Cloud Computing

ECPE 293A

Overview

Based on “Above the Clouds: A Berkeley View of Cloud Computing”, 2009

Schedule

- Tue, Jan 21st What is Cloud Computing?
  - Continuation of today's discussion
  - **Your Homework:** Pick 2-3 papers from the approved reading list that you could present and **email me**
- Thur, Jan 23rd – First paper presentation
  - Presenter: Dr. Shafer *(use an an example)*
  - MapReduce paper *(used for your first project)*
  - **Your Homework 3:** Audience members role
    - Read paper and prepare summary document

Cloud Computing

- How are we defining cloud computing again?
- And why do people use it?

What is Old and What is New?

- Old idea – **utility computing**
  - What if computing was as ubiquitous as the power grid? Just flip a switch, and (presto!) computation!
  - Billed for only the resources you consume
  - This vision took decades to be achieved!

"If computers of the kind I have advocated become the computers of the future, then computing may someday be organized as a public utility just as the telephone system is a public utility... The computer utility could become the basis of a new and important industry."

—1961, *John McCarthy* (inventor of Lisp, Turing Award winner)

What is Old and What is New?

- New ideas:
  - No up-front cost
  - Fine-grained billing (hourly)
  - Illusion of infinite resources

Why Now for Cloud Computing?

- First .com boom created companies with experience in very large datacenters
  - Economies of scale – 5-7 times cheaper (going from a 1,000 machine to 50,000 machine datacenter)

Table 2: Economies of scale in 2006 for medium-sized datacenter (≈1000 servers) vs. very large datacenter (≈50,000 servers). [24]

Technology	Cost in Medium-sized DC	Cost in Very Large DC	Ratio
Network	$95 per Mbit/sec/month	$13 per Mbit/sec/month	7.1
Storage	$2.20 per GByte / month	$0.40 per GByte / month	5.7
Administration	≈140 Servers / Administrator	>1000 Servers / Administrator	7.1

Datacenter

Apple's 1 billion dollar datacenter in North Carolina

- Warehouse for computers
- Design goals
  - Maximum density for minimum space
  - Economy of scale – few people managing large numbers of computers
  - Security
  - Network and power redundancy

The Datacenter

The Datacenter

Google Datacenter (1 of many…)

Microsoft Datacenter (Dublin, Ireland)

NSA Datacenter (Bluffdale, Utah. 2+ Billion $$)

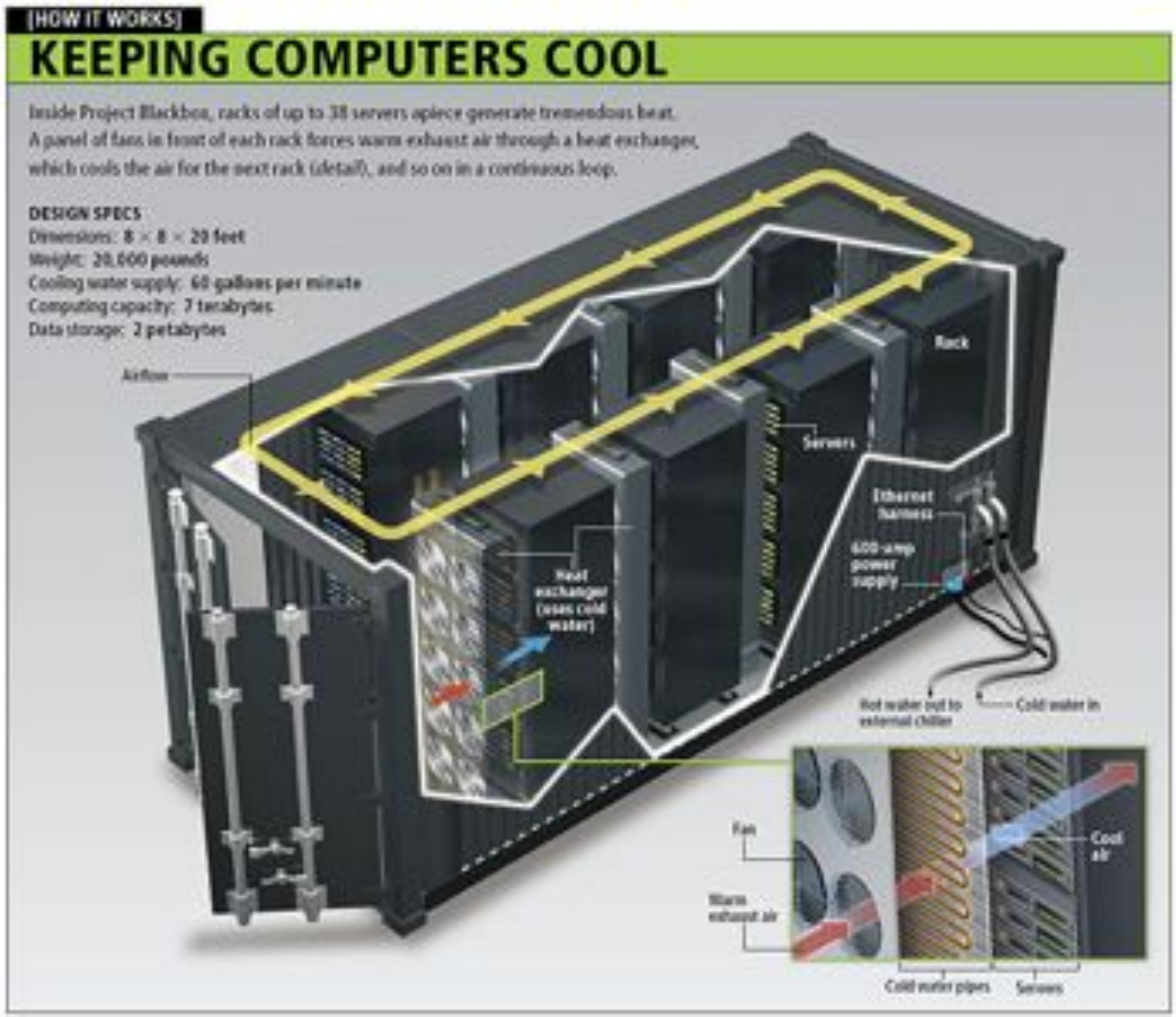
Datacenter Designs – Traditional Racks

42U rack = 42 “1U” servers

Datacenter Designs – Traditional Racks

Datacenter Design – Innovative

- Shipping containers with 2000+ servers pre-installed?
- Water cooled?



Datacenter Design – Innovative

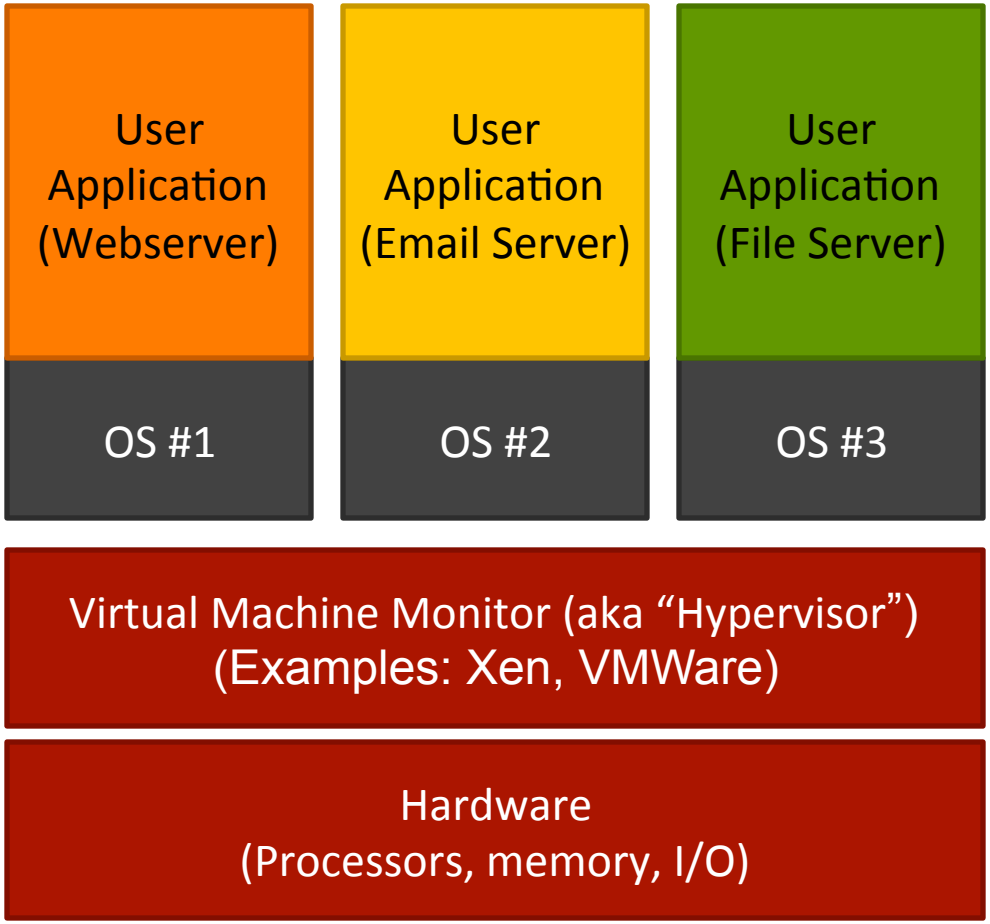
- Traditional cooling (chilled water or air) is expensive and bad for the environment
- Can we run servers hotter and use ambient air instead?

Why Now for Cloud Computing?

- Pervasive broadband Internet
- Standard hardware/software stack
- Fast x86 / x86-64 virtualization

x86 Virtualization

- **Virtual machine monitor** controls several guest domains
- Services
  - CPU scheduling
  - Memory allocation
  - Resource sharing
  - Protection/Isolation
- A virtual machine provides the same type of services to a guest domain that a general OS provides to individual processes!



Sharing Homogeneous Resources

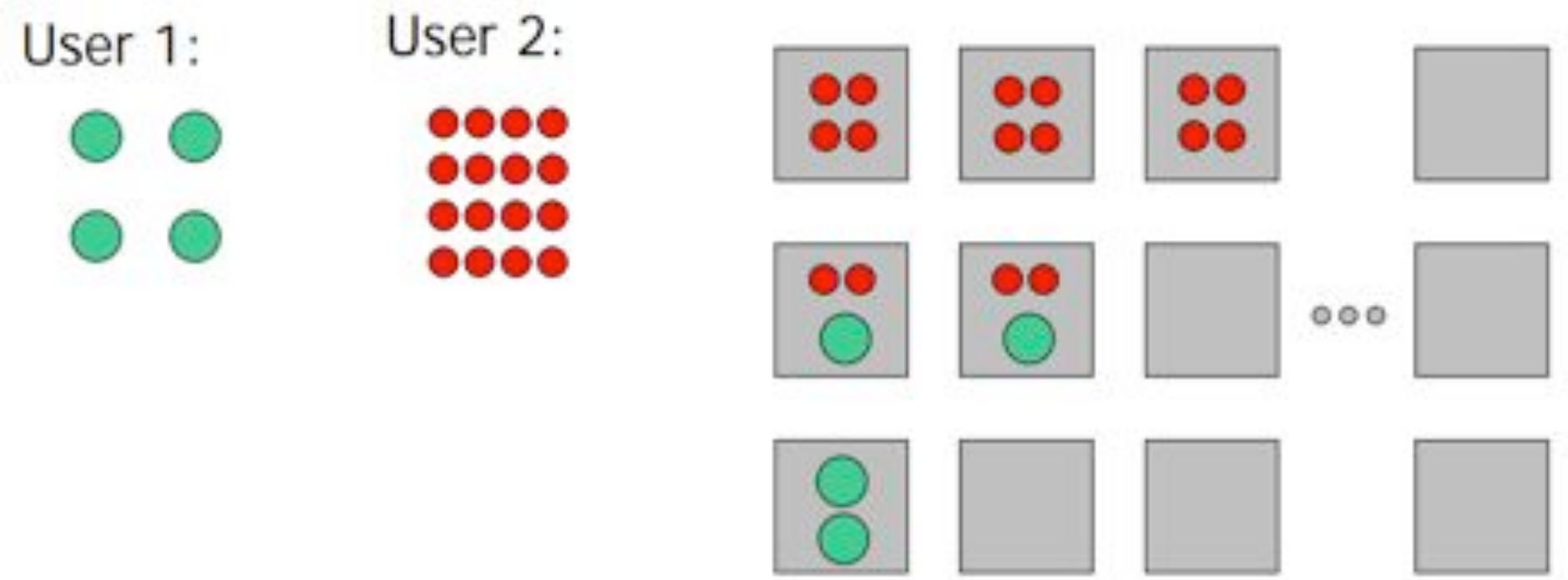


Figure from http://www.qatar.cmu.edu/~msakr/15319-s10/lectures/lecture02.pdf

Sharing Heterogeneous Resources

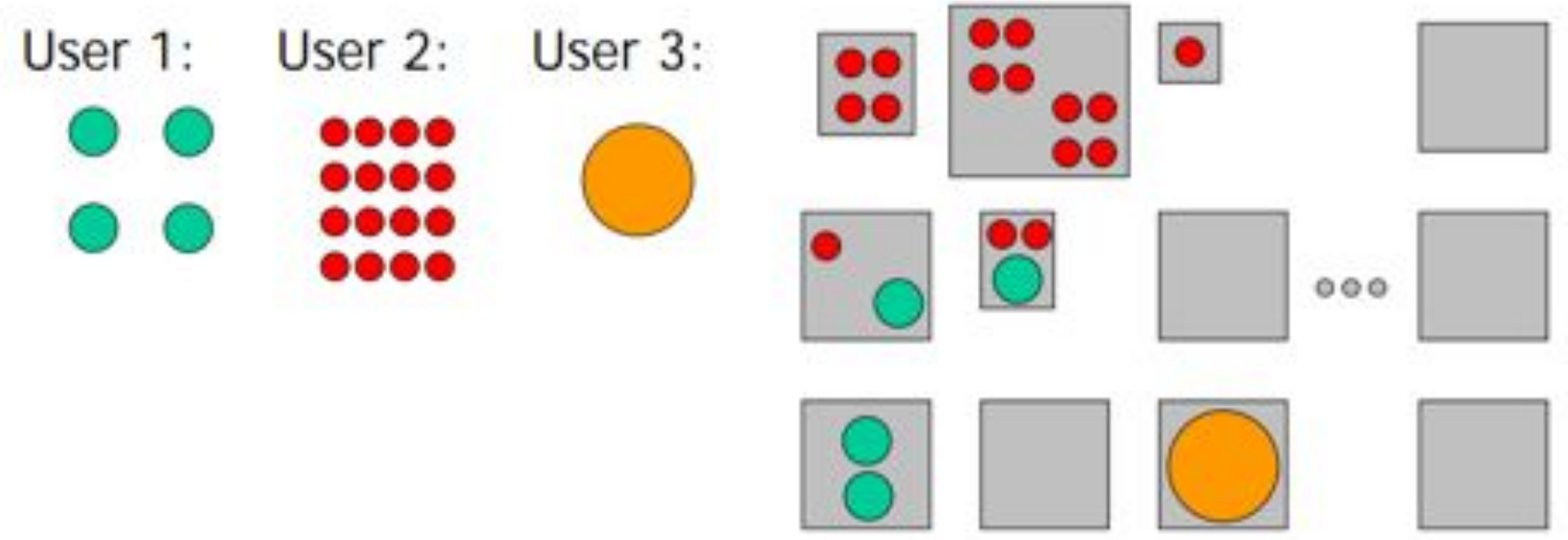


Figure from http://www.qatar.cmu.edu/~msakr/15319-s10/lectures/lecture02.pdf

More Virtualization

Virtual Networks

- One physical datacenter network that is shared
  - Each customer thinks that their virtual machines are in the same rack connected to the same private network
  - But in reality, they could be widely separated!
- **Why is this useful?**

Virtual Disks

- One storage array in datacenter that is shared
  - Each customer OS thinks it is managing its own private disk
  - But in reality, it's just a file spread out across many disks of a large array!
- **Why is this useful?**

Spectrum of Cloud Designs

- Virtualization provides **isolation** between customers
  - Share CPU, memory, disk dynamically
- Tradeoff: Flexibility/portability versus built-in features
  - Amazon EC2
    - Virtualization at the **instruction/hardware level**
  - Microsoft Azure
    - Virtualization at the **bytecode level**
  - Google AppEngine
    - Virtualization at the **framework level**

Amazon EC2

- Amazon sells you one virtual machine instance (or a thousand!)
  - You configure the OS
  - You configure the application software
  - Thin API (related to starting/stopping machines)
  - Virtualization: raw CPU cycles, block-device storage, IP-level connectivity
- **Advantages?**
- **Disadvantages?**

Specs as of Jan 2014
1 "unit" = One 1.0 GHz "2007-era" Xeon/Opteron CPU

Amazon EC2 – Instance Types

Node Type	RAM	CPU	Storage (local)	Notes
t1.micro	613 MB	2 units (**burst** only!)	None	"Free"
m3.xlarge	15GB	13 units (4 cores)	80 GB SSD	
m3.2xlarge	30GB	26 units (8 cores)	160 GB SSD	
m2.xlarge	17.1 GB	6.5 units (2 cores)	420 GB	Greater RAM
…	…	…	…	
m2.4xlarge	68.4GB	26 units (8 cores / 3.25)	1690 GB	
c3.large	3.75GB	7 units (2 cores)	32 GB SSD	Greater CPU
…	…	…	…	
c3.8xlarge	60GB	108 units (32 cores)	640 GB SSD	
cc1.8xlarge	244GB	88 units	240 GB	10 GigE
g2.2xlarge	15GB	26 units + 1 GK104 GPU	60 GB	GPU

Amazon EC2 – January 2014 Pricing

Linux | RHEL | SLES | Windows | Windows with SQL Standard | Windows with SQL Web

On-Demand Instance Prices

Region: US East (N. Virginia)

	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
General Purpose - Current Generation					
m3.xlarge	4	13	15	2 x 40 SSD	$0.450 per Hour
m3.2xlarge	8	26	30	2 x 80 SSD	$0.900 per Hour
General Purpose - Previous Generation					
m1.small	1	1	1.7	1 x 160	$0.060 per Hour
m1.medium	1	2	3.75	1 x 410	$0.120 per Hour
m1.large	2	4	7.5	2 x 420	$0.240 per Hour
m1.xlarge	4	8	15	4 x 420	$0.480 per Hour
Compute Optimized - Current Generation					
c3.large	2	7	3.75	2 x 16 SSD	$0.150 per Hour
c3.xlarge	4	14	7.5	2 x 40 SSD	$0.300 per Hour
c3.2xlarge	8	28	15	2 x 80 SSD	$0.600 per Hour
c3.4xlarge	16	55	30	2 x 160 SSD	$1.200 per Hour
c3.8xlarge	32	108	60	2 x 320 SSD	$2.400 per Hour
Compute Optimized - Previous Generation					
c1.medium	2	5	1.7	1 x 350	$0.145 per Hour
c1.xlarge	8	20	7	4 x 420	$0.580 per Hour
cc2.8xlarge	32	88	60.5	4 x 840	$2.400 per Hour
GPU Instances - Current Generation					
g2.2xlarge	8	26	15	60 SSD	$0.650 per Hour
GPU Instances - Previous Generation					
cg1.4xlarge	16	33.5	22.5	2 x 840	$2.100 per Hour

Amazon EC2 – January 2014 Pricing

- **Why are the Windows instances more expensive?**

Linux | RHEL | SLES | **Windows** | Windows with SQL Standard | Windows with SQL Web

On-Demand Instance Prices

Region: US East (N. Virginia)

	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Windows Usage
General Purpose - Current Generation					
m3.xlarge	4	13	15	2 x 40 SSD	$0.702 per Hour
m3.2xlarge	8	26	30	2 x 80 SSD	$1.404 per Hour
General Purpose - Previous Generation					
m1.small	1	1	1.7	1 x 160	$0.091 per Hour
m1.medium	1	2	3.75	1 x 410	$0.182 per Hour
m1.large	2	4	7.5	2 x 420	$0.364 per Hour
m1.xlarge	4	8	15	4 x 420	$0.728 per Hour
Compute Optimized - Current Generation					
c3.large	2	7	3.75	2 x 16 SSD	$0.233 per Hour
c3.xlarge	4	14	7.5	2 x 40 SSD	$0.466 per Hour
c3.2xlarge	8	28	15	2 x 80 SSD	$0.932 per Hour
c3.4xlarge	16	55	30	2 x 160 SSD	$1.864 per Hour
c3.8xlarge	32	108	60	2 x 320 SSD	$3.728 per Hour
Compute Optimized - Previous Generation					
c1.medium	2	5	1.7	1 x 350	$0.225 per Hour
c1.xlarge	8	20	7	4 x 420	$0.900 per Hour
cc2.8xlarge	32	88	60.5	4 x 840	$2.970 per Hour
GPU Instances - Current Generation					
g2.2xlarge	8	26	15	60 SSD	$0.767 per Hour
GPU Instances - Previous Generation					
cg1.4xlarge	16	33.5	22.5	2 x 840	$2.600 per Hour

Amazon EC2 – January 2014 Pricing

Region: US East (N. Virginia)

	Pricing
Data Transfer IN To Amazon EC2 From	
Internet	$0.00 per GB
Another AWS Region (from any AWS Service)	$0.00 per GB
Amazon S3, Amazon Glacier, Amazon DynamoDB, Amazon SQS, or Amazon SimpleDB in the same AWS Region	$0.00 per GB
Amazon EC2, Amazon RDS and Amazon ElastiCache instances or Elastic Network Interfaces in the same Availability Zone	
Using a private IP address	$0.00 per GB
Using a public or Elastic IP address	$0.01 per GB
Amazon EC2, Amazon RDS and Amazon ElastiCache instances or Elastic Network Interfaces in another Availability Zone in the same AWS Region	$0.01 per GB
Data Transfer OUT From Amazon EC2 To	
Amazon S3, Amazon Glacier, Amazon DynamoDB, Amazon SQS, Amazon SimpleDB in the same AWS Region	$0.00 per GB
Amazon EC2, Amazon RDS, or Amazon ElastiCache instances, Amazon Elastic Load Balancing, or Elastic Network Interfaces in the same Availability Zone	
Using a private IP address	$0.00 per GB
Using a public or Elastic IP address	$0.01 per GB
Amazon EC2, Amazon RDS or Amazon ElastiCache instances, Amazon Elastic Load Balancing, or Elastic Network Interfaces in another Availability Zone in the same AWS Region	$0.01 per GB
Another AWS Region or Amazon CloudFront	$0.02 per GB
Data Transfer OUT From Amazon EC2 To Internet	
First 1 GB / month	$0.00 per GB
Up to 10 TB / month	$0.12 per GB
Next 40 TB / month	$0.09 per GB
Next 100 TB / month	$0.07 per GB
Next 350 TB / month	$0.05 per GB

Microsoft Azure

- Microsoft sells you a "platform"
  - You write your application in **.NET**, Java, PHP, JavaScript (node.js), C++, or Python and compile to a common language runtime
  - No control over underlying framework and OS beyond what their API allows
- Application model
  - Web role – HTTP request comes in, your app runs (on one of ∞ nodes), and then finishes
  - Worker role – Background program (not triggered by user)
  - VM role – (Amazon EC2 style, gives you a Windows or Linux server VM that can be customized)

Microsoft Azure

- Data storage options
  - Blobs (unstructured data = doc, picture, video, etc..)
  - Tables (non-relational database: key and many values)
    - Imagine a row in Excel, but each row could have different columns
  - Azure SQL: Full-fledged parallel relational SQL database
  - Local storage: Like Amazon's (doesn't move with your VM!)
- **Advantages? Disadvantages?**

Google AppEngine

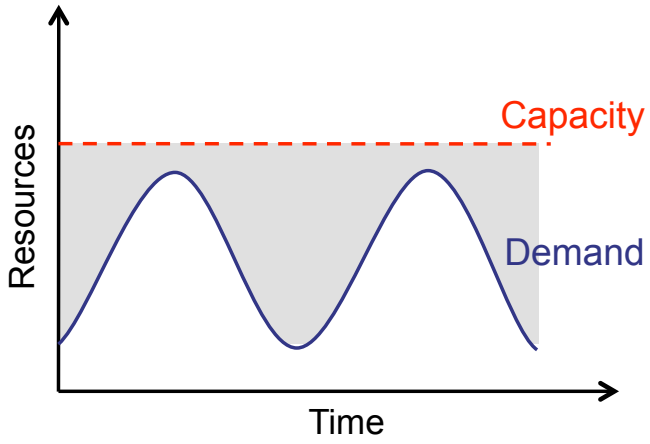
- Google (also) sells you a “platform” targeted at web apps
  - Supports Python, Java, PHP, and Go
  - Stateless computation, stateful storage
  - Request/reply operation
- Constraints (your app is in a sandbox on frontend servers that appear and disappear)
  - No writing to files
  - No network sockets
  - 60 seconds max execution after a request
- **Advantages? Disadvantages?**

Analogy with Programing Languages

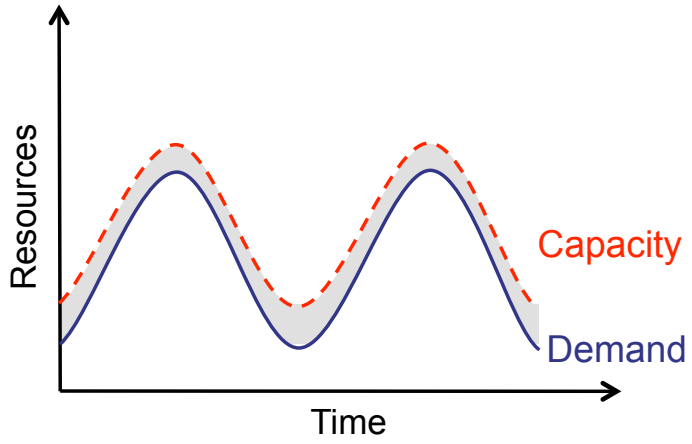
- Assembly or C programming provides you with hardware-level access and fine grained control
- But writing a web app is tedious!
  - Managing sockets, memory, threads, etc…
  - Good libraries help but it's still hard work

Cloud Economics

- Pay per use instead of provisioning for peak usage



Static data center

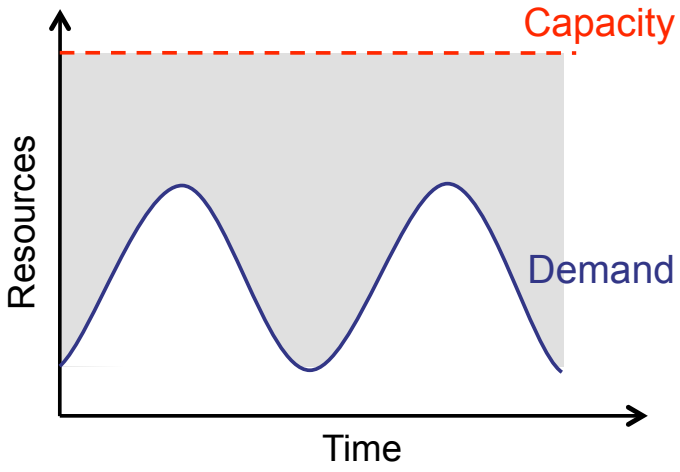


Data center in the cloud

Unused resources

Cloud Economics

- What if we **over**-provision?

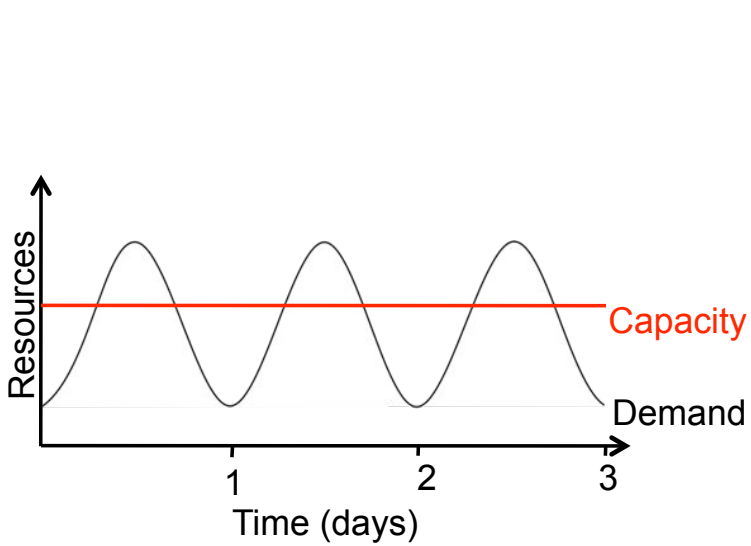


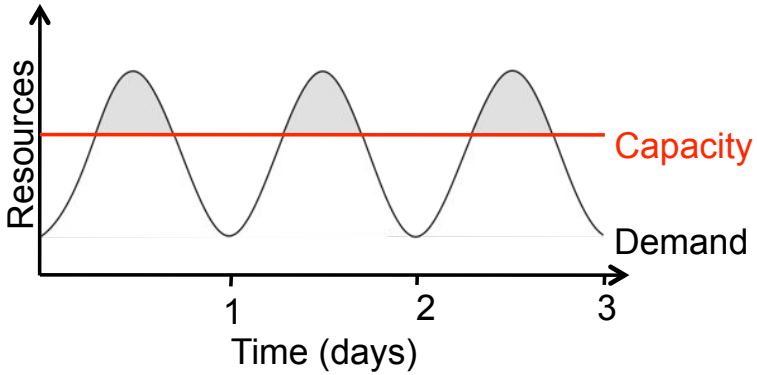
Unused resources

Static data center

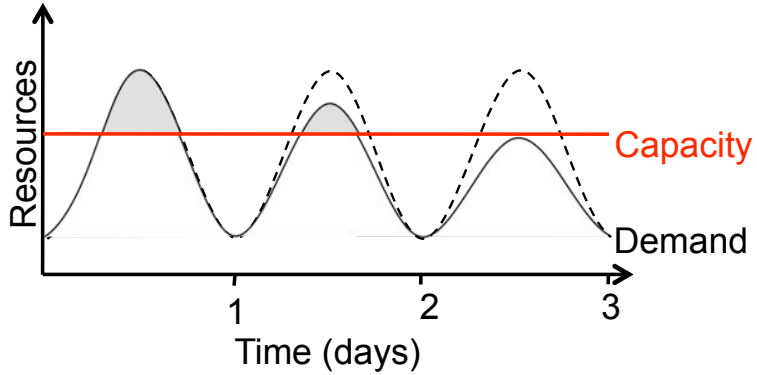
Cloud Economics

- What if we **under**-provision?





Lost revenue



Lost users

Cloud Economics

- Note that it is just as important to be able to scale **down** as it is to scale **up – why?**
- Typical usage case
  - You're a startup and need 10 servers for your average traffic
  - Your website is suddenly mentioned on *Good Morning America!* and traffic spikes 10x
  - 24 hours later, traffic is back to your usual average

Cloud Economics

- Cheaper to ship photons than electrons
  - Place your datacenter close to cheap power (hydro dams in rural areas?)
  - Link to the national fiber optic network
- Cheaper to go **LARGE**!

Table 2: Economies of scale in 2006 for medium-sized datacenter (≈1000 servers) vs. very large datacenter (≈50,000 servers). [24]

Technology	Cost in Medium-sized DC	Cost in Very Large DC	Ratio
Network	$95 per Mbit/sec/month	$13 per Mbit/sec/month	7.1
Storage	$2.20 per GByte / month	$0.40 per GByte / month	5.7
Administration	≈140 Servers / Administrator	>1000 Servers / Administrator	7.1

Why be a Cloud Vendor?

- **Why have Amazon, Google, Microsoft entered this market?**
- Amazon and Google
  - Utilize off-peak capacity in datacenter
  - Reuse existing infrastructure and technical know-how
  - Grow datacenters even larger, and achieve even greater economies of scale (which benefits both them and their customers)
- Microsoft
  - Sell .NET tools (defend the franchise!)

Cloud Challenges & Opportunities

- Challenge 1: Availability of Service (avoiding downtime)
- **Challenges?** ***(for you as a customer of cloud services)***
  - Single point of failure
    - What if your rack fails?
    - What if the entire datacenter is cut offline?
    - What if all of Amazon EC2 goes offline due to common bug?
    - What if Amazon goes out of business?
  - DDOS attacks
- **Solutions / Opportunities?**
  - Use multiple cloud providers to provide business continuity
  - Use elasticity to defend against DDOS attack

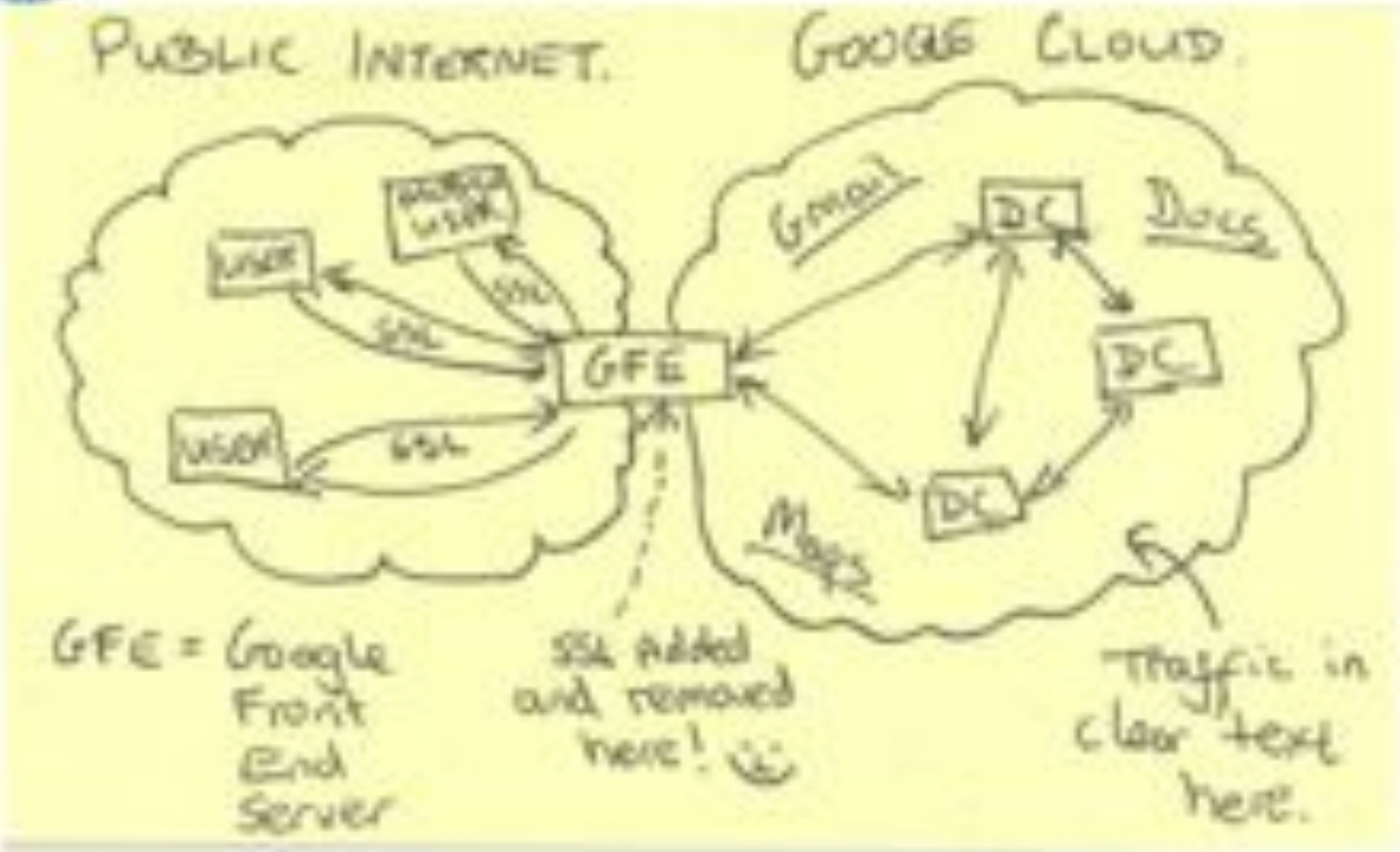
Cloud Challenges & Opportunities

- Challenge 2: Data Lock-in
- **Why is this a problem?** ***(for you as a customer of cloud services)***
  - Your vendor might start raising prices, decrease quality, or go out of business, and you can't easily take your data and go elsewhere
- **What can be done about it?**
  - Standardized APIs?
    - Example: Eucalyptus

Cloud Challenges & Opportunities

- Challenge 3: Data Confidentiality and Auditability
- **Why is this a problem?** ***(for you as a customer of cloud services)***
  - Who can access my data?
  - How can my data be audited if it is stored outside my organization?
  - Regulatory compliance?
  - Access by foreign governments?
- **What can be done about it?**
  - Encrypt (storage), encrypt (network/VPN)
  - Storage within country boundaries
  - Have the cloud provider (in the VM itself) guarantee data

Current Efforts - Google



Cloud Challenges & Opportunities

- Challenge 4: Data Transfer Bottlenecks
- **Why is this a problem?**
  - Limited upload/download bandwidth to cloud
    (at least, relative to the TBs of data you might like to move)
- **What can be done about it?**
  - FedEx your hard drives! (Seriously)
    - http://aws.amazon.com/importexport/
    - $80 per disk + $2.49/hr to copy
  - Do all of your data processing internal to the cloud system
    (i.e. inside Amazon's datacenter)
  - Better network architectures?

Cloud Challenges & Opportunities

- Challenge 5: Performance Unpredictability
- **Why does this problem exist?**
  - CPU and main memory is easy to virtualize (high bandwidth + context switches between users are quick)
  - Disks are hard to virtualize (hard drive bandwidth shared among 10 users is paltry + seek times are high)
- **What can be done about it?**
  - SSDs?
  - More disks = more spindles?
  - Better VM software to manage disks?

Cloud Challenges & Opportunities

- Challenge 6: Scalable Storage
- **Why is this a problem?**
  - As long as my data is in Amazon's cloud, I'm paying for it, regardless of whether or not I'm actively using it
- **What can be done about it?**
  - Nothing?
  - Don't keep so much data lying around?

Cloud Challenges & Opportunities

- Challenge 7: Bugs in Large-Scale Distributed Systems
- **Why is this a problem?**
  - How do you debug a problem that only occurs when you have > 100, > 1000, > 10000 machines working together?
- **What can be done about it?**
  - Log, log, log! (and have automated log analysis tools)
  - Can the VM help capture information beyond the view of the application?

Cloud Challenges & Opportunities

- Challenge 8: Scaling Quickly
- **Why is this a problem?**
  - Not every cloud service will automatically scale up/ down resources depending on your current load
- **What can be done about it?**
  - Better software

Cloud Challenges & Opportunities

- Challenge 9: Reputation Fate Sharing (with other customers of your cloud provider)
- **Why is this a problem?**
  - If some jerk sends spam from an Amazon EC2 instance, those IPs are probably blacklisted for all future customers
- **What can be done about it?**
  - Can the blacklists adapt?

Cloud Challenges & Opportunities

- Challenge 10: Software Licensing
- **Why is this a problem?**
  - How many licenses of Windows (or Oracle, etc..) do you need to buy if you run between 10 and 100 concurrent EC2 servers on any given day?
- **What can be done about it?**
  - Hope the software vendors offer better license terms? (Pay-per-use, bulk sales, etc…)
  - Open-source software?

What does the Cloud Change?

- **Application software** has to change
- New apps should be written in two pieces
  - Client piece (local) – must be useful if disconnected (temporarily) from the cloud
  - Cloud piece (remote)

What does the Cloud Change?

- **Infrastructure software** has to change
- Should be aware that it is running inside of a virtual machine (i.e. sharing a machine, instead of owning the hardware)
- Integrated billing/accounting system

What does the Cloud Change?

- **Hardware** has to change
- Larger scale! (Not just one machine, but dozens as the minimum unit)
- Energy efficiency (this was already becoming an issue)
  - Put N% of the CPU, memory, and disks to sleep when not needed (*energy proportionality*)
- Integrate virtualization into the system? (no such thing as bare hardware anymore?)

Is Every App Suitable for the Cloud?

- **What apps are good for the cloud?**
  - Web-style apps
  - Desktop apps (e.g. Google docs)
  - Batch processing
- **What apps are not good? (or "challenged"?)**
  - Jitter-sensitive apps
    - Latency over the Internet
    - Virtualization-imposed latency
  - Bulk data apps (*unless* the data is already in the cloud)

Public and Private Clouds

- **Public cloud**
  - Commercially available in a pay-as-you-go manner
  - Example: i.e. Amazon EC2
- **Private cloud**
  - Built by and available for only your company (or government)

Cloud Benefits: Public versus Private

Benefit	Public	Private
Economy of scale	Yes	No
Illusion of infinite resources on-demand	Yes	Unlikely
Eliminate up-front commitment by users*	Yes	No
True fine-grained pay-as-you-go **	Yes	??
Better utilization (workload multiplexing)	Yes	Depends on size**
Better utilization & simplified operations through virtualization	Yes	Yes

* What about nonrecoverable engineering/capital costs?

** Implies ability to meter & incentive to release idle resources

Public, Private, Hybrid Clouds

- **Public cloud**
  - Commercially available in a pay-as-you-go manner
  - Example: i.e. Amazon EC2
- **Private cloud**
  - Built by and available for only your company (or government)
- **Hybrid cloud – what's this?**
  - Using your local (private) computing resources first, but bursting (scaling up) to public cloud resources in periods of high demand
  - **Strengths and weaknesses?**