# ECEN 757
# Spring 2022

*Lecture 2: Introduction to Cloud Computing*

# Announcement

- Log onto canvas this week


Google
App Engine
amazon
web services
RIGHTSCALE
ORACLE
GridGain
2.0
Elastic Server
VirtualBox
virtualLogix
CITRIX
vmware
force.com
platform as a service
OpenVZ
KVM
10gen
Microsoft
parascale
Powering Cloud Storage
hp
Xen
GIGASPACES
Parallels
Optimized Computing
RIGHTSCALE
elastra
enomaly
ORACLE
BUNGEEconnect
PowerVM
Virtualiron

# The Hype!

- Forbes: In 2015, worldwide cloud computing market grew 28% to $110 billion in revenue
- Gartner in 2019 - The worldwide public cloud services market is projected to grow 17.5 percent in 2019 to total $214.3 billion, up from $182.4 billion in 2018.
- Allied Market Research in 2021 - The global cloud services market was valued at $325,689 million in 2019, and is projected to reach $1,620,597 million by 2030, registering a CAGR of 15.8%.
  - "The COVID-19 outbreak has considerably boosted growth of the cloud services market"

# Many Cloud Providers

- AWS: Amazon Web Services
  - EC2: Elastic Compute Cloud
  - S3: Simple Storage Service
  - EBS: Elastic Block Storage
- Microsoft Azure
- Google Compute Engine
- Rightscale, Salesforce, EMC, Gigaspaces, 10gen, Datastax, Oracle, VMWare, Yahoo, Cloudera
- And many many more!

# Two Categories of Clouds

- Can be either a (i) public cloud, or (ii) private cloud
- Private clouds are accessible only to company employees
- Public clouds provide service to any paying customer:
  - Amazon S3 (Simple Storage Service): store arbitrary datasets, pay per GB-month stored
  - Amazon EC2 (Elastic Compute Cloud): upload and run arbitrary OS images, pay per CPU hour used
  - Google AppEngine/Compute Engine: develop applications within their appengine framework, upload data that will be imported into their format, and run

# Customers Save Time and $$$

- Dave Power, Associate Information Consultant at Eli Lilly and Company: “With AWS, Powers said, a new server can be up and running in **three minutes** (it used to take Eli Lilly **seven and a half weeks** to deploy a server internally) and a 64-node Linux cluster can be online in five minutes (compared with three months internally). … It's just shy of instantaneous.”
- Ingo Elfering, Vice President of Information Technology Strategy, GlaxoSmithKline: “With Online Services, we are able to reduce our IT operational costs by roughly **30%** of what we’re spending”
- Jim Swartz, CIO, Sybase: “At Sybase, a private cloud of virtual servers inside its datacenter has saved nearly **$US2 million annually** since 2006, Swartz says, because the company can share computing power and storage resources across servers.”
- 100s of startups in Silicon Valley can harness large computing resources without buying their own machines.

# But what exactly IS a cloud?

# What is a Cloud?

- It's a cluster!
- It's a supercomputer!
- It's a datastore!
- It's superman!

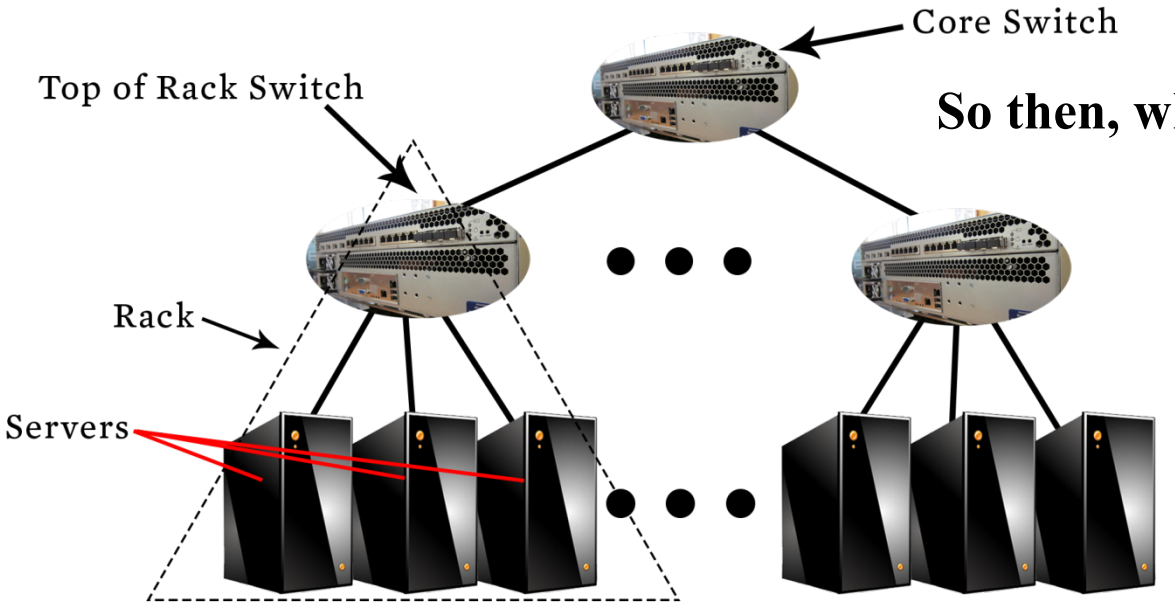
- None of the above
- All of the above

- Cloud = Lots of storage + compute cycles nearby

# What is a Cloud?

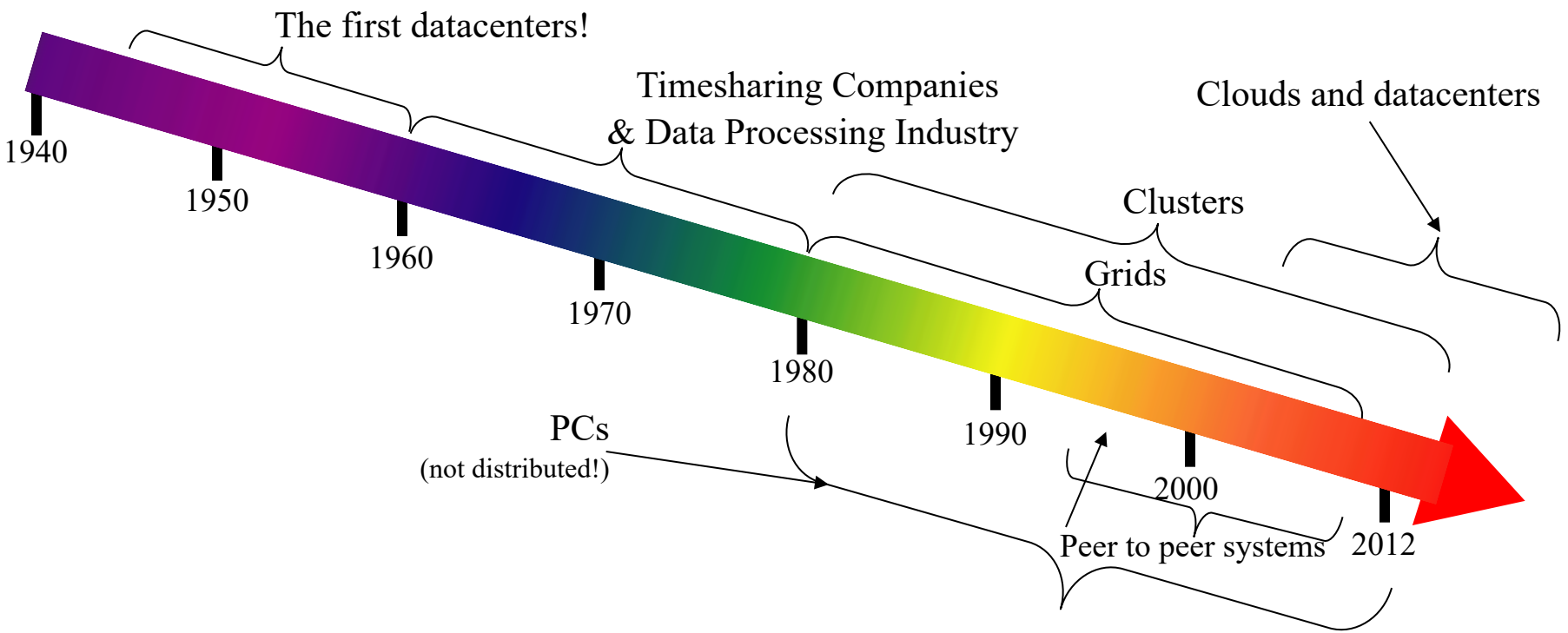
- A single-site cloud (aka “Datacenter”) consists of
  - Compute nodes (grouped into racks)
  - Switches, connecting the racks
  - A network topology, e.g., hierarchical
  - Storage (backend) nodes connected to the network
  - Front-end for submitting jobs and receiving client requests
  - (Often called 3-tier architecture)
  - Software Services
- A geographically distributed cloud consists of
  - Multiple such sites
  - Each site perhaps with a different structure and services

# A Sample Cloud Topology




**So then, what is a cluster?**

# “A Cloudy History of Time”



# Vacuum Cube Computer

- Built in 1954
- 60K Vacuum Cubes
- 75K instructions per second
- 0.5 acres
- Cost: 10 billion dollars

# IBM 7090

- “In 1960, a typical system sold for $2.9 million (equal to $23.1 million in 2014) or could be rented for $63,500 a month (equal to $520,000 in 2014). “

# “A Cloudy History of Time”

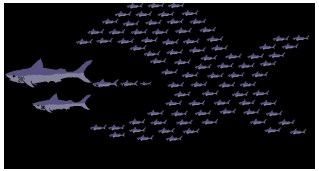
First large datacenters: ENIAC, ORDVAC, ILLIAC
Many used vacuum tubes and mechanical relays

1940

1950

1960

IBM
xerox
Honeywell



Berkeley NOW Project
Supercomputers
Server Farms (e.g., Oceano)

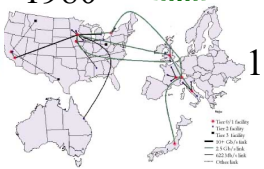
1970

CRAY
THE SUPERCOMPUTER COMPANY

P2P Systems (90s-00s)
•Many Millions of users
•Many GB per day

1980

Open Science Grid



1990

2000

LimeWire
KaZaA
BitTorrent

2012

Clouds

Data Processing Industry
- 1968: $70 M. 1978: $3.15 Billion

Timesharing Industry (1975):
•Market Share: Honeywell 34%, IBM 15%,
•Xerox 10%, CDC 10%, DEC 10%, UNIVAC 10%
•Honeywell 6000 & 635, IBM 370/168,
Xerox 940 & Sigma 9, DEC PDP-10, UNIVAC 1108

Grids (1980s-2000s):
•GriPhyN (1970s-80s)
•Open Science Grid and Lambda Rail (2000s)
•Globus & other standards (1990s-2000s)

- ICDCS 2018 Keynote: Manfred Hauswirth, “Been there, done that, bought the t-shirt: Trends in distributed systems architecture”

- The history of distributed system architecture exhibits many extremes – from decentralization to centralization and back again. Each phase of evolution had improvements and replaced the previous generation of architectures in a wide range of applications. When the limitations of the new dominating paradigm were understood, the pendulum went back to the other side. This was the “normal evolution cycle” in computer science for quite some time.

# Trends: Technology

- Doubling Periods – storage: 12 mos, bandwidth: 9 mos, and (what law is this?) cpu compute capacity: 18 mos
  - Which one increases the fastest?
- Then and Now
  - Bandwidth
    - 1985: mostly 56Kbps links nationwide
    - 2014: Tbps links widespread
  - Disk capacity
    - Today's PCs have TBs, far more than a 1990 supercomputer

# Trends: Users

- Then and Now

  Biologists:

  - 1990: were running small single-molecule simulations
  - 2012: CERN's Large Hadron Collider producing many PB/year

# Prophecies

- In 1965, MIT's Fernando Corbató and the other designers of the Multics operating system envisioned a computer facility operating “like a power company or water company”.

- Plug your thin client into the computing Utility and Play your favorite Intensive Compute & Communicate Application

# Four Features New in Today's Clouds

I. Massive scale.

II. On-demand access: Pay-as-you-go, no upfront commitment.
- And anyone can access it

III. Data-intensive Nature: What was MBs has now become TBs, PBs and XBs.
- Daily logs, forensics, Web data, etc.
- Humans have data numbness: Wikipedia (large) compressed is only about 10 GB!

IV. New Cloud Programming Paradigms: MapReduce/Hadoop, NoSQL/Cassandra/MongoDB and many others.
- High in accessibility and ease of programmability
- Lots of open-source

Combination of one or more of these gives rise to novel and unsolved distributed computing problems in cloud computing.

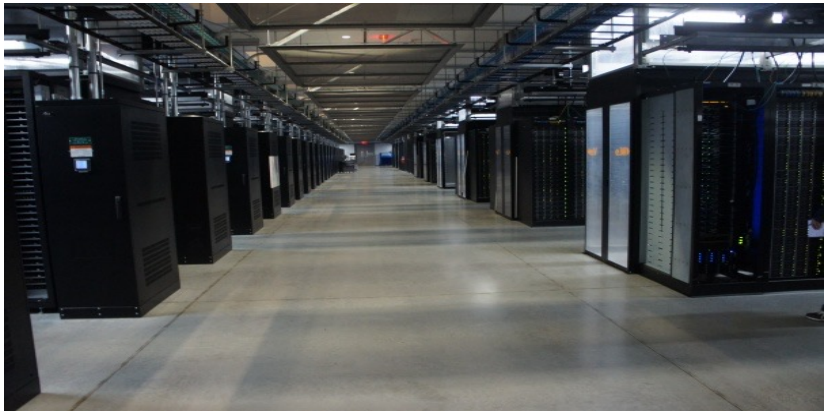
# I. Massive Scale

- Facebook [GigaOm, 2012]
  - 30K in 2009 -> 60K in 2010 -> 180K in 2012
- Microsoft [NYTimes, 2008]
  - 150K machines in 2008 (over one million in 2013)
  - Growth rate of 10K per month
  - 80K total running Bing
- AWS EC2 [Randy Bias, 2009]
  - 40K machines (Netcraft estimates the number to be 158k in 2013)
  - 8 cores/machine
- eBay [2012]: 50K machines
- HP [2012]: 380K in 180 DCs
- Google: A lot

# What does a datacenter look like from inside?

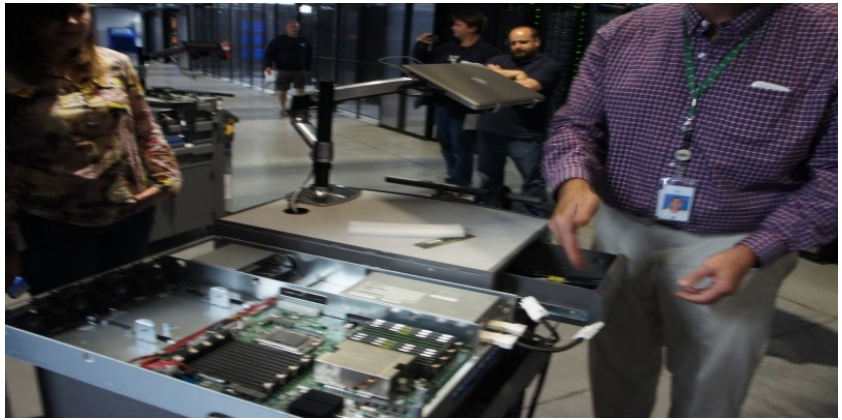
- A virtual walk through a datacenter
- Reference: https://gigaom.com/2012/08/17/a-rare-look-inside-facebooks-oregon-data-center-photos-video/ /

# Servers



Front

Back



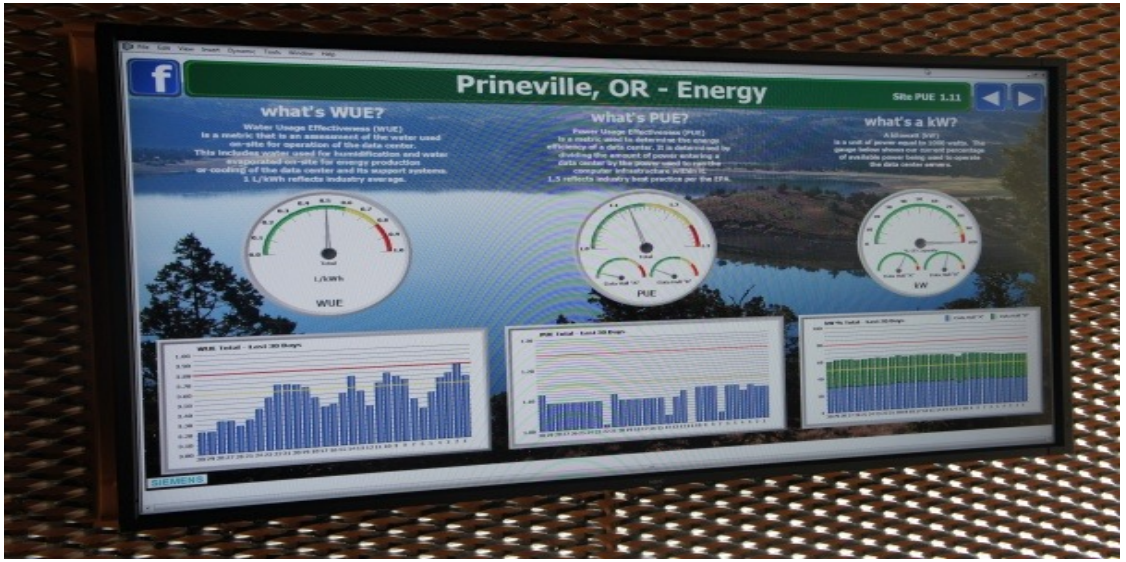
In

Some highly secure  (e.g., financial info)

# Power

Off-site

On-site

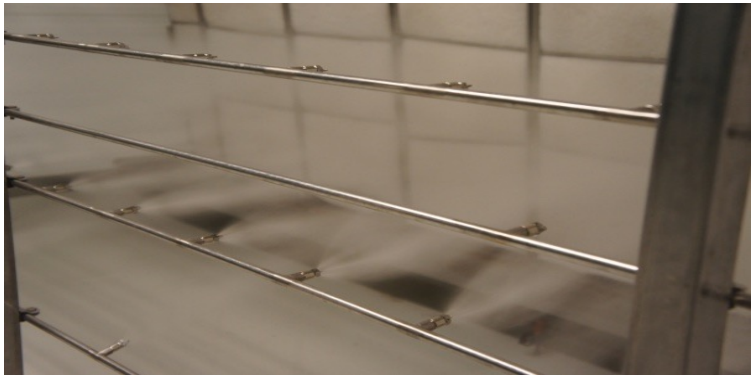



- WUE = Annual Water Usage / IT Equipment Energy (L/kWh) – low is good
- PUE = Total facility Power / IT Equipment Power – low is good (e.g., Google~1.11)

# Cooling

Air sucked in from top (also, Bugzappers)


Water purified



Water sprayed into air

15 motors per server bank

# Extra - Fun Videos to Watch

- A visit into the cloud
  - https://www.youtube.com/watch?v=94PO2-TL4Vs&ab_channel=CBSSundayMorning
- Microsoft GFS Datacenter Tour (Youtube)
  - https://www.youtube.com/watch?v=9VWA-7_-Pb0
- Timelapse of a Datacenter Construction on the Inside (Fortune 500 company)
  - http://www.youtube.com/watch?v=ujO-xNvXj3g

# II. On-demand access: *aaS Classification

On-demand: renting a cab vs. (previously) renting a car, or buying one. E.g.:

- AWS Elastic Compute Cloud (EC2): a few cents to a few $ per CPU hour
- AWS Simple Storage Service (S3): a few cents to a few $ per GB-month

- HaaS: Hardware as a Service
  - You get access to barebones hardware machines, do whatever you want with them, Ex: Your own cluster
  - Not always a good idea because of security risks
- IaaS: Infrastructure as a Service
  - You get access to flexible computing and storage infrastructure. Virtualization is one way of achieving this (what's another way, e.g., using Linux). Often said to subsume HaaS.
  - Ex: Amazon Web Services (AWS: EC2 and S3), Eucalyptus, Rightscale, Microsoft Azure, Google Compute Engine.

# II. On-demand access: *aaS Classification

- PaaS: Platform as a Service
  - You get access to flexible computing and storage infrastructure, coupled with a software platform (often tightly coupled)
  - Ex: Google's AppEngine (Python, Java, Go)
- SaaS: Software as a Service
  - You get access to software services, when you need them. Often said to subsume SOA (Service Oriented Architectures).
  - Most of them can be accessed through a browser without installing any software
  - Ex: Google docs, MS Office on demand

# III. Data-intensive Computing

- Computation-Intensive Computing
  - Example areas: MPI-based, High-performance computing, Grids
  - Typically run on supercomputers (e.g., NCSA Blue Waters)
- Data-Intensive
  - Typically store data at datacenters
  - Use compute nodes nearby
  - Compute nodes run computation services
- In data-intensive computing, the focus shifts from computation to the data: CPU utilization no longer the most important resource metric, instead I/O is (disk and/or network)

# IV. New Cloud Programming Paradigms

- Easy to write and run highly parallel programs in new cloud programming paradigms:
  - Google: MapReduce and Sawzall
  - Amazon: Elastic MapReduce service (pay-as-you-go)
  - Google (MapReduce)
    - Indexing: a chain of 24 MapReduce jobs
    - ~200K jobs processing 50PB/month (in 2006)
  - Yahoo! (Hadoop + Pig)
    - WebMap: a chain of several MapReduce jobs
    - 300 TB of data, 10K cores, many tens of hours
  - Facebook (Hadoop + Hive)
    - ~300TB total, adding 2TB/day (in 2008)
    - 3K jobs processing 55TB/day
  - Similar numbers from other companies, e.g., Yieldex, eharmony.com, etc.
  - NoSQL: MySQL is an industry standard, but Cassandra is 2400 times faster!

# Two Categories of Clouds

- Can be either a (i) public cloud, or (ii) private cloud
- Private clouds are accessible only to company employees
- Public clouds provide service to any paying customer

- You're starting a new service/company: should you use a public cloud or purchase your own private cloud?

# Single site Cloud: to Outsource or Own?

- Medium-sized organization: wishes to run a service for *M* months
  - Service requires 128 servers (1024 cores) and 524 TB
- Outsource (e.g., via AWS): *monthly* cost
  - S3 costs: $0.12 per GB month. EC2 costs: $0.10 per CPU hour (costs from 2009)
  - Storage = $ 0.12 X 524 X 1000 ~ $62 K
  - Total = Storage + CPUs = $62 K + $0.10 X 1024 X 24 X 30 ~ $136 K
- Own: monthly cost
  - Storage ~ $349 K / *M*
  - Total ~ $ 1555 K / *M* + 7.5 K (includes 1 sysadmin / 100 nodes)
    - using 0.45:0.4:0.15 split for hardware:power:network and 3 year lifetime of hardware

# Single site Cloud: to Outsource or Own?

- Breakeven analysis: more preferable to own if:
  - $349 K / *M* < $62 K (storage)
  - $ 1555 K / *M* + 7.5 K < $136 K (overall)

  *Breakeven points*
  - *M* > 5.55 months (storage)
  - *M* > 12 months (overall)

- As a result
  - Startups use clouds a lot
  - Cloud providers benefit monetarily most from storage

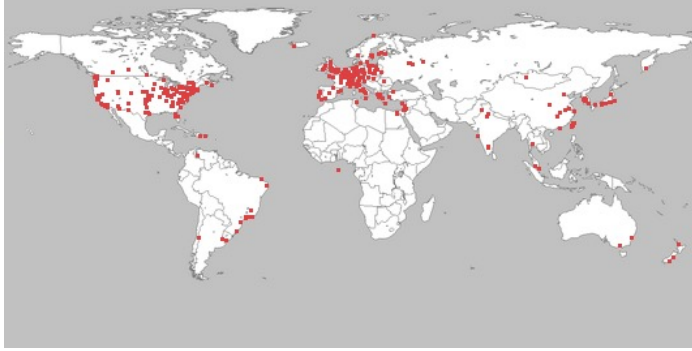
# Academic Clouds: Emulab

- A community resource open to researchers in academia and industry. Very widely used by researchers everywhere today.
- https://www.emulab.net/
- A cluster, with currently ~500 servers
- Founded and owned by University of Utah (led by Late Prof. Jay Lepreau)

- As a user, you can:
  - Grab a set of machines for your experiment
  - You get root-level (sudo) access to these machines
  - You can specify a network topology for your cluster
  - You can emulate any topology

PLANETLAB

An open platform for developing, deploying, and accessing planetary-scale services



All images © PlanetLab

- A community resource open to researchers in academia and industry
- http://www.planet-lab.org/
- Currently, ~ 1077 nodes at ~500 sites across the world
- Founded at Princeton University (led by Prof. Larry Peterson), but owned in a federated manner by the sites

- Node: Dedicated server that runs components of PlanetLab services.
- Site: A location, e.g., TAMU, UT, Texas Tech, etc., that hosts a number of nodes.
- Sliver: Virtual division of each node. Currently, uses VMs, but it could also other technology. Needed for timesharing across users.
- Slice: A spatial cut-up of the PL nodes. Per user. A slice is a way of giving each user (Unix-shell like) access to a subset of PL machines, selected by the user. A slice consists of multiple slivers, one at each component node.
- Thus, PlanetLab allows you to run real world-wide experiments.
- Many services have been deployed atop it, used by millions (not just researchers): Application-level DNS services, Monitoring services, CoralCDN, etc.

# Summary

- Clouds build on many previous generations of distributed systems
- Especially the timesharing and data processing industry of the 1960-70s.
- Need to identify unique aspects of a problem to classify it as a new cloud computing problem
  - Scale, On-demand access, data-intensive, new programming
- Otherwise, the solutions to your problem may already exist!