



# COEN 241

# Introduction to Cloud Computing

Lecture 1 - Introduction & Course Overview





# Agenda for Today

- Welcome!
- Cloud Computing Primer
- Course Overview
  - Course Objectives
  - Course Structure
  - TODOs
  - Logistics
  - Instructor Information
- Readings
  - Recommended: CCSA 1.1 - 1.3
  - Optional: CCSA 1.5 - 1.15





# Welcome to COEN 241!

- About the instructor
- Let's introduce ourselves!
  - Name
  - Department
  - Degree of study
  - Full/Part-Time
  - What do you expect to learn from this class?



# What is Cloud Computing?





# What is Cloud Computing?

**Cloud computing** is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., server, network, storage & applications) that can be rapidly provisioned and released with minimal management effort or service provider interaction.\*



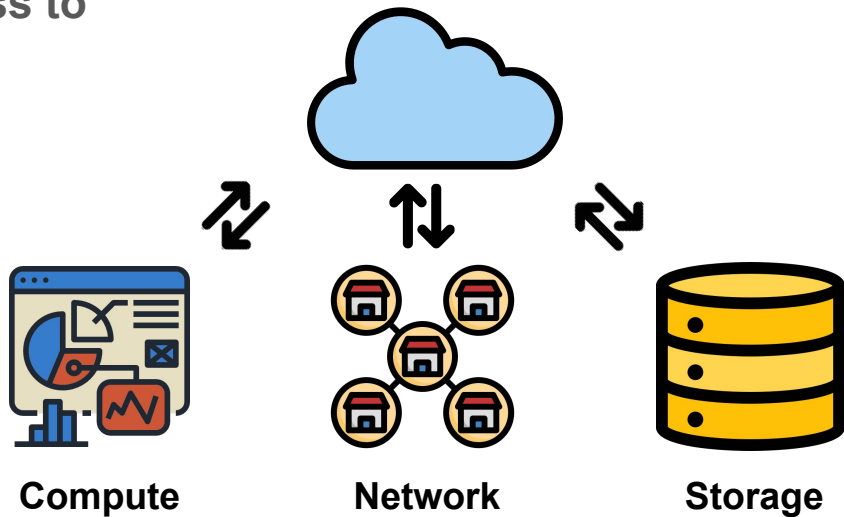
# What is Cloud Computing?

- Cloud computing is when you access computing services—like servers, storage, networking, software—over the internet (“the cloud”) from a cloud provider. - Microsoft
- Cloud computing is the on-demand delivery of IT resources over the Internet with pay-as-you-go pricing. Instead of buying, owning, and maintaining physical data centers and servers, you can access technology services, such as computing power, storage, and databases, on an as-needed basis from a cloud provider. - AWS



# What is Cloud Computing?

On demand access to



over the internet

# What is Cloud Computing?



Clients



Internet



Data Centers





# Where did “Cloud Computing” come from?

- Cloud symbol used in diagrams to symbolize the Internet since 1960s.
- The first reported public use of the phrase was in August of 2006 by Eric Schmidt (Ex-Google CEO)



# Cloud Computing Enables:

- The illusion of “Ubiquitous” & infinite “Shared Pool” of computing resources available “On-Demand”.
  - Eliminates the need to plan far ahead for provisioning.
- Rapidly and automatically provisioned computing resources.
- Minimal hardware / infrastructure management with increased reliability.

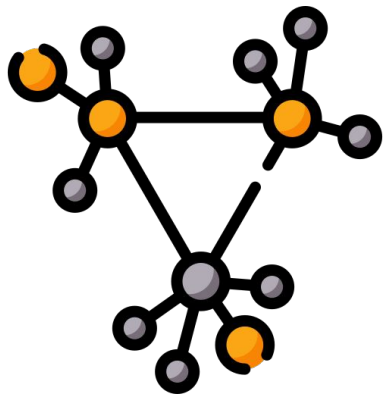


# Cloud Computing Enables:

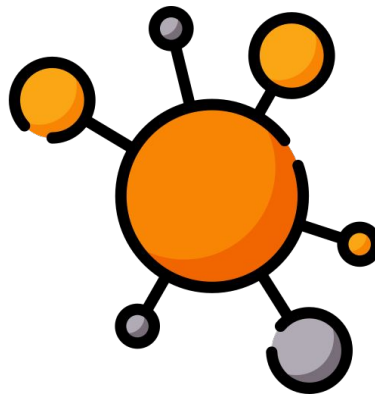
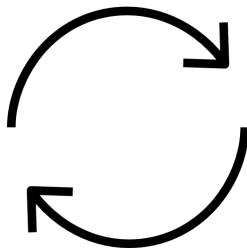
- Access to need-based performance.
- The elimination of an up-front commitment by Cloud users.
  - Allows companies to start small and expand only when needed.
- The ability to pay for use of computing resources on a short-term basis as needed (e.g., processors by the hour and storage by the day)
  - Rewards freeing resources when they are no longer useful.



# Why Cloud Computing?



**Decentralized**

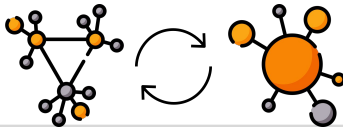


**Centralized**

\*<https://www.geeksforgeeks.org/comparison-centralized-decentralized-and-distributed-systems/>

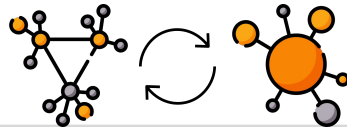
# Why Cloud Computing Now?

- 1950 ~ 1970s (Centralized)
  - Huge computers, shared by multiple users
  - Hard for people to own a computer own their own
- 1980s ~ 2000s (Decentralized)
  - People started owning their personal machines (Personal Computers)
  - PC speed was doubling every two years (Moore's Law)
  - Less need to share a large computer unless it is for specific purposes

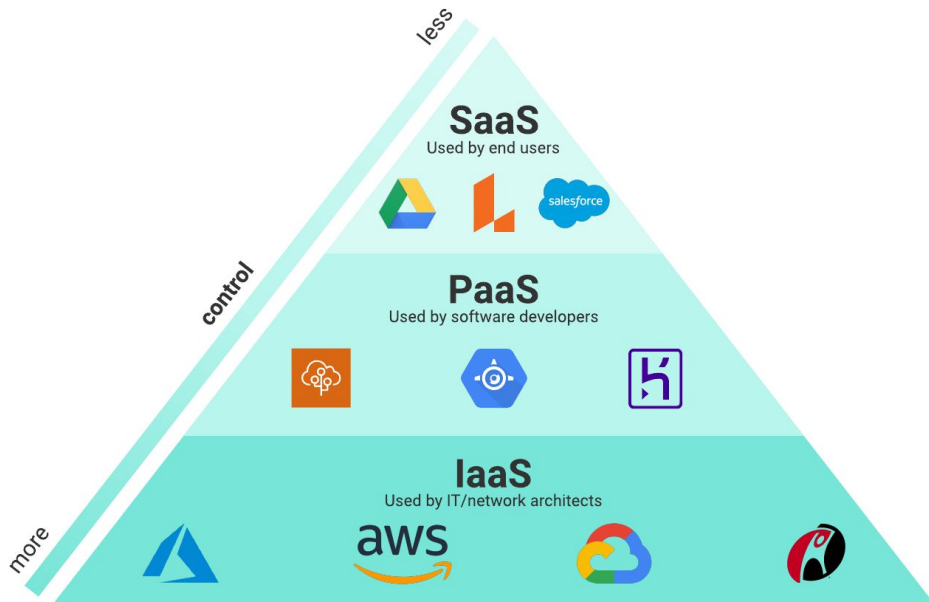


# Why Cloud Computing Now?

- 2000s ~ Now (Centralized Cloud)
  - Moore's Law is ending: End of Scale-up, Start of Scale-out
  - People are more mobile than ever
  - Applications need global availability
  - Hard to scale fast when maintaining infrastructure

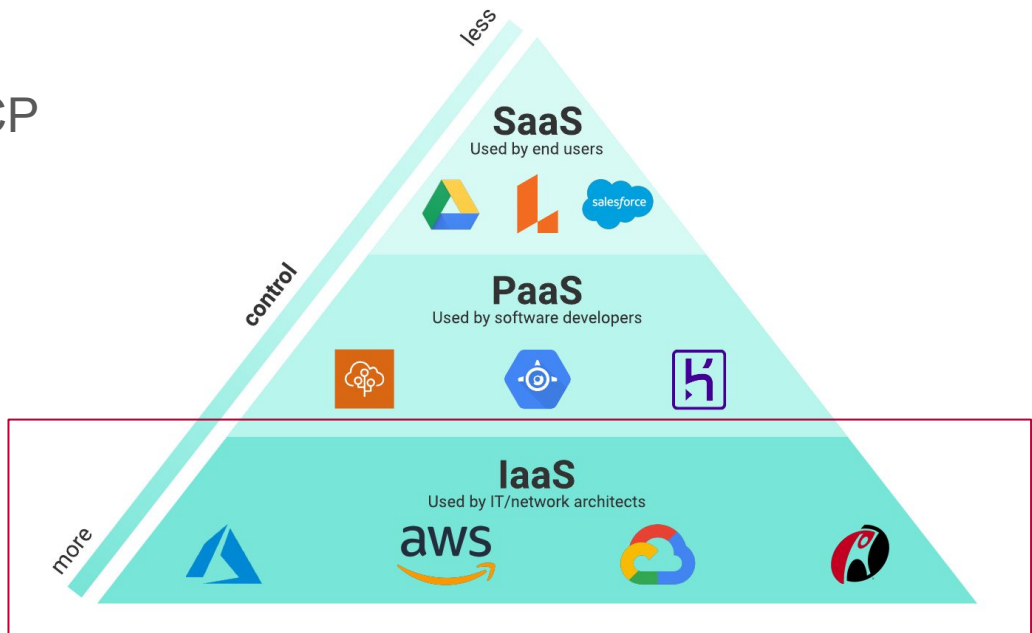


# Cloud Service Models



# laaS : Infrastructure as a Service

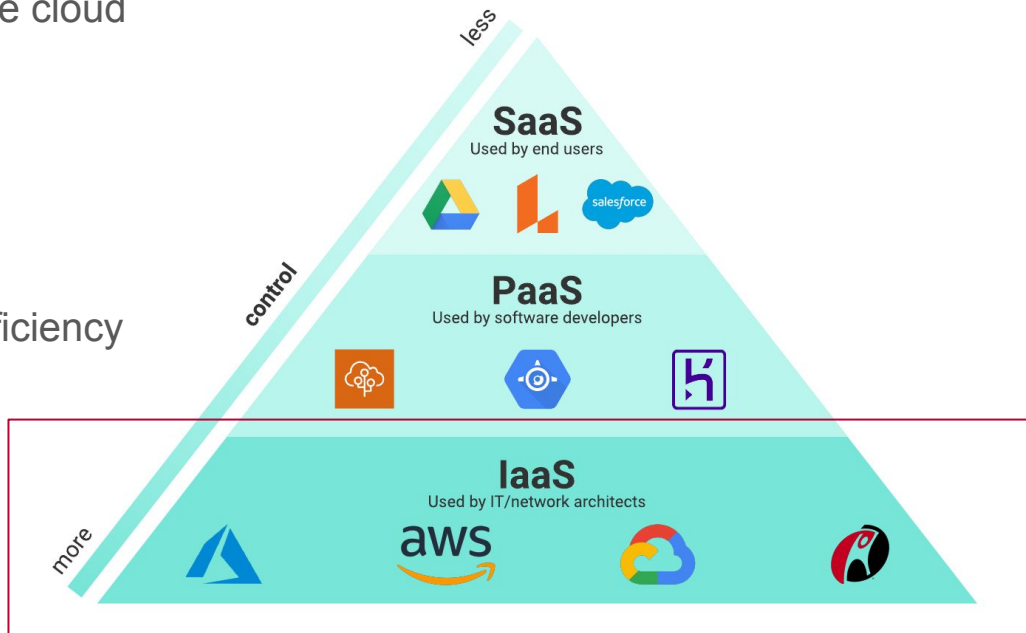
- Provides capabilities to provision computing resources
- Examples: AWS, Azure, GCP
- Users:
  - Network Architect
  - ITs
  - Enterprises





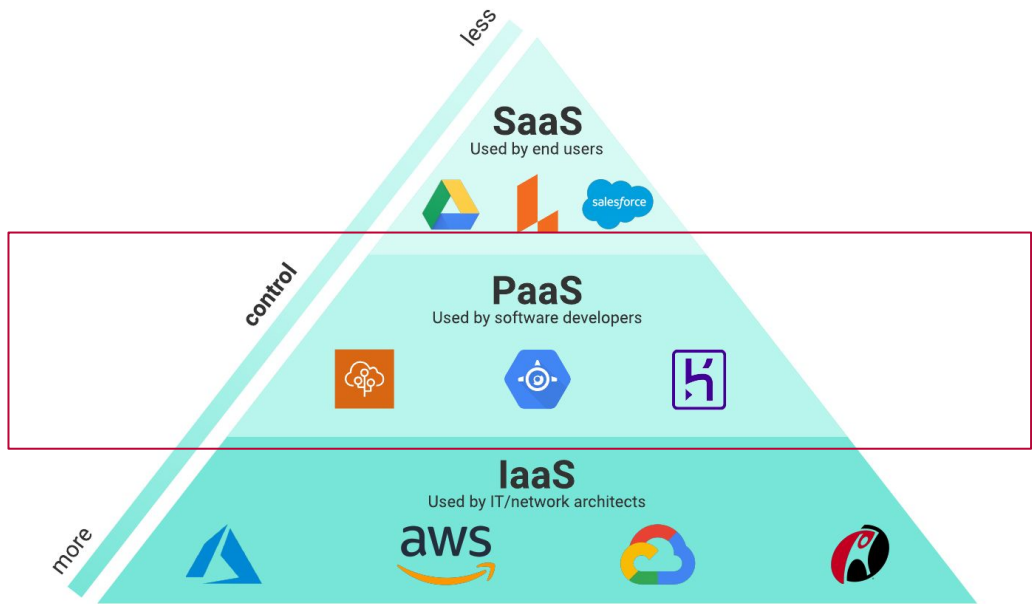
# IaaS : Infrastructure as a Service

- Pros:
  - Flexible cost
  - Less maintenance vs private cloud
  - Guaranteed performance
  - Scalable
- Cons:
  - Higher learning curve
  - Hard to optimize for cost efficiency
  - Cloud Lock-in



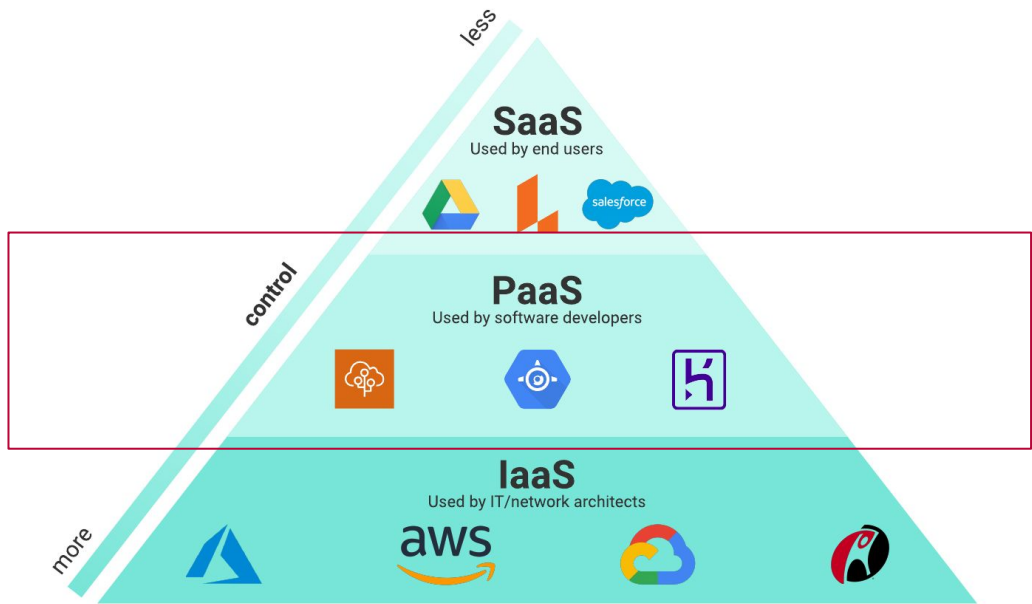
# PaaS : Platform as a Service

- Provides ability to develop and deploy applications on the cloud
- Examples
  - Google App Engine
  - Heroku
  - Github
- Users:
  - Software Developers
  - Large Enterprises



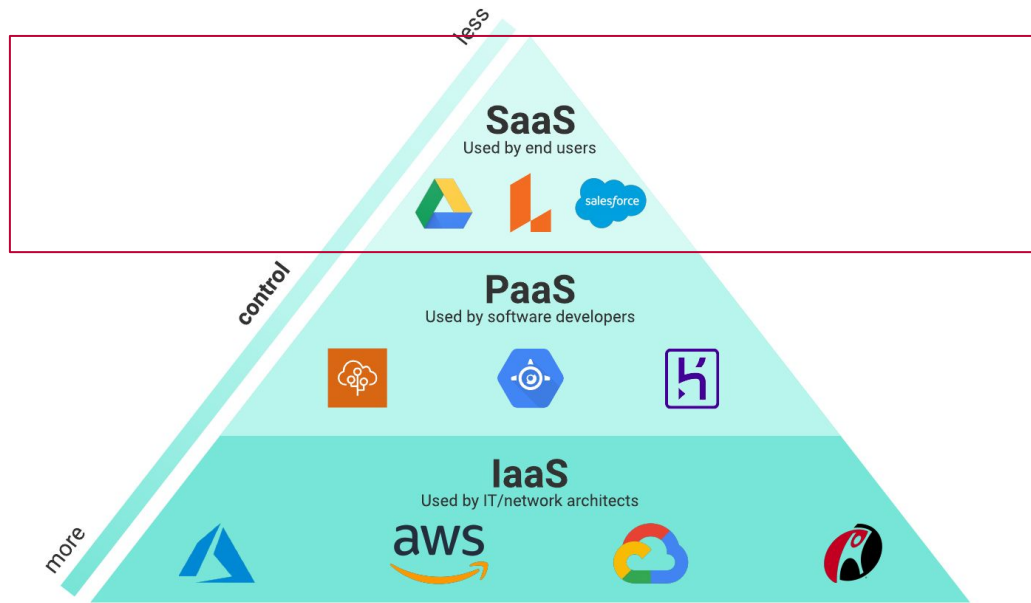
# PaaS : Platform as a Service

- Pros:
  - Lower learning curve
  - Less operational cost
  - Improved scalability
- Cons:
  - Costs more than IaaS
  - Platform Lock-In



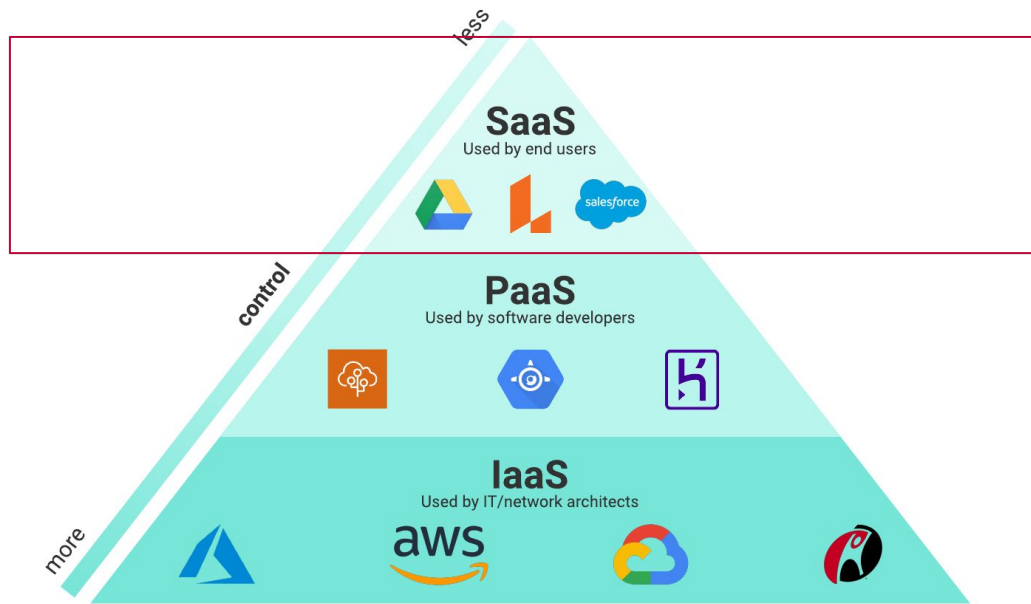
# SaaS : Software as a Service

- Provides a complete application environment
- Examples
  - Google Apps
  - Salesforce
  - Dropbox
- Users:
  - End users



# SaaS : Software as a Service

- Pros:
  - Lowest learning curve
  - Easy upgrade
  - Secure
  - Fast and High adoption
- Cons:
  - Costs the most
  - Lack of transparency



# Cloud Deployment Models



**PUBLIC  
CLOUD**



**PRIVATE  
CLOUD**



**COMMUNITY  
CLOUD**



**HYBRID  
CLOUD**

# Cloud Deployment Models

- **Public**
  - Infrastructure built operated by cloud providers
  - Anyone can provision and use
- **Private**
  - Infrastructure provisioned for a single consumer
- **Community**
  - Cloud shared by organizations that have similar policy and compliance needs
- **Hybrid**
  - Two or more cloud models combined



## Cloud Use Cases #1 : Netflix on AWS

Netflix operates “many tens of thousands of servers and many tens of petabytes of storage” in the Amazon cloud\*



\*<https://aws.amazon.com/solutions/case-studies/netflix/>



# Cloud Use Case #1 : Netflix on AWS

- Runs **Everything** on AWS, Netflix shuts down its own data centers
  - Netflix only focuses on its business logic
- ~100k virtual machine instances running at peak time
  - Dynamic environment
  - Cost saving (pay-as-you-go)
- ~500 microservices running on Containers
  - Allows for agile development





# Cloud Use Case #1 : Netflix on AWS

- Many companies are moving to the public cloud
- Enables rapid development with less management overhead
  - Availability
  - Reliability
  - Scalability
- Not necessarily cost saving or the most performant



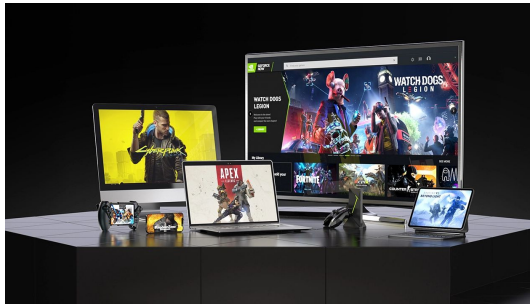
## Cloud Use Case #2 : Dropbox 'off' AWS

- Some companies are moving away from the 'public' cloud as well
- Started off with Hybrid Cloud, but needed more performance and cost efficiency
- Now running their own private cloud



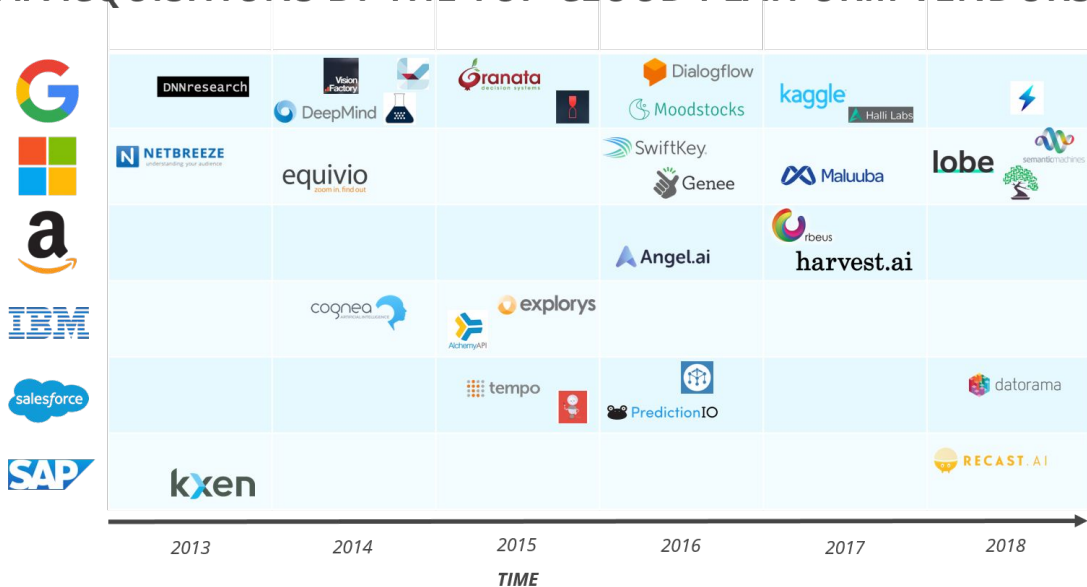
## Cloud Use Case #3 : Cloud Gaming

- Stream games from powerful servers in the cloud
- Clients no longer need powerful machines
- Just need a fast and quick network connection
- VR, Self-Driving



# Cloud Use Case #4 : Data Analytics

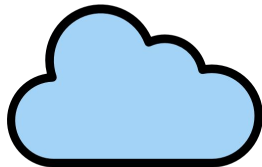
## AI ACQUISITIONS BY THE TOP CLOUD PLATFORM VENDORS



# What does it mean to “Learn” cloud computing?



Clients



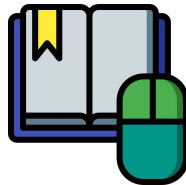
Internet



Data Centers

# Course Objectives

- Introduce the following concepts:
  - Cloud architecture and service models
  - **Technologies that enable cloud computing**
  - Technologies that use cloud computing (e.g., MapReduce, Databases)
- Provide hands-on experience on building / using cloud resources
  - Building an (open-ended) application using cloud framework
  - Be creative!



# What does it mean to “Learn” cloud computing?

- What technologies in the “data centers” enable cloud computing?
  - Virtualization, Containers, Serverless
  - Software-Defined Networking
  - Databases
  - More...
- How to optimize data centers for
  - Colocation
  - Various Applications
- How does cloud-specific applications work?
  - Mapreduce
  - Many many More...





# What This Course Will NOT Teach

- Programming & Languages
  - No preference on languages and IDEs
- How to use different type of OS or shells
  - Prefer to use Linux (e.g., Ubuntu) for assignments
- Algorithms & Data Structures
- How to use commands from a specific cloud provider
  - We won't cover AWS, GCP, or Azure specific contents



# Word of Caution!

- Cloud computing is not about just spinning VMs and containers up & down in AWS or GCP or Azure
- First part of the course will be very system & OS heavy!
  - Check if you have enough OS knowledge!
- Second part of the course will have lot of networking material
  - Check if you have enough Networking knowledge!



# Course Topics

1. Cloud Computing Introduction
2. Virtualization Overview: Virtual Machines and Hypervisors
3. Containerization: Docker and Kubernetes
4. Serverless Computing
5. Microservices and Orchestration
6. Availability, Reliability and Scalability



## Course Topics

6. Computer Networks 101
7. Software Defined Networks & Network Virtualization
8. Storage Virtualization
9. Distributed Databases / NoSQL
10. MapReduce and Spark/Hadoop
11. Consistency (Zookeeper, Raft)
12. Sustainable Cloud



# Course Structure & Grading Breakdown

- 2 Lectures per week (5% attendance & participation)
  - Hybrid (30%)
  - Must participate in lectures and discussion to get full credit
- 1 Midterm Exam (25%)
  - Will be on Zoom, remotely
  - **Must get a C or higher** to get a B or better grade in the class!
- 3 Assignments (**not including HW 0**) (15%)
  - 7 late days to use
- 6 Quizzes (15%)
- Final Team Project (40%)
  - We will talk more about this next class
  - **Must get a C or higher** to get a B or better grade in the class!





# Course Logistics

- Course Schedule
  - Mon, Wed 5:10~7:00PM
  - Hybrid Fashion (~30% remote)
- Course Website
  - On Camino
  - Syllabus!
- All class related contacts via Piazza (link in Camino)
  - Access code: **COEN241Wi2023**
- Instructor: Prof. Sean Choi
  - Office Hours: Mon 3-5PM, Heafey 211 and on Zoom
- Grader: Lakshmi Naarayanan Vaigai Shrinivasan



# Late & Absence Policy

- Assignments
  - 7 late days to use at a daily increment (no hour or minute usage)
  - 20% deduction per day late after the 7 late days have been spent
- Project & Quiz & Exams
  - **No late submission for any** parts of the project, presentations, quiz or exams
- Lectures
  - Up to 2 lectures can be missed
  - Deduction of the **attendance** grade after the first two absence





# About the Final Project

- A team project of 3~4 people with two possible directions
- Build an application on the cloud using the techniques we will learn
- Study, analyze, and report an existing cloud computing technique/system
- Students are strongly encouraged to explore problems of existing techniques, and possible solutions during this project
- More details will come as the course progress





# Course Textbook

- Cloud Computing Solutions Architect (CCSA) by Bahga and Madiseti
- Textbook has substantial AWS information (Unlike this course)
  - We will not cover any tools/commands for a specific cloud provider
- Recommended for additional high-level information
- Optional Readings
  - OS Three easy pieces: <https://pages.cs.wisc.edu/~remzi/OSTEP/>
  - Each lecture will have relevant optional reading materials



# TODOs

- Please participate in the survey in the following link  
<https://forms.gle/SLJxWsMixWE7JujG6>
- Sign up for Piazza (Will be part of participation!)
- Syllabus Quiz!
- **FIND YOUR GROUPS! NO EXCEPTIONS!**
- Watch the following videos
  - Above the Clouds: A Berkeley View of Cloud Computing
    - <https://www.youtube.com/watch?v=IJCxgoh5ep4>
  - Migrating to Cloud — Lessons from Netflix, Brought Up to Date
    - <https://www.youtube.com/watch?v=XrWII4ewrXA>
  - Large-scale Data Centers
    - <https://www.youtube.com/watch?v=r97qdyQtlk>
- (Optional): Apply for cloud credits



# Agenda for Today

- Cloud Computing Primer
- Course Overview
  - Course Objectives
  - Course Structure
  - TODOs
  - Logistics
  - Instructor Information
- Readings
  - Recommended: CCSA 1.1 - 1.3
  - Optional: CCSA 1.5 - 1.15





# Questions?

