would the traffic to an online book store look like over time?

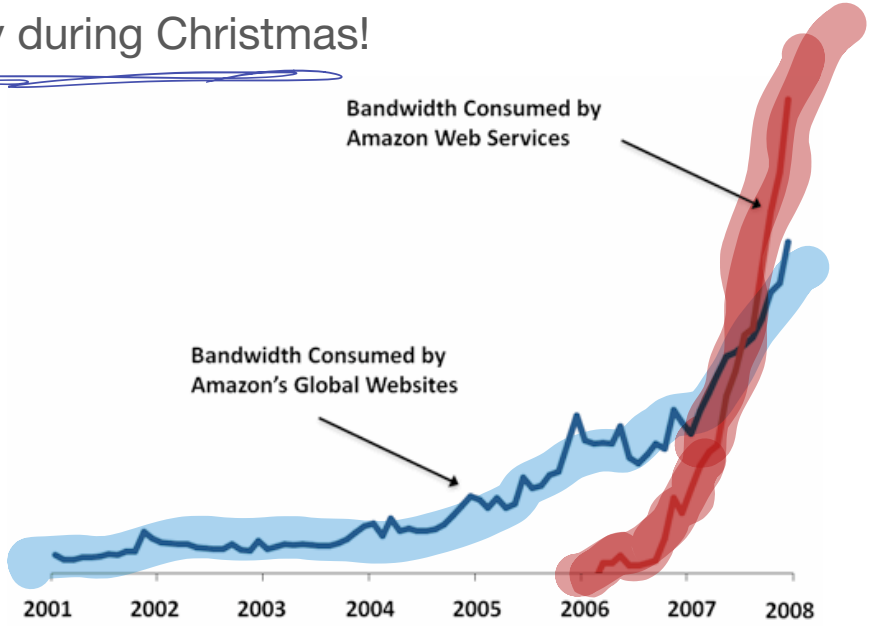


Amazon's Cloud

Amazon built its cloud platform so that other people could pay for its infrastructure during the rest of the year…

- Only needed peak capacity during Christmas!

Now its cloud users are far bigger than its own sites



Scale Estimates

1.5-2 million servers - Bloomberg 2014

50-80K per data center, 68 total data centers
= 3.4-5.4 million

- re:Invent 2016

*Every day Amazon adds as many servers as it had in **2000** (when it was a **$2 billion** company)*
— talk at UW **2011**

*Every day Amazon adds as many servers as it had in **2005** (when it was a **$8.5 billion** company)*
— AWS re:Invent **2016**

Why use the cloud?

Pay-as-you go

Expand quickly on demand

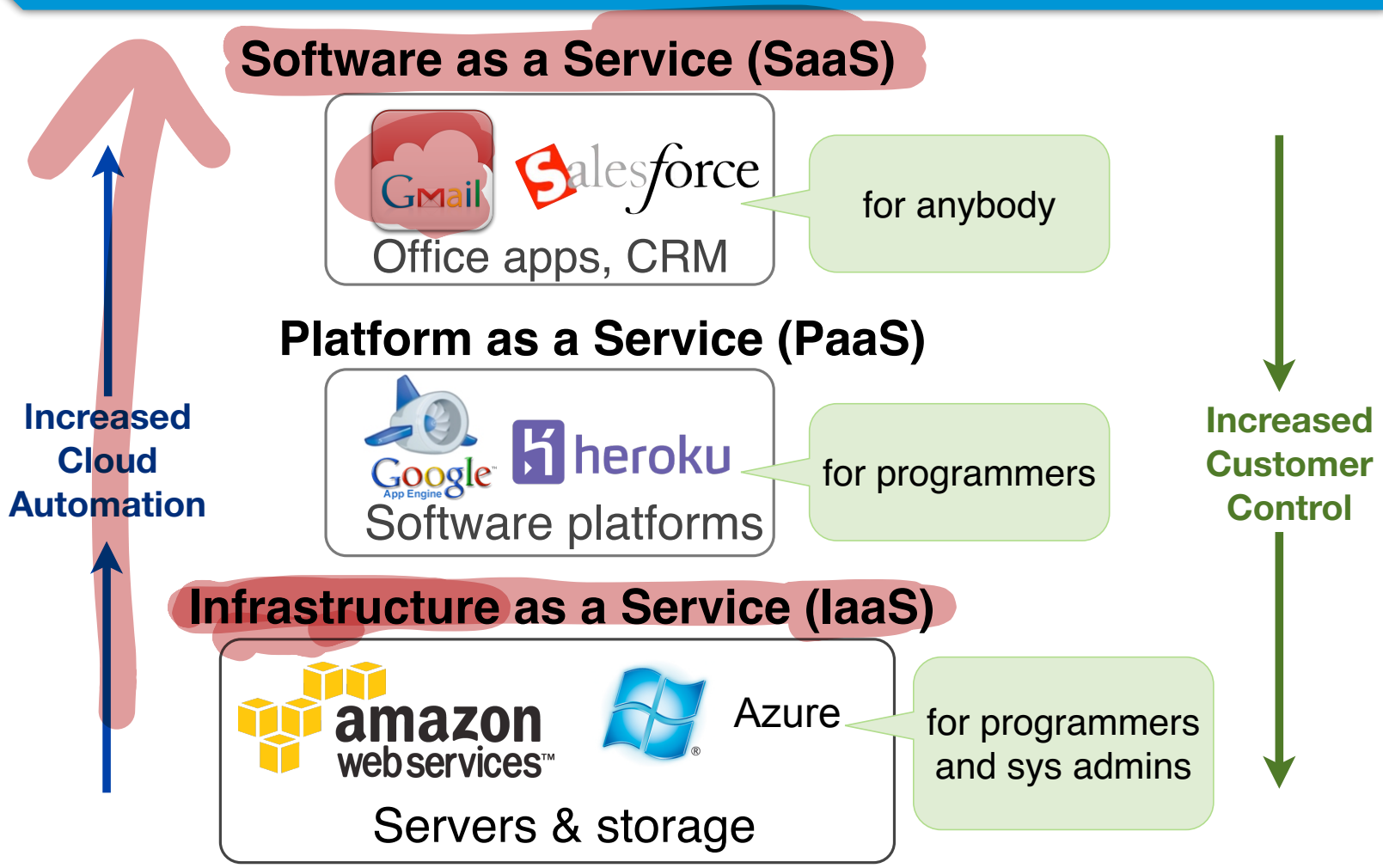
Don't need to worry about (many) IT issues

Cheap! (?)

... but is the cloud perfect?

[spoiler alert] no.

Types of Clouds



Types of Cloud Services

Infrastructure as a Service (**IaaS**)

- Rent VMs, Containers, physical servers, disks, etc. by the hour
- Pricing: $ per hour
- Examples: EC2, EBS, S3

Platform as a Service (**PaaS**)

- Cloud provides a software layer on top of its resources
- Exposes a programming API for users to develop cloud-based apps
- Cloud provider manages all underlying resources (autoscaling)
- Pricing: $ per hour *if service is getting requests*
- Examples: Beanstalk, Lambda, EMR, Heroku

Serverless / FaaS

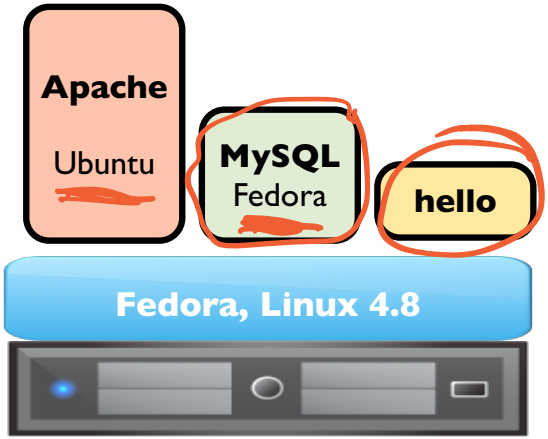
Functions as a Service (**FaaS**)

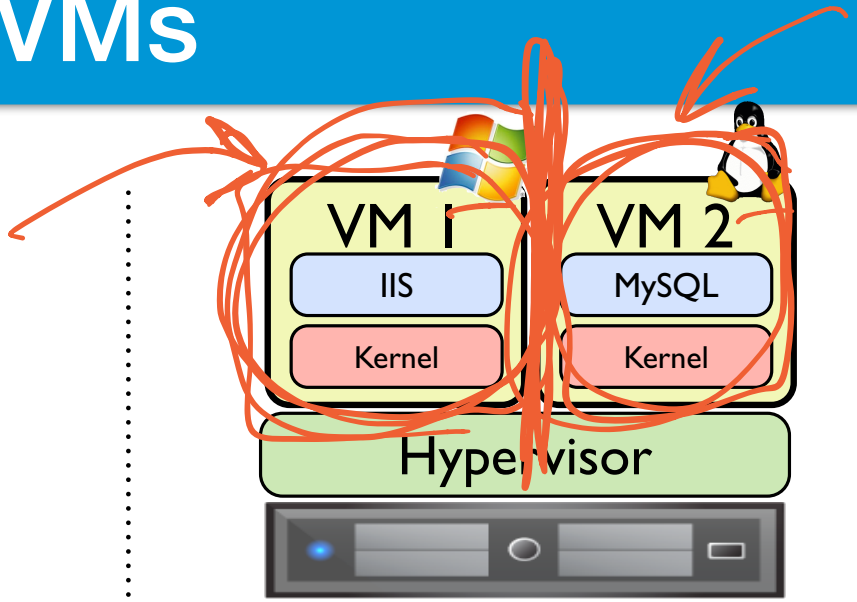
- Newer, trendier version of PaaS
- You provide a function or executable with HTTP/REST interface
- Cloud automatically deploys and autoscales VMs or containers as workload changes
- Pricing: $ *per request*
- Examples: Lambda, Google Cloud Functions, Azure Functions

PaaS and FaaS generally require **stateless** functions

- All important data must be sent to DB!

Container vs VMs





Containers

- A group of applications, files, and resources
- Share the underlying OS kernel
- Lightweight

Virtual Machines

- A full virtual computer including OS, applications, files, and resources
- Strong isolation between VMs

Process Isolation

Processes

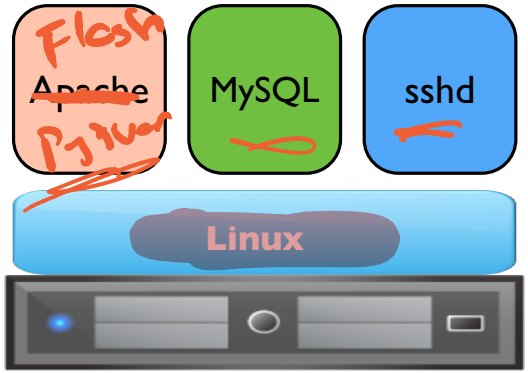
- OS provides isolation

Isolated:

- Memory

Shared:

- File system
- Network
- Devices
- OS Kernel

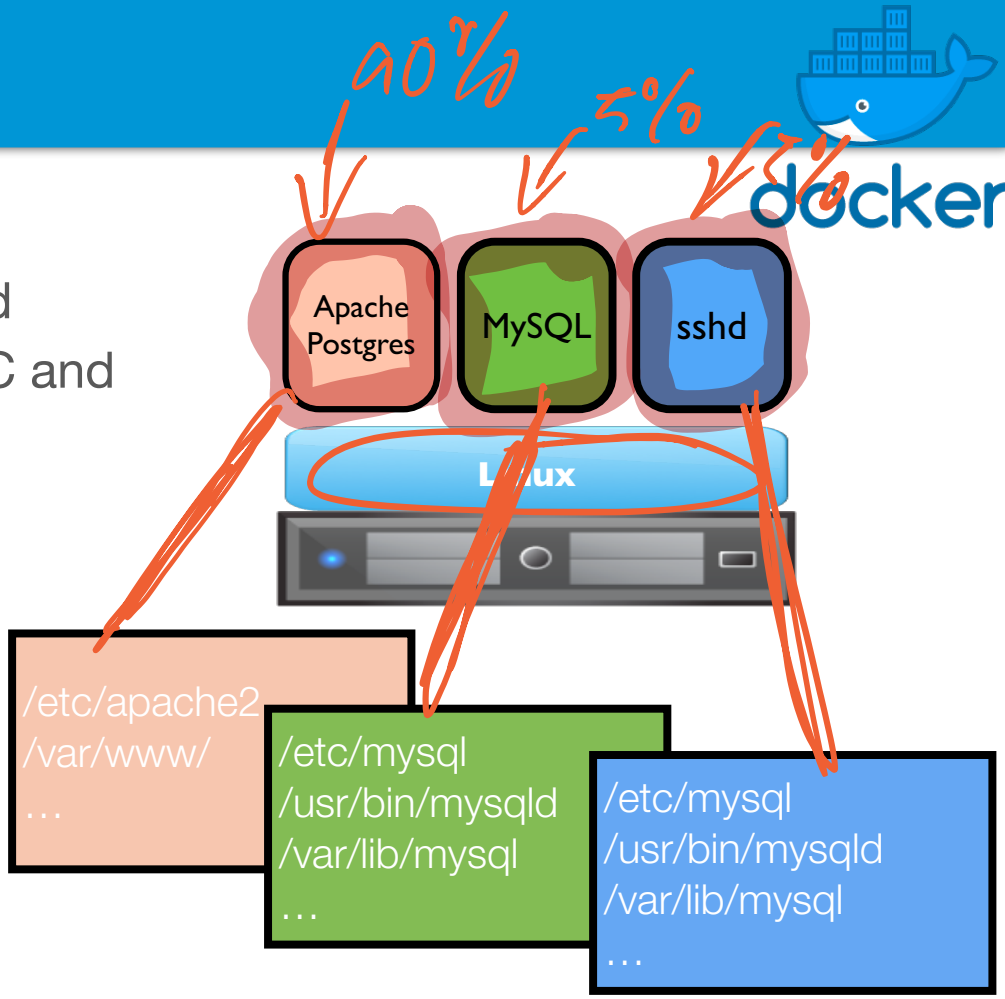


/etc/
/etc/apache2
/etc/sshd.conf
/etc/mysql
/usr/bin/mysqld
…

Containers

Containers

- Namespace-based isolation using LXC and cgroups



Isolated:

- Memory
- File system
- Network
- Devices

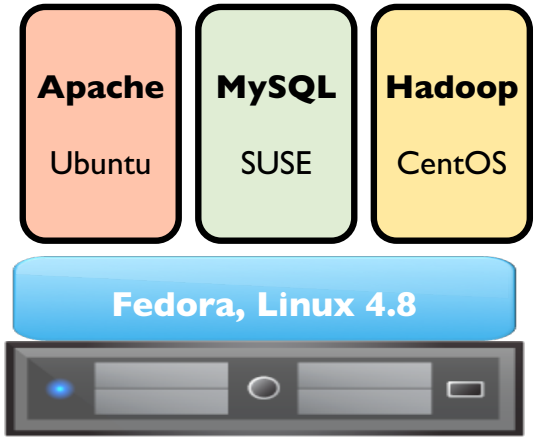
Shared:

- OS Kernel

Container Packaging

Deployment - big benefit of containers/virtualization

- Lets you package up an application and all of its requirements
- Even the distribution and 3rd party utilities!
- Very helpful for system administrators

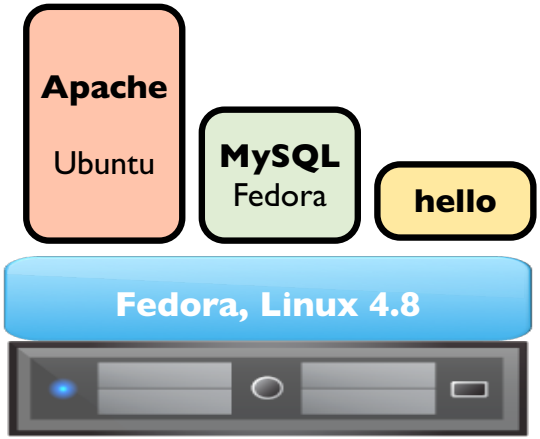


Container “image” includes:

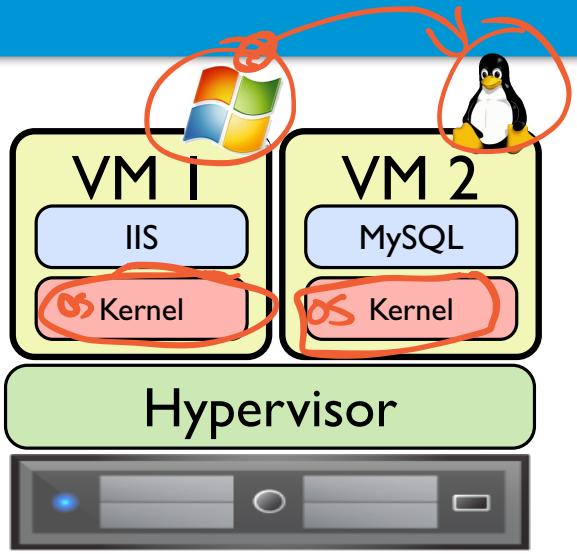
- Linux distribution base files
- Dependency libs/utils
- Configuration files
- Application to run

Does not include…?

Container vs VMs

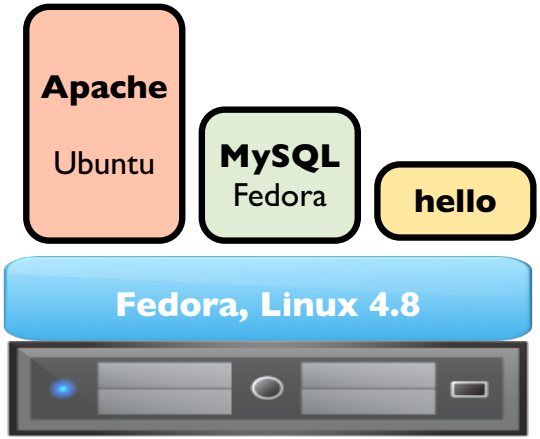


Pros:
- Light weight isolation
- Packaging / deployment



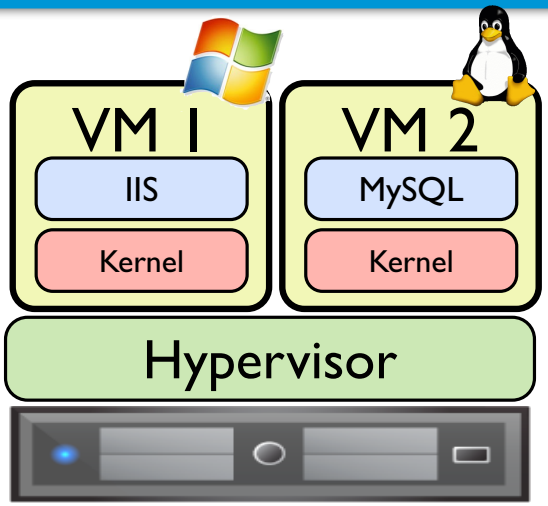
Pros:
- Stronger isolation
- Multiple OS

Container vs VMs



Pros:

- lightweight (no duplication)
- less resource consumption
- easier to deploy
- specify resources just for application
- faster startup time



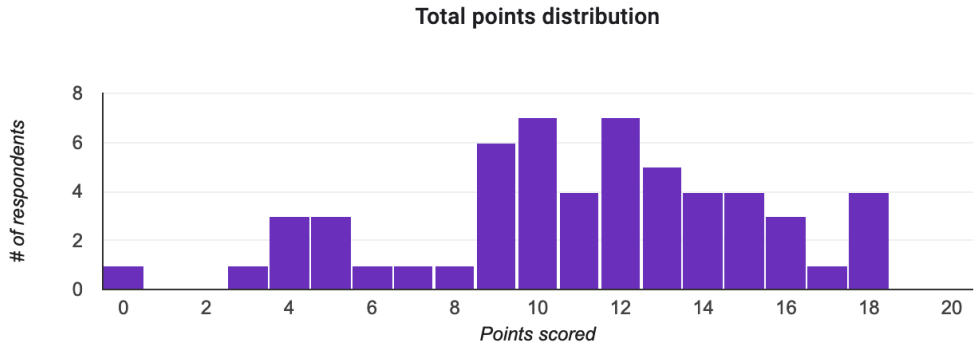
Pros:

- stronger isolation
- different kernel versions/ OSes
- fault tolerance / isolation
- combine with containers

No Grade Quiz

Let's discuss answers

Average	Median	Range
11.09 / 20 points	11 / 20 points	0 - 18 points



What ~~is~~ was this course?

Database systems design and implementation

- Theory of relational database design and query languages
  - Relational Model, Relational algebra, SQL
- Application development using Relational DBMS (MySQL), with ~~PHP~~ Python web apps

Intro to database models for unstructured data (Big data)

- Overview of NoSQL database models

but wait there's more!

What is this course?

Database systems design and implementation

- Theory of relational database design and query languages
  - Relational Model, Relational algebra, SQL
- Application development using Relational DBMS (MySQL), with ~~PHP~~ Python web apps

Intro to database models for unstructured data (Big data)

- Overview of NoSQL database models

Database System Project: Full stack development

Teamwork – SW development in teams

- Project (SW) integration

Improving technical communication skills:

- Writing in the disciplines (WID)* in tandem with CS2501

*Course is not just about Database design – you have to learn and participate in the other two course objectives (WID, Team SW).

What is this course?

One of the most **useful** and **applicable** courses you will take while at GW!

(I hope)