Big Data and the Cloud

Part I

Overview

- Computing at scale
  - The need for scalability; scale of current services
  - Scaling up: From PCs to data centers
  - Problems with 'classical' scaling techniques
- Utility computing and cloud computing
  - What are utility computing and cloud computing?
  - What kinds of clouds exist today?
  - What kinds of applications run on the cloud?
  - Virtualization: How clouds work 'under the hood'
  - Some cloud computing challenges

How many users and objects?

- Flickr has >6 billion photos
- Facebook has 1.15 billion active users
- Google is serving >1.2 billion queries/day on more than 27 billion items
- >2 billion videos/day watched on YouTube

How much data?

- Modern applications use massive data:
  - Rendering 'Avatar' movie required >1 petabyte of storage
  - eBay has >6.5 petabytes of user data
  - CERN's LHC will produce about 15 petabytes of data per year
  - In 2008, Google processed 20 petabytes per day
  - German Climate computing center dimensioned for 60 petabytes of climate data
  - Google now designing for 1 exabyte of storage
  - NSA Utah Data Center is said to have 5 zettabyte (!)
- How much is a zettabyte?
  - 1,000,000,000,000,000,000,000 bytes
  - A stack of 1TB hard disks that is 25,400 km high

How much computation?

- No single computer can process that much data
  - Need many computers!
- How many computers do modern services need?
  - Facebook is thought to have more than 60,000 servers
  - 1&1 Internet has over 70,000 servers
  - Akamai has 95,000 servers in 71 countries
  - Intel has ~100,000 servers in 97 data centers
  - Microsoft reportedly had at least 200,000 servers in 2008
  - Google is thought to have more than 1 million servers, is planning for 10 million (according to Jeff Dean)

Why should I care?

- Suppose you want to build the next Google
- How do you...
  - ... download and store billions of web pages and images?
  - ... quickly find the pages that contain a given set of terms?
  - ... find the pages that are most relevant to a given search?
  - ... answer 1.2 billion queries of this type <u>every day</u>?

- Suppose you want to build the next Facebook
- How do you...
  - ... store the profiles of over 500 million users?
  - ... avoid losing any of them?
  - ... find out which users might want to be friends?

- Stay tuned!

Plan for today

- Computing at scale
  - The need for scalability; scale of current services ✓
  - Scaling up: From PCs to data centers NEXT
  - Problems with 'classical' scaling techniques
- Utility computing and cloud computing
  - What are utility computing and cloud computing?
  - What kinds of clouds exist today?
  - What kinds of applications run on the cloud?
  - Virtualization: How clouds work 'under the hood'
  - Some cloud computing challenges

Scaling up

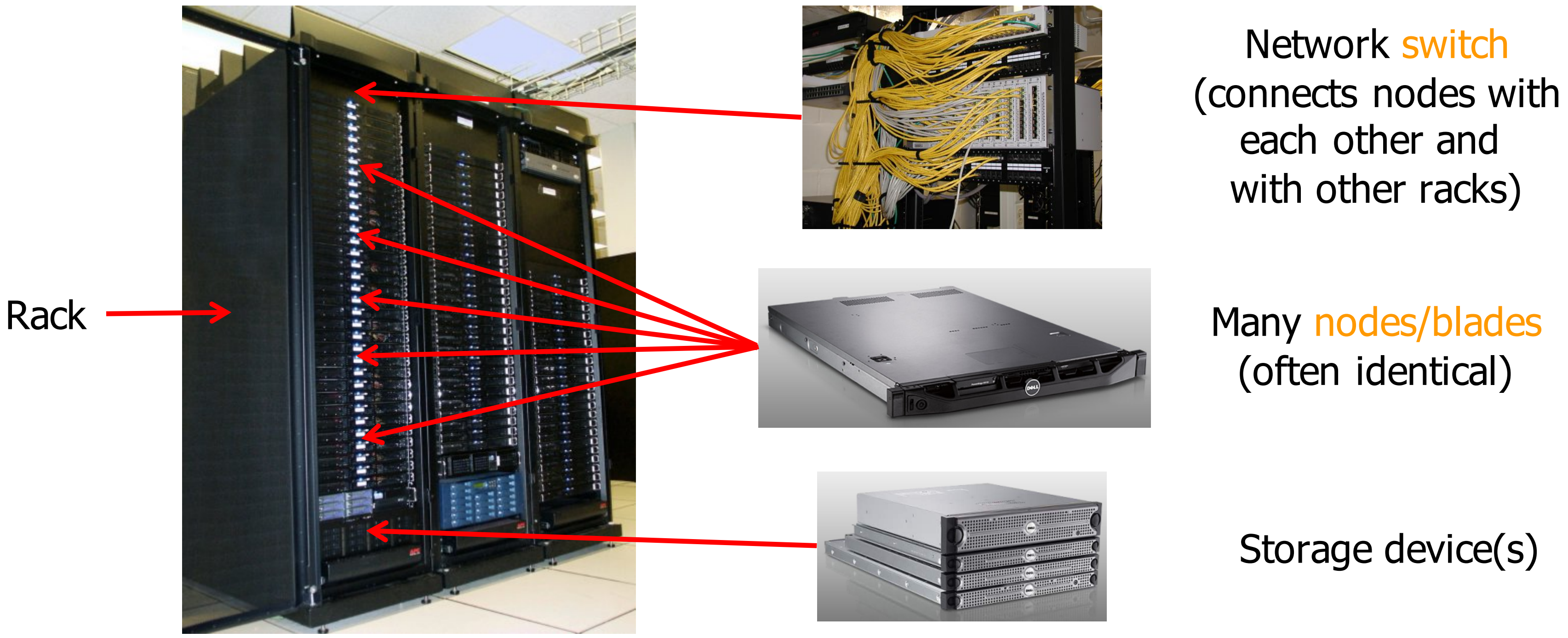
PC

Server

Cluster

- What if one computer is not enough?
  - Buy a bigger (server-class) computer
- What if the biggest computer is not enough?
  - Buy many computers

Clusters


Network switch (connects nodes with each other and with other racks)
Rack
Many nodes/blades (often identical)
Storage device(s)

- Characteristics of a cluster:
  - Many similar machines, close interconnection (same room?)
  - Often special, standardized hardware (racks, blades)
  - Usually owned and used by a single organization

Power and cooling

- Clusters need lots of power
  - Example: 140 Watts per server
  - Rack with 32 servers: 4.5kW (needs special power supply!)
  - Most of this power is converted into heat

- Large clusters need massive cooling
  - 4.5kW is about 3 space heaters
  - And that's just one rack!

Scaling up

PC

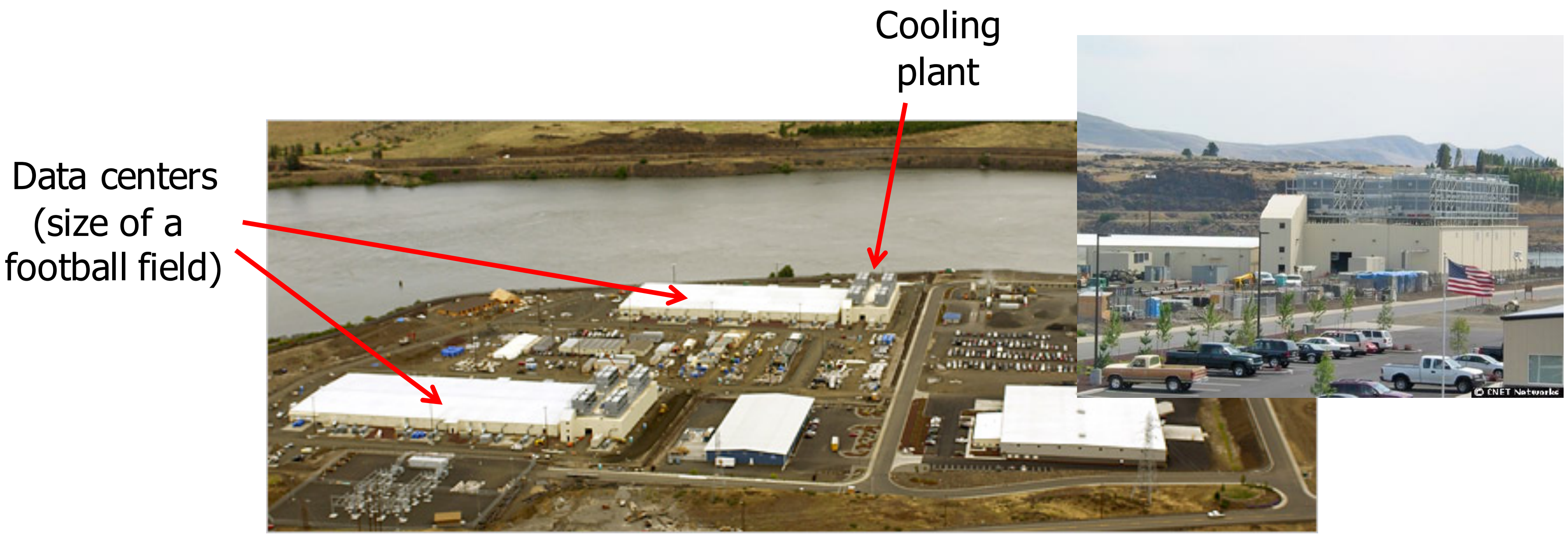
Server

Cluster

Data center

- What if your cluster is too big (hot, power hungry) to fit into your office building?
  - Build a separate building for the cluster
  - Building can have lots of cooling and power
  - Result: Data center

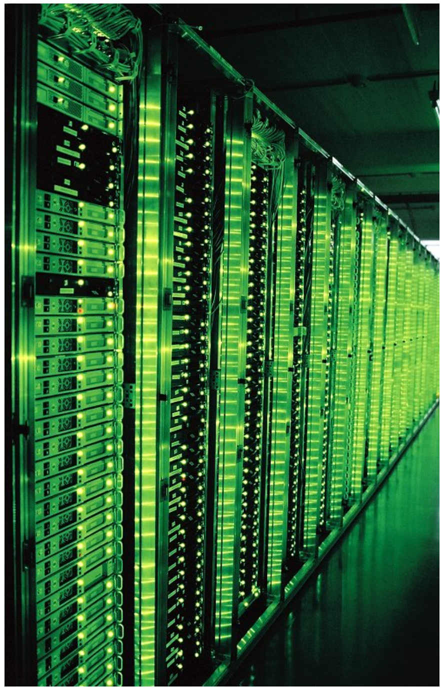
What does a data center look like?



Google data center in The Dalles, Oregon

- A warehouse-sized computer
  - A single data center can easily contain 10,000 racks with 100 cores in each rack (1,000,000 cores total)

What's in a data center?



Source: 1&1

- Hundreds or thousands of racks

What's in a data center?

Source: 1&1

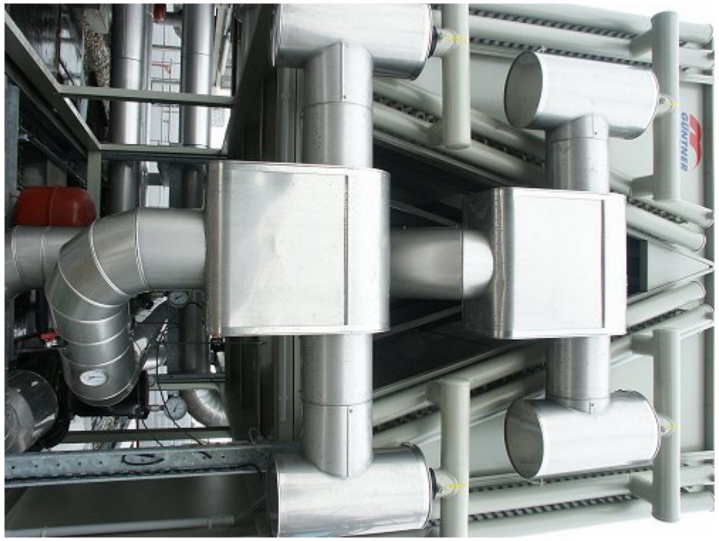
- Massive networking

What's in a data center?

Source: 1&1

- Emergency power supplies

What's in a data center?



Source: 1&1

- Massive cooling

Energy matters!

Company	Servers	Electricity	Cost
eBay	16K	~$0.6*10^5$ MWh	~$3.7M/yr
Akamai	40K	~$1.7*10^5$ MWh	~$10M/yr
Rackspace	50K	~$2*10^5$ MWh	~$12M/yr
Microsoft	>200K	>$6*10^5$ MWh	>$36M/yr
Google	>500K	>$6.3*10^5$ MWh	>$38M/yr
USA (2006)	10.9M	$610*10^5$ MWh	$4.5B/yr

Source: Qureshi et al., SIGCOMM 2009

- Data centers consume a lot of energy
  - Makes sense to build them near sources of cheap electricity
  - Example: Price per KWh is 3.6ct in Idaho (near hydroelectric power), 10ct in California (long distance transmission), 18ct in Hawaii (must ship fuel)
  - Most of this is converted into heat → Cooling is a big issue!

Scaling up

PC

Server

Cluster

Data center

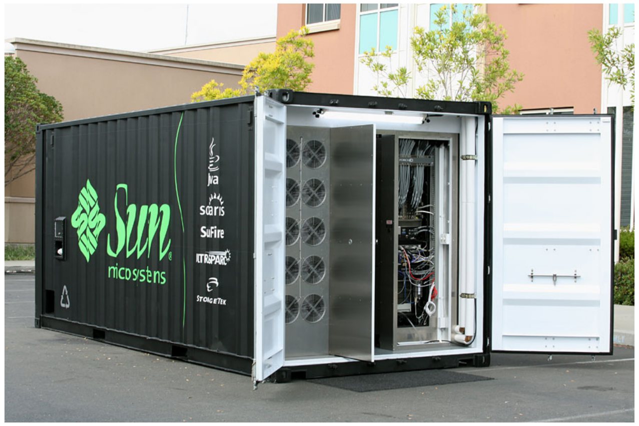
Network of data centers

- What if even a data center is not big enough?
  - Build additional data centers
  - Where? How many?

Global distribution

- Data centers are often globally distributed
  - Example above: Google data center locations (inferred)
- Why?
  - Need to be close to users (physics!)
  - Cheaper resources
  - Protection against failures

Trend: Modular data center

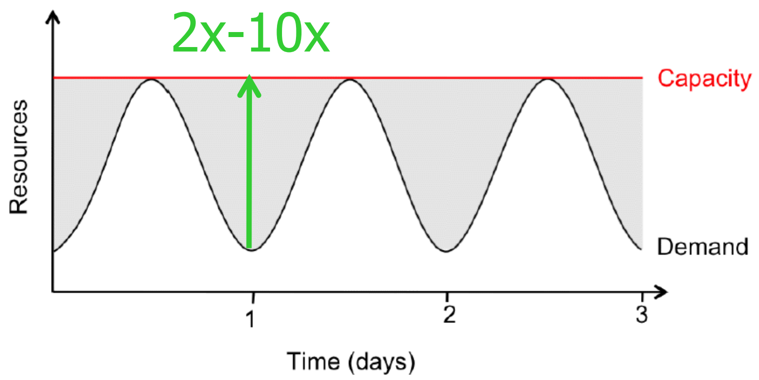


- Need more capacity? Just deploy another container!

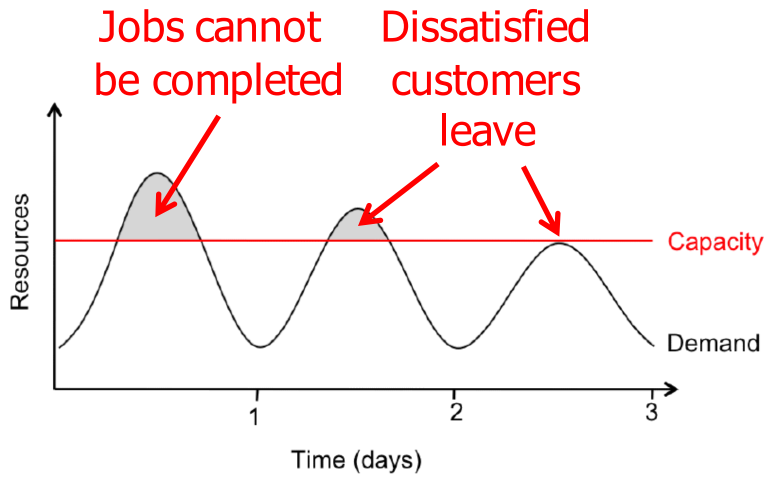
Plan for today

- Computing at scale
  - The need for scalability; scale of current services ✓
  - Scaling up: From PCs to data centers ✓
  - Problems with 'classical' scaling techniques ⇐ NEXT
- Utility computing and cloud computing
  - What are utility computing and cloud computing?
  - What kinds of clouds exist today?
  - What kinds of applications run on the cloud?
  - Virtualization: How clouds work 'under the hood'
  - Some cloud computing challenges

Problem #1: Difficult to dimension



Provisioning for the peak load



Provisioning below the peak

- Problem: Load can vary considerably
  - Peak load can exceed average load by factor 2x-10x [Why?]
  - But: Few users deliberately provision for less than the peak
  - Result: Server utilization in existing data centers ~5%-20%!!
  - Dilemma: Waste resources or lose customers!

Problem #2: Expensive

- Need to invest many $$$ in hardware
  - Even a small cluster can easily cost $100,000
  - Microsoft recently invested $499 million in a single data center
- Need expertise
  - Planning and setting up a large cluster is highly nontrivial
  - Cluster may require special software, etc.
- Need maintenance
  - Someone needs to replace faulty hardware, install software upgrades, maintain user accounts, ...

Problem #3: Difficult to scale

- Scaling up is difficult
  - Need to order new machines, install them, integrate with existing cluster - can take weeks
  - Large scaling factors may require major redesign, e.g., new storage system, new interconnect, new building (!)
- Scaling down is difficult
  - What to do with superfluous hardware?
  - Server idle power is about 60% of peak → Energy is consumed even when no work is being done
  - Many fixed costs, such as construction

Recap: Computing at scale

- Modern applications require huge amounts of processing and data
  - Measured in petabytes, millions of users, billions of objects
  - Need special hardware, algorithms, tools to work at this scale

- Clusters and data centers can provide the resources we need
  - Main difference: Scale (room-sized vs. building-sized)
  - Special hardware; power and cooling are big concerns

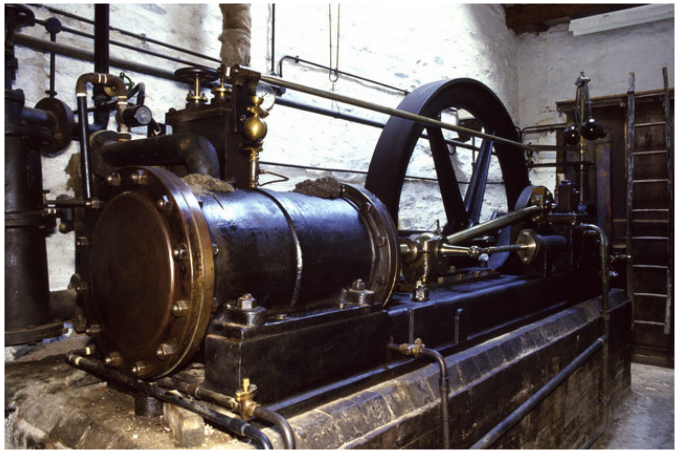
- Clusters and data centers are not perfect
  - Difficult to dimension; expensive; difficult to scale

Plan for today

- Computing at scale
  - The need for scalability; scale of current services ✓
  - Scaling up: From PCs to data centers ✓
  - Problems with 'classical' scaling techniques ✓
- Utility computing and cloud computing
  - What are utility computing and cloud computing? NEXT
  - What kinds of clouds exist today?
  - What kinds of applications run on the cloud?
  - Virtualization: How clouds work 'under the hood'
  - Some cloud computing challenges

The power plant analogy

Waterwheel at the Neuhausen ob Eck Open-Air Museum

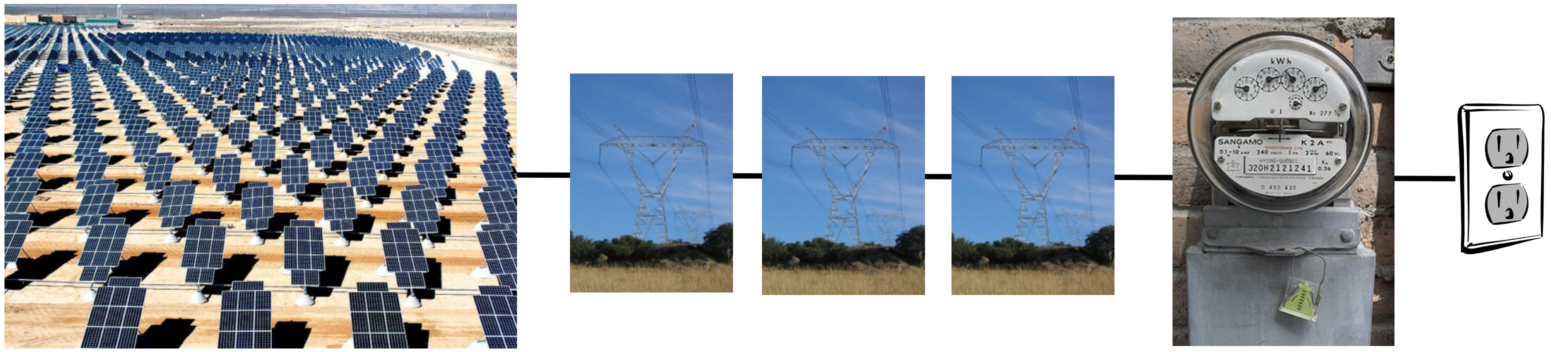

Steam engine at Stott Park Bobbin Mill

- It used to be that everyone had their own power source
    - Challenges are similar to the cluster: Needs large up-front investment, expertise to operate, difficult to scale up/down...

Scaling the power plant

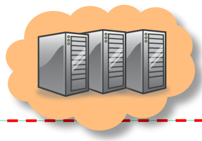
- Then people started to build large, centralized power plants with very large capacity...

Metered usage model



- Power plants are connected to customers by a network
- Usage is metered, and everyone (basically) pays only for what they actually use

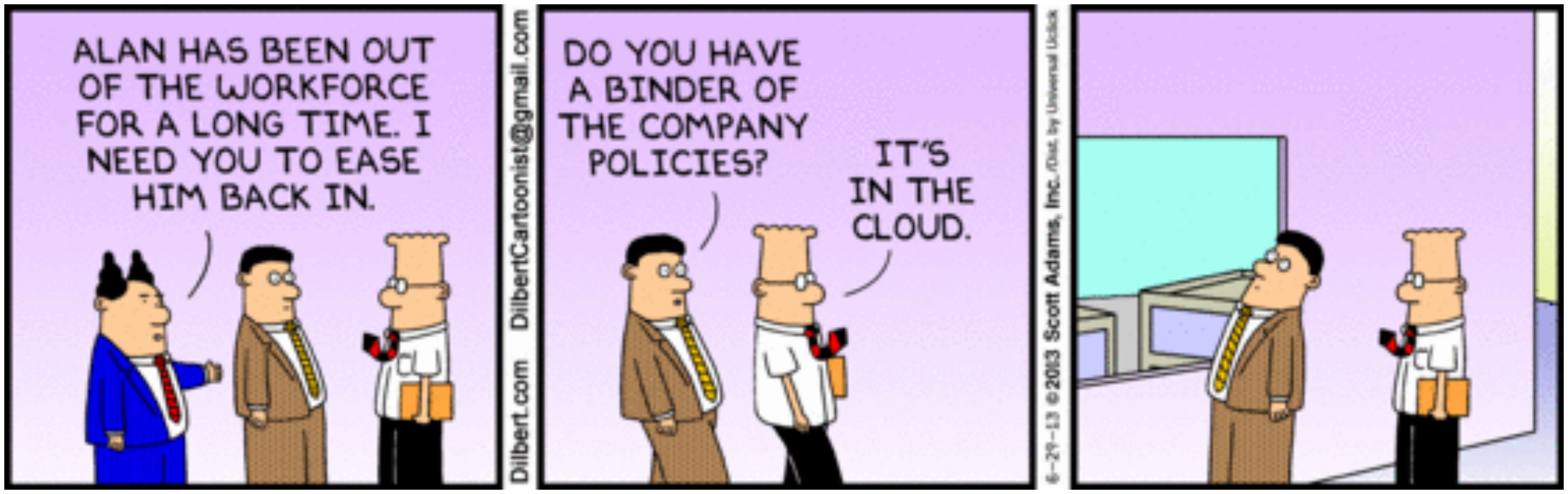
Why is this a good thing?



Electricity	Computing
Economies of scale	
Cheaper to run one big power plant than many small ones	Cheaper to run one big data center than many small ones
Statistical multiplexing	
High utilization!	High utilization!
No up-front commitment	
No investment in generator; pay-as-you-go model	No investment in data center; pay-as-you-go model
Scalability	
Thousands of kilowatts available on demand; add more within seconds	Thousands of computers available on demand; add more within seconds

What is cloud computing?

http://www.dilbert.com/fast/2013-06-29/


ALAN HAS BEEN OUT OF THE WORKFORCE FOR A LONG TIME. I NEED YOU TO EASE HIM BACK IN.
DO YOU HAVE A BINDER OF THE COMPANY POLICIES?
IT'S IN THE CLOUD.
Dilbert.com DilbertCartoonist@gmail.com
6-29-13 © 2013 Scott Adams, Inc./Dist. by Universal Uclick

What is cloud computing?

The interesting thing about Cloud Computing is that we've redefined Cloud Computing to include everything that we already do.... I don't understand what we would do differently in the light of Cloud Computing other than change the wording of some of our ads.

Larry Ellison, quoted in the Wall Street Journal, September 26, 2008

A lot of people are jumping on the [cloud] bandwagon, but I have not heard two people say the same thing about it. There are multiple definitions out there of "the cloud".

Andy Isherwood, quoted in ZDnet News, December 11, 2008

So what is it, really?

- According to NIST:

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

- Essential characteristics:
  - On-demand self service
  - Broad network access
  - Resource pooling
  - Rapid elasticity
  - Measured service

Other terms you may have heard

- Utility computing
  - The service being sold by a cloud
  - Focuses on the business model (pay-as-you-go), similar to classical utility companies
- The Web
  - The Internet's information sharing model
  - Some web services run on clouds, but not all
- The Internet
  - A network of networks.
  - Used by the web; connects (most) clouds to their customers

Plan for today

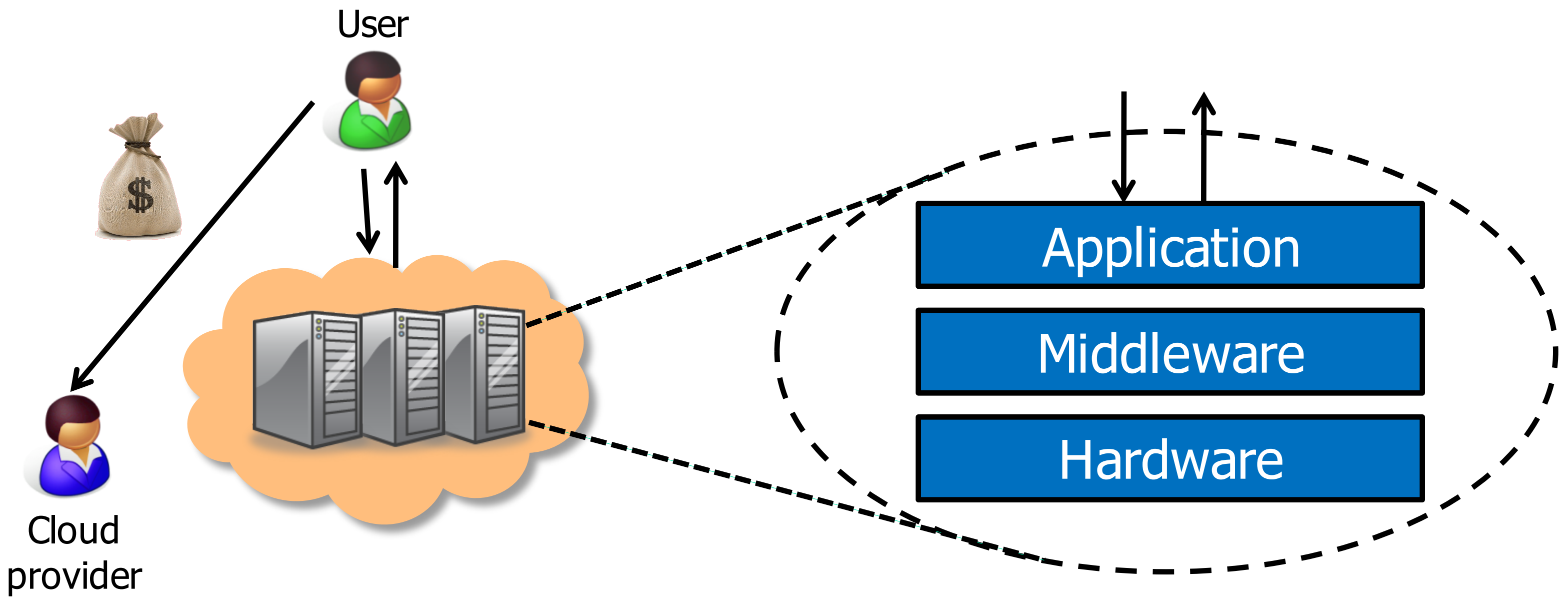
- Computing at scale
  - The need for scalability; scale of current services ✓
  - Scaling up: From PCs to data centers ✓
  - Problems with 'classical' scaling techniques ✓
- Utility computing and cloud computing
  - What are utility computing and cloud computing? ✓
  - What kinds of clouds exist today? NEXT
  - What kinds of applications run on the cloud?
  - Virtualization: How clouds work 'under the hood'
  - Some cloud computing challenges

Everything as a Service

- What kind of service does the cloud provide?
  - Does it offer an entire application, or just resources?
  - If resources, what kind / level of abstraction?

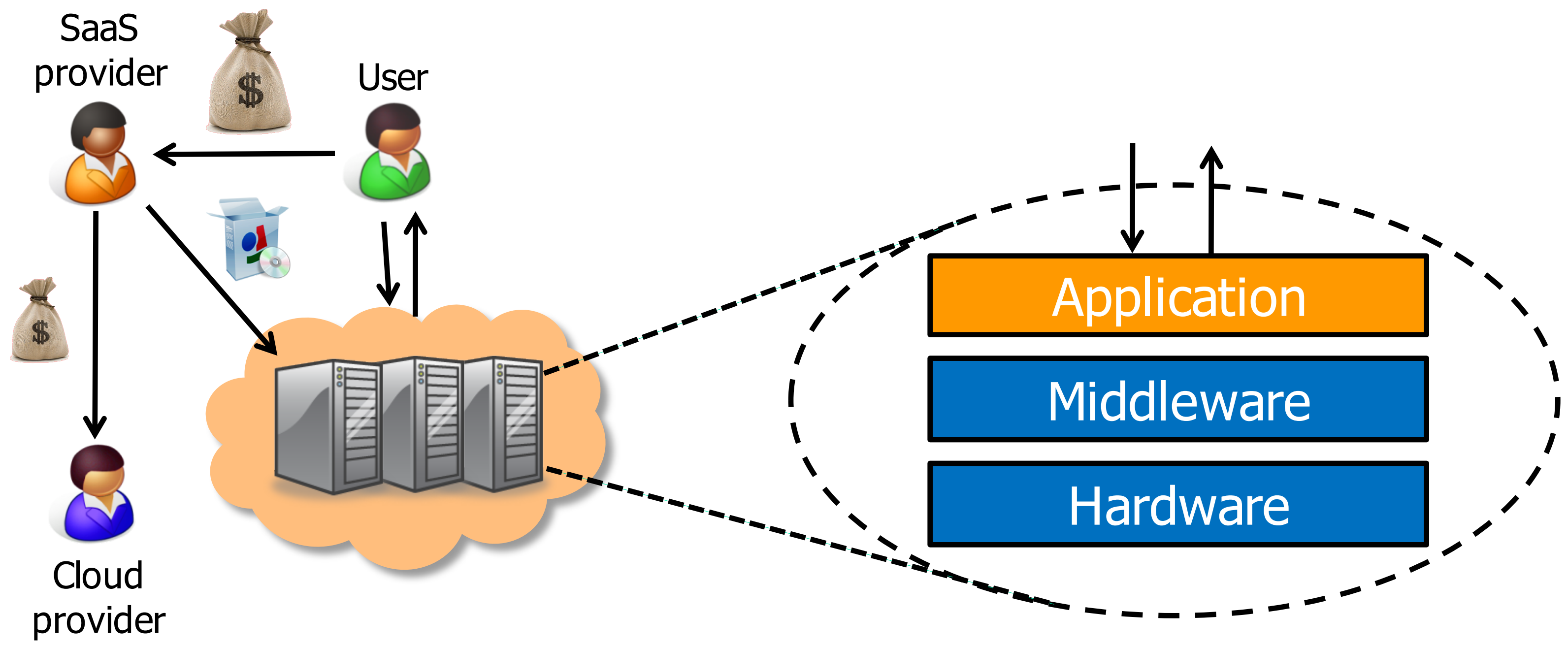
- Three types commonly distinguished:
  - Software as a service (SaaS)
    - Analogy: Restaurant. Prepares&serves entire meal, does the dishes, ...
  - Platform as a service (PaaS)
    - Analogy: Take-out food. Prepares meal, but does not serve it.
  - Infrastructure as a service (IaaS)
    - Analogy: Grocery store. Provides raw ingredients.
  - Other xaaS types have been defined, but are less common
    - Desktop, Backend, Communication, Network, Monitoring, ...

Software as a Service (SaaS)



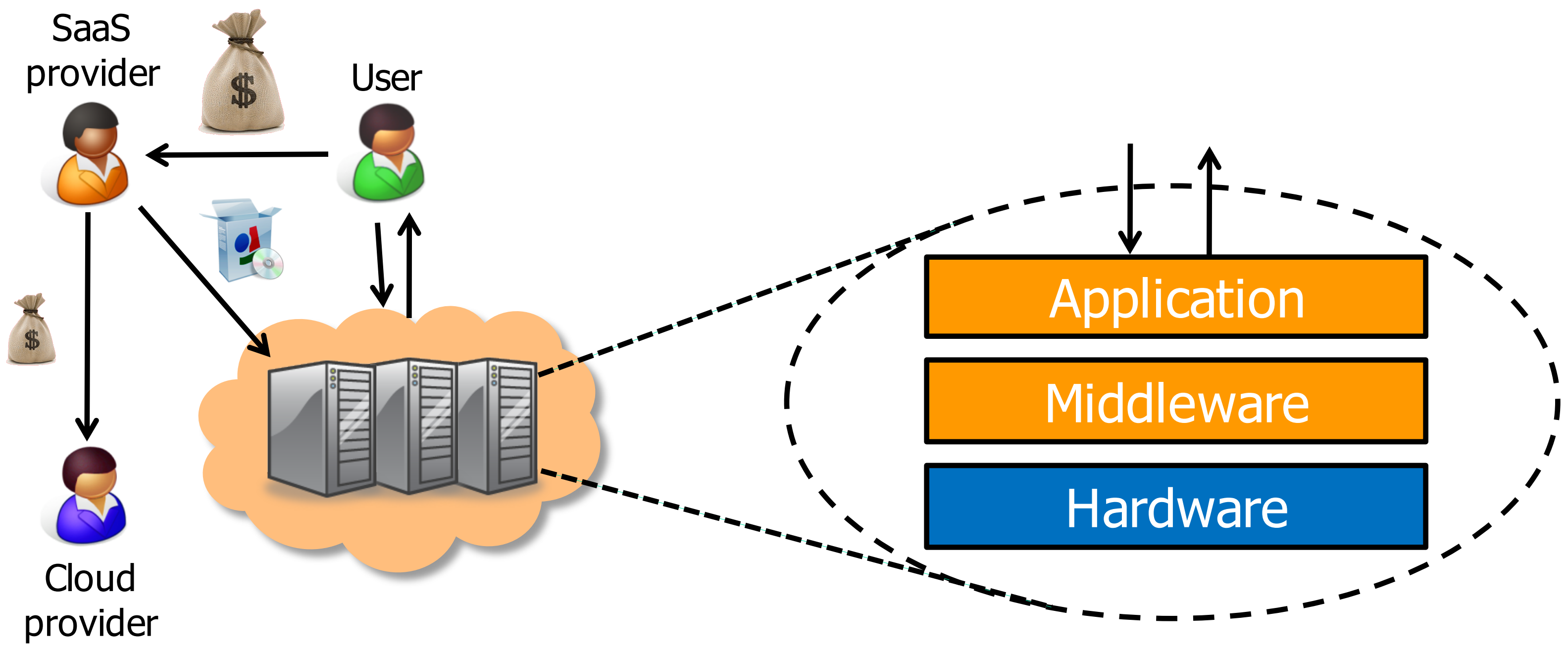
- Cloud provides an entire application
  - Word processor, spreadsheet, CRM software, calendar...
  - Customer pays cloud provider
  - Example: Google Apps, Salesforce.com

Platform as a Service (PaaS)



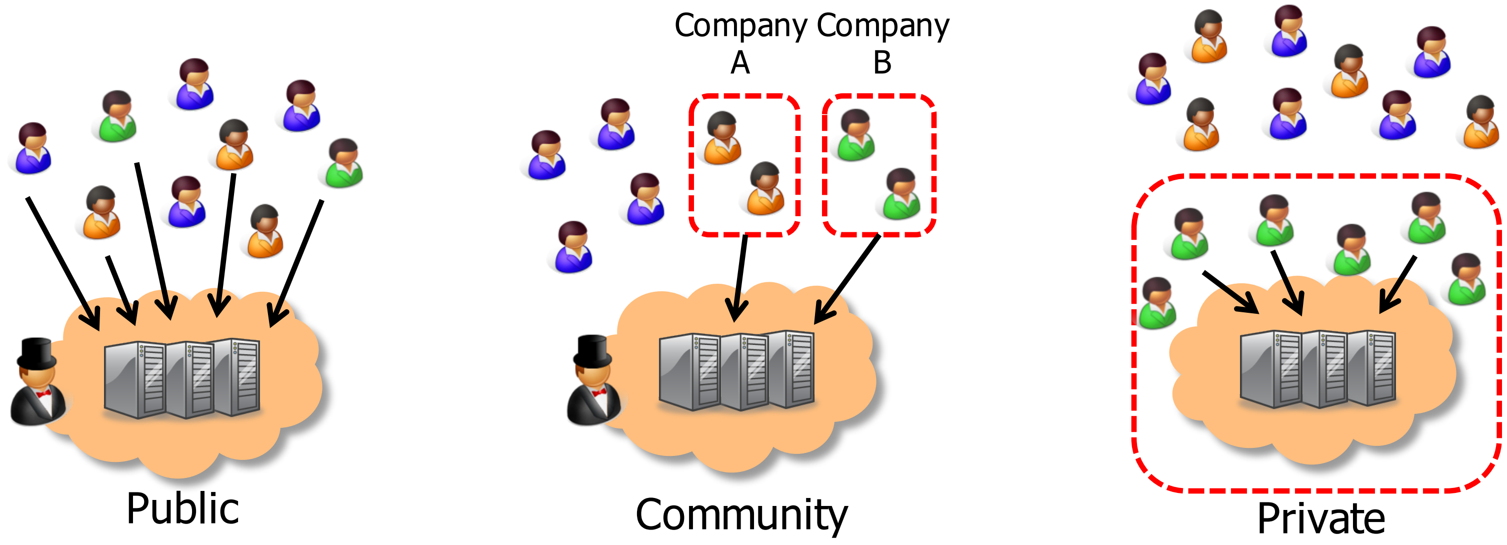
- Cloud provides middleware/infrastructure
  - For example, Microsoft Common Language Runtime (CLR)
  - Customer pays SaaS provider for the service; SaaS provider pays the cloud for the infrastructure
  - Example: Windows Azure, Google App Engine

Infrastructure as a Service (IaaS)



- Cloud provides raw computing resources
  - Virtual machine, blade server, hard disk, ...
  - Customer pays SaaS provider for the service; SaaS provider pays the cloud for the resources
  - Examples: Amazon Web Services, Rackspace Cloud, GoGrid

Private/hybrid/community clouds



- Who can become a customer of the cloud?
  - Public cloud: Commercial service; open to (almost) anyone. Example: Amazon AWS, Microsoft Azure, Google App Engine
  - Community cloud: Shared by several similar organizations. Example: Google's "Gov Cloud"
  - Private cloud: Shared within a single organization. Example: Internal datacenter of a large company.

Focus of this class →

Is this a 'real' cloud? →

Plan for today

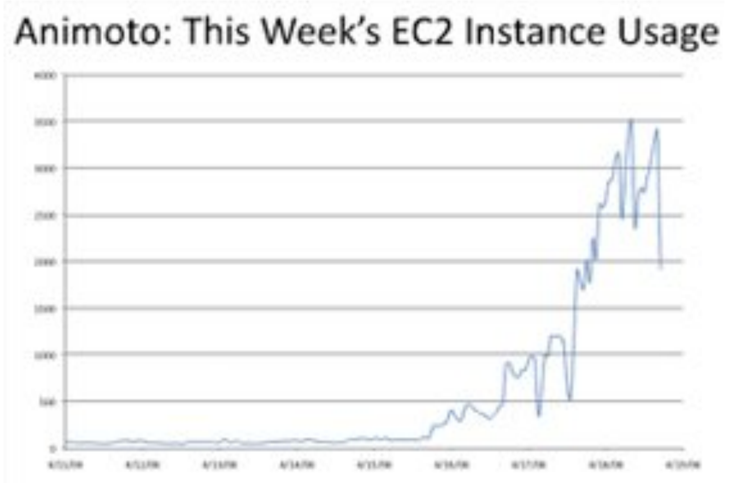
- Computing at scale
  - The need for scalability; scale of current services ✓
  - Scaling up: From PCs to data centers ✓
  - Problems with 'classical' scaling techniques ✓
- Utility computing and cloud computing
  - What are utility computing and cloud computing? ✓
  - What kinds of clouds exist today? ✓
  - What kinds of applications run on the cloud? NEXT
  - Virtualization: How clouds work 'under the hood'
  - Some cloud computing challenges

Examples of cloud applications

- Application hosting
- Backup and Storage
- Content delivery
- E-commerce
- High-performance computing
- Media hosting
- On-demand workforce
- Search engines
- Web hosting

Case study: animoto

- Animoto: Lets users create videos from their own photos/music
  - Auto-edits photos and aligns them with the music, so it "looks good"



Source: Jeff Bezos' talk at Stanford on 4/19/08

- Built using Amazon EC2+S3+SQS
- Released a Facebook app in mid-April 2008
  - More than 750,000 people signed up within 3 days
  - EC2 usage went from 50 machines to 3,500 (x70 scalability!)

http://animoto.com/blog/company/amazon-com-ceo-jeff-bezos-on-animoto/

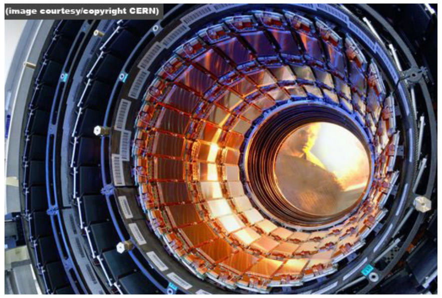
Case study: The Washington Post

http://aws.amazon.com/solutions/case-studies/washington-post/

- March 19, 2008: Hillary Clinton's official White House schedule released to the public
  - 17,481 pages of non-searchable, low-quality PDF
  - Very interesting to journalists, but would have required hundreds of man-hours to evaluate
  - Peter Harkins, Senior Engineer at The Washington Post: Can we make that data available more quickly, ideally within the same news cycle?
  - Tested various Optical Character Recognition (OCR) programs; estimated required speed
  - Launched 200 EC2 instances; project was completed within nine hours (!) using 1,407 hours of VM time ($144.62)
  - Results available on the web only 26 hours after the release

Other examples

- DreamWorks is using the Cerelink cloud to render animation movies
  - Cloud was already used to render parts of *Shrek Forever After* and *How to Train your Dragon*



- CERN is working on a "science cloud" to process experimental data
- Virgin atlantic is hosting their new travel portal on Amazon AWS

Recap: Utility/cloud computing

- Why is cloud computing attractive?
  - Analogy to 'classical' utilities (electricity, water, ...)
  - No up-front investment (pay-as-you-go model)
  - Low price due to economies of scale
  - Elasticity - can quickly scale up/down as demand varies
- Different types of clouds
  - SaaS, PaaS, IaaS; public/private/community clouds
- What runs on the cloud?
  - Many potential applications: Application hosting, backup/storage, scientific computing, content delivery, ...
  - Not yet suitable for certain applications (sensitive data, compliance requirements)

Is the cloud good for everything?

- No.
- Sometimes it is problematic, e.g., because of auditability requirements
- Example: Processing medical records
  - HIPAA (Health Insurance Portability and Accountability Act) privacy and security rule
- Example: Processing financial information
  - Sarbanes-Oxley act
- Would you put your medical data on the cloud?
  - Why / why not?

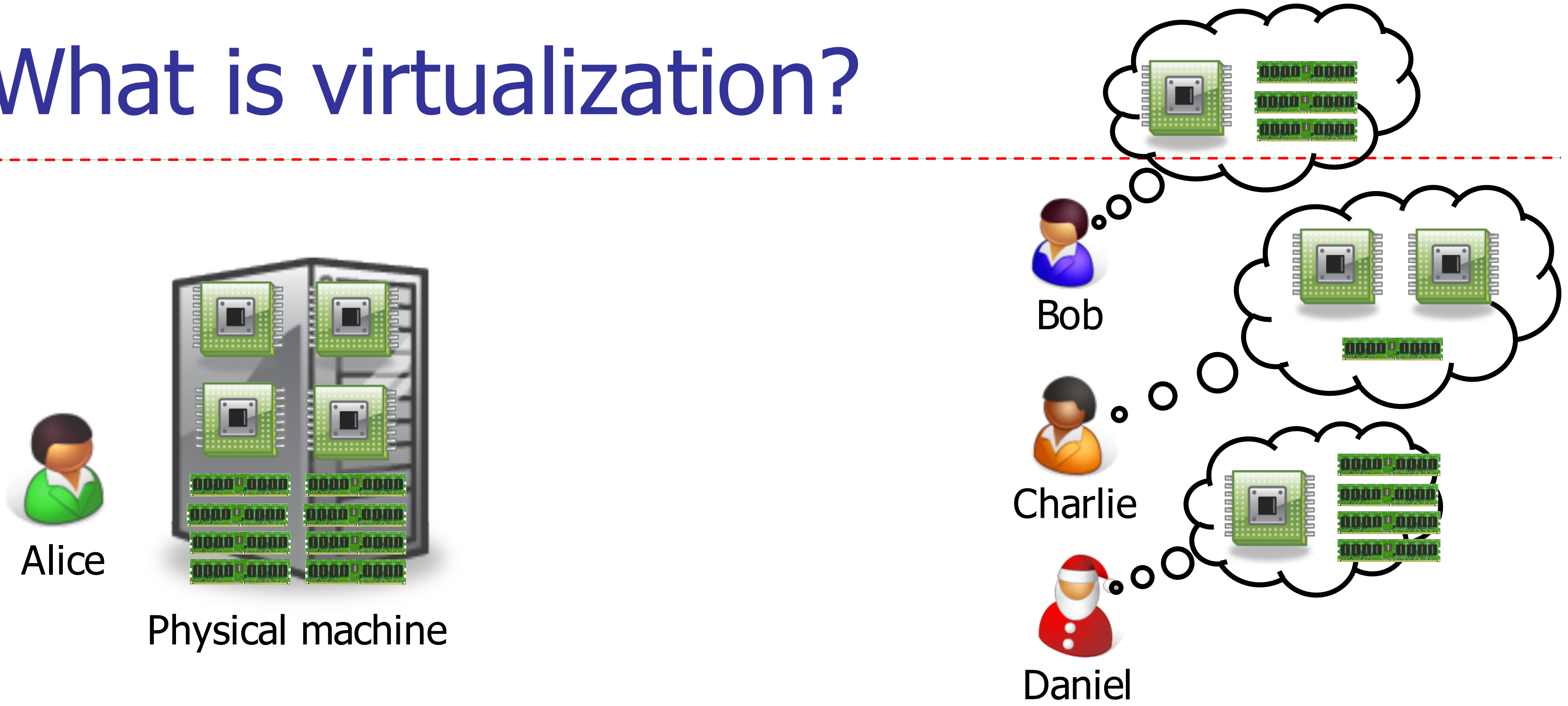
Recap: Cloud applications

- Clouds are good for many things...
  - Applications that involve large amounts of computation, storage, bandwidth
  - Especially when lots of resources are needed quickly (Washington Post example) or load varies rapidly (TicketLeap example)
- ... but not for all things

Plan for today

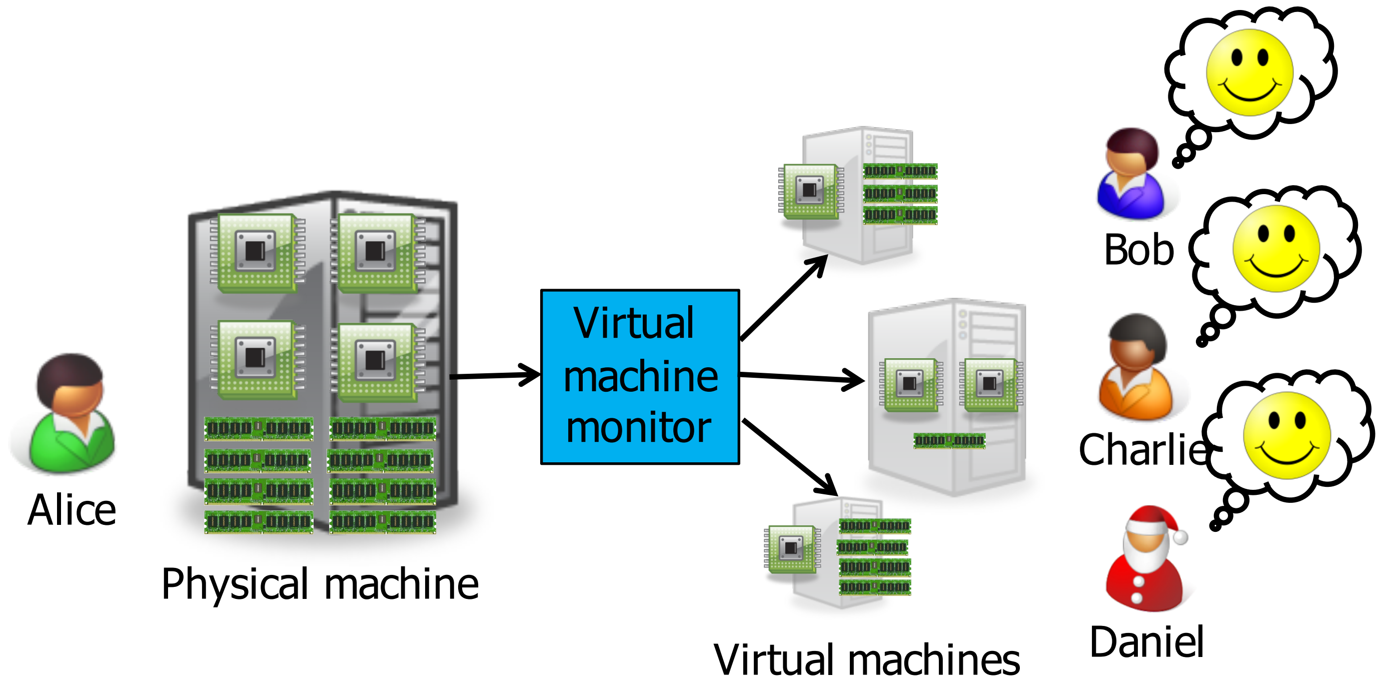
- Computing at scale
  - The need for scalability; scale of current services ✓
  - Scaling up: From PCs to data centers ✓
  - Problems with 'classical' scaling techniques ✓
- Utility computing and cloud computing
  - What are utility computing and cloud computing? ✓
  - What kinds of clouds exist today? ✓
  - What kinds of applications run on the cloud? ✓
  - Virtualization: How clouds work 'under the hood' ← NEXT
  - Some cloud computing challenges

What is virtualization?



- Suppose Alice has a machine with 4 CPUs and 8 GB of memory, and three customers:
  - Bob wants a machine with 1 CPU and 3GB of memory
  - Charlie wants 2 CPUs and 1GB of memory
  - Daniel wants 1 CPU and 4GB of memory
- What should Alice do?

What is virtualization?

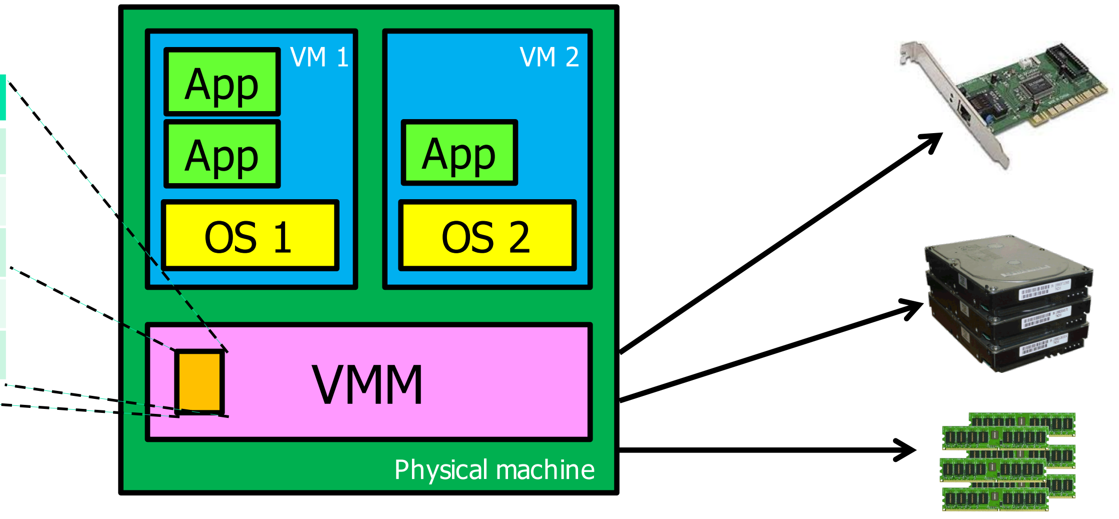


- Alice can sell each customer a virtual machine (VM) with the requested resources
  - From each customer's perspective, it appears as if they had a physical machine all by themselves (isolation)

How does it work?

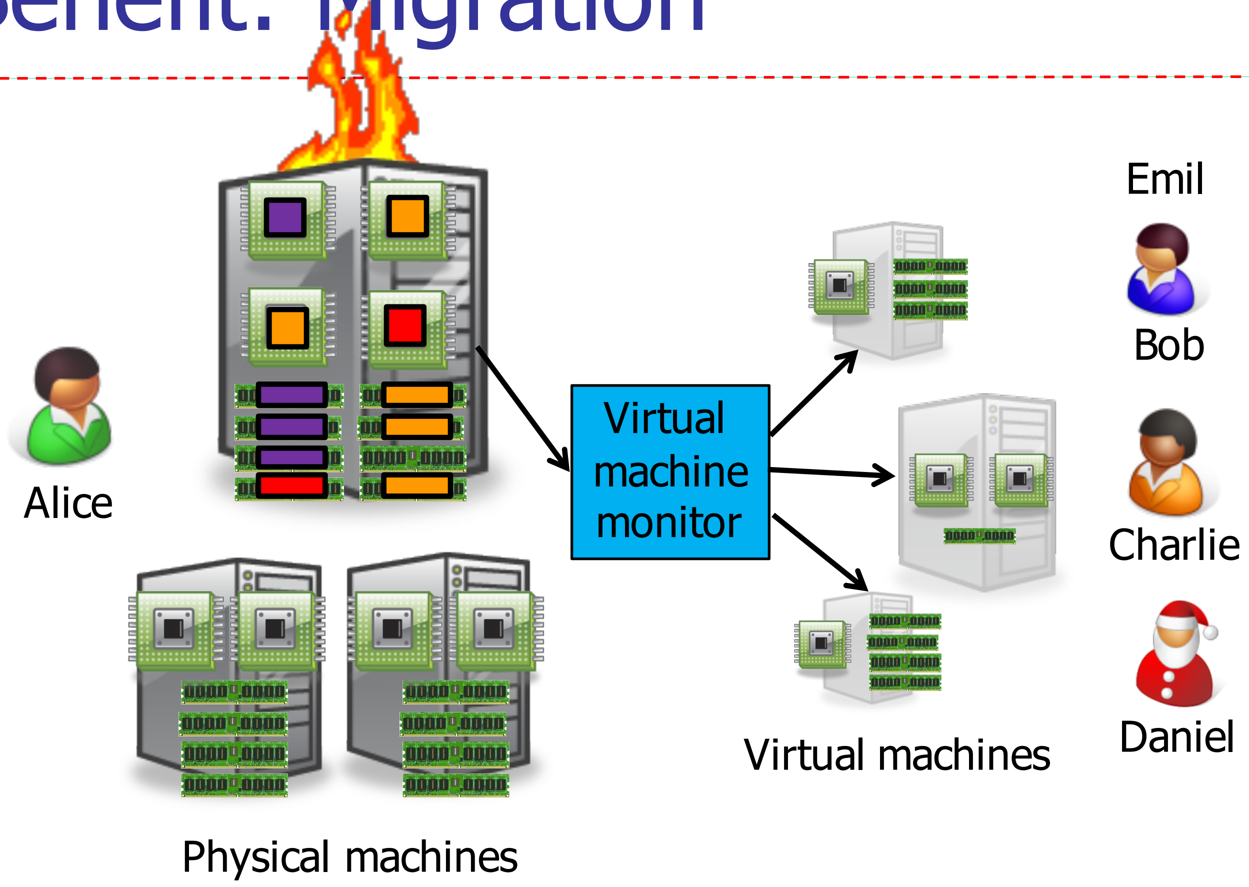
VM	Virt	Phys
1	0-99	0-99
1	299-399	100-199
2	0-99	300-399
2	200-299	500-599
2	600-699	400-499

Translation table



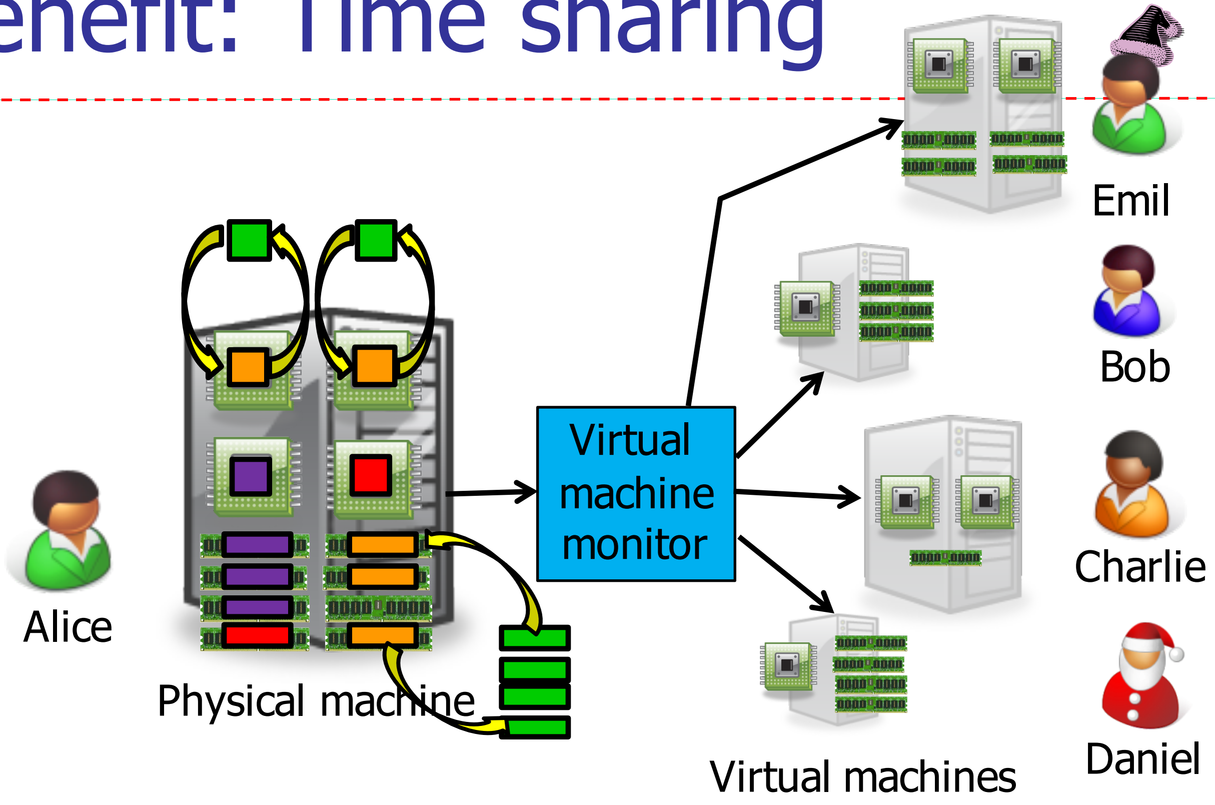
- Resources (CPU, memory, ...) are virtualized
  - VMM ("Hypervisor") has translation tables that map requests for virtual resources to physical resources
  - Example: VM 1 accesses memory cell #323; VMM maps this to memory cell 123.
  - For which resources does this (not) work?
  - How do VMMs differ from OS kernels?

Benefit: Migration



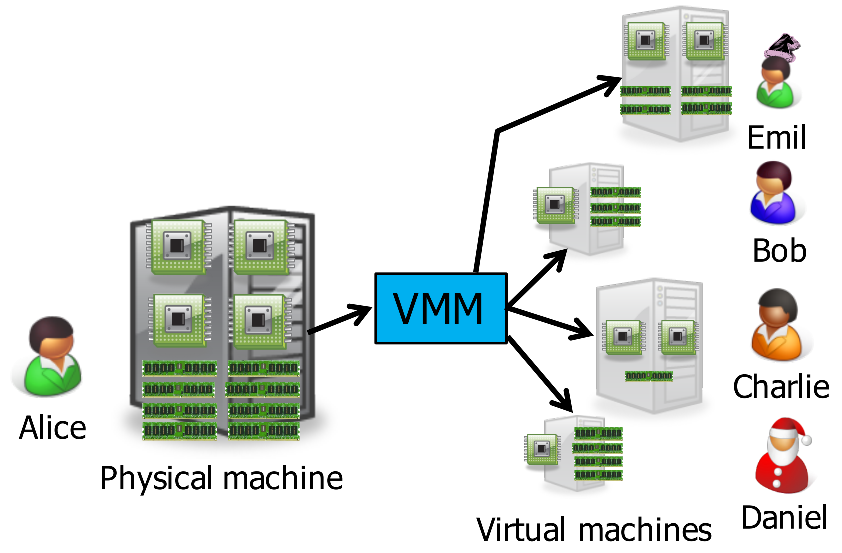
- What if the machine needs to be shut down?
  - e.g., for maintenance, consolidation, ...
  - Alice can migrate the VMs to different physical machines without any customers noticing

Benefit: Time sharing



- What if Alice gets another customer?
  - Multiple VMs can time-share the existing resources
  - Result: Alice has more virtual CPUs and virtual memory than physical resources (but not all can be active at the same time)

Benefit and challenge: Isolation



- Good: Emil can't access Charlie's data
- Bad: What if the load suddenly increases?
  - Example: Emil's VM shares CPUs with Charlie's VM, and Charlie suddenly starts a large compute job
  - Emil's performance may decrease as a result
  - VMM can move Emil's software to a different CPU, or migrate it to a different machine

Recap: Virtualization in the cloud

- Gives cloud provider a lot of flexibility
  - Can produce VMs with different capabilities
  - Can migrate VMs if necessary (e.g., for maintenance)
  - Can increase load by overcommitting resources
- Provides security and isolation
  - Programs in one VM cannot influence programs in another
- Convenient for users
  - Complete control over the virtual 'hardware' (can install own operating system own applications, ...)
- But: Performance may be hard to predict
  - Load changes in other VMs on the same physical machine may affect the performance seen by the customer

Plan for today

- Computing at scale
  - The need for scalability; scale of current services ✓
  - Scaling up: From PCs to data centers ✓
  - Problems with 'classical' scaling techniques ✓
- Utility computing and cloud computing
  - What are utility computing and cloud computing? ✓
  - What kinds of clouds exist today? ✓
  - What kinds of applications run on the cloud? ✓
  - Virtualization: How clouds work 'under the hood' ✓
  - Some cloud computing challenges NEXT

10 obstacles and opportunities

1. Availability
   - What happens to my business if there is an outage in the cloud?

Service	Duration	Date
S3	6-8 hrs	7/20/08
AppEngine	5 hrs	6/17/08
Gmail	1.5 hrs	8/11/08
Azure	22 hrs	3/13/09
Intuit	36 hrs	6/16/10
EBS	>3 days	4/21/11
ECC	~2 hrs	6/30/12

Some recent cloud outages

2. Data lock-in
   - How do I move my data from one cloud to another?

3. Data confidentiality and auditability
   - How do I make sure that the cloud doesn't leak my confidential data?
   - Can I comply with regulations like HIPAA and Sarbanes/Oxley?

10 obstacles and opportunities

4. Data transfer bottlenecks
   - How do I copy large amounts of data from/to the cloud?
   - Example: 10 TB from UC Berkeley to Amazon in Seattle, WA
   - Motivated Import/Export feature on AWS

Method	Time
Internet (20Mbps)	45 days
FedEx	1 day

Time to transfer 10TB [AF10]

5. Performance unpredictability
   - Example: VMs sharing the same disk → I/O interference
   - Example: HPC tasks that require coordinated scheduling

Primitive	Mean perf.	Std dev
Memory bandwidth	1.3GB/s	0.05GB/s (4%)
Disk bandwidth	55MB/s	9MB/s (16%)

Performance of 75 EC2 instances in benchmarks

10 obstacles and opportunities

6. Scalable storage
    - Cloud model (short-term usage, no up-front cost, infinite capacity on demand) does not fit persistent storage well
7. Bugs in large distributed systems
    - Many errors cannot be reproduced in smaller configs
8. Scaling quickly
    - Problem: Boot time; idle power
    - Fine-grain accounting?

10 obstacles and opportunities

9. Reputation fate sharing
    - One customer's bad behavior can affect the reputation of others using the same cloud
    - Example: Spam blacklisting, FBI raid after criminal activity
10. Software licensing
    - What if licenses are for specific computers?
        - Example: Microsoft Windows
    - How to scale number of licenses up/down?
        - Need pay-as-you-go model as well

Plan for today

- Scalable computing
  - The need for scalability; scale of current services ✓
  - Scaling up: From PCs to data centers ✓
  - Problems with 'classical' scaling techniques ✓
- Utility computing and cloud computing
  - What are utility computing and cloud computing? ✓
  - What kinds of clouds exist today? ✓
  - What kinds of applications run on the cloud? ✓
  - Virtualization: How clouds work 'under the hood' ✓
  - Some cloud computing challenges ✓

Plan for today

- A brief history of cloud computing NEXT
- Introduce one specific commercial cloud
  - Amazon Web Services (AWS)
  - Elastic Compute Cloud (EC2)
  - Elastic Block Storage (EBS)
  - Other services: Mechanical Turk, CloudFront, ...
  - Next time: S3 and SimpleDB

History: The early days

- Cloud computing: A new term for a concept that has been around since the 1960s

- Who invented it?
- No agreement. Some candidates:
  - John McCarthy (Stanford professor and inventor of Lisp; proposed the 'service bureau' model in 1961)
  - J.C.R. Licklider (contributed key ideas to ARPANET; published a memo on the "Intergalactic Computer Network" in 1963)
  - Douglas Parkhill (published a book on "The Challenge of the Computer Utility" in 1966)

History: Becoming a cloud provider

Technology	Cost in medium DC (~1,000 servers)	Cost in large DC (~50,000 servers)	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/admin	>1,000 servers/admin	7.1

Source: James Hamilton's Keynote, LADIS 2008

- Early 2000s: Phenomenal growth of web services
  - Many large Internet companies deploy huge data centers, develop scalable software infrastructure to run them
  - Due to economies of scale, these companies were now able to run computation very cheaply
  - What else can we do with this?

History: Incentives

- Idea: Use your existing data center to provide cloud services
- Why is this a good idea?
- Make a lot of money
  - Price advantage of 3x-7x → Can offer services much cheaper than medium-size company and still make profit
- Leverage existing investment
  - New revenue stream at low incremental cost (example: many Amazon AWS technologies were initially developed for Amazon's internal operations)
- Defend a franchise
  - Example: Microsoft enterprise apps → Microsoft Azure

History: Incentives (continued)

- Attack an incumbent
  - Company with requisite datacenter may want to establish a 'beach head' before a '800 pound gorilla' emerges
- Leverage existing customer relationships
  - IT service organizations like IBM Global Services have extensive customer relationships; provide anxiety-free migration path to existing customers
- Become a platform
  - Example: Facebook's initiative to enable plug-in applications is a great fit for cloud computing

History: The pioneers

- Jul 2002: Amazon Web Services launched
  - Third-party sites can search and display products from Amazon's web site, add items to Amazon shopping carts
  - Available through XML and SOAP
- Mar 2006: Amazon S3 launched
  - Innovative 'pay-per-use' pricing model, which is now the standard in cloud computing
  - Cheaper than many small/medium storage solutions: $0.15/GB/month of storage, $0.20/GB/month for traffic
  - Amazon no longer a pure retailer, entering technology space
- Aug 2006: EC2 launched
  - Core computing infrastructure becomes available

History: Wide-spread adoption

- Apr 2008: Google App Engine launched
  - Same building blocks Google uses for its own applications: Bigtable and GFS for storage, automatic scaling and load balancing, ...
- Nov 2009: Windows Azure Beta launched
  - Becomes generally available in 21 countries in Feb 2010

Plan for today

- A brief history of cloud computing ✓
- Introduce one specific commercial cloud NEXT
  - Amazon Web Services (AWS)
  - Elastic Compute Cloud (EC2)
  - Elastic Block Storage (EBS)
  - Other services: Mechanical Turk, CloudFront, ...
  - Next time: S3 and SimpleDB

Insert your favorite
cloud here

Why Amazon AWS and not $\boxed{}$?

- Amazon is only one of several cloud providers
  - Others include Microsoft Azure, Google App Engine, ...
- But there is no common standard (yet)
  - App Engine is PaaS and supports Java/JVM, Go or Python
  - Azure is PaaS and supports .NET/CLR
  - AWS is PaaS/IaaS and supports IA-32 virtual machines
- So we have to pick one specific provider
  - Amazon AWS is going to be used as an example

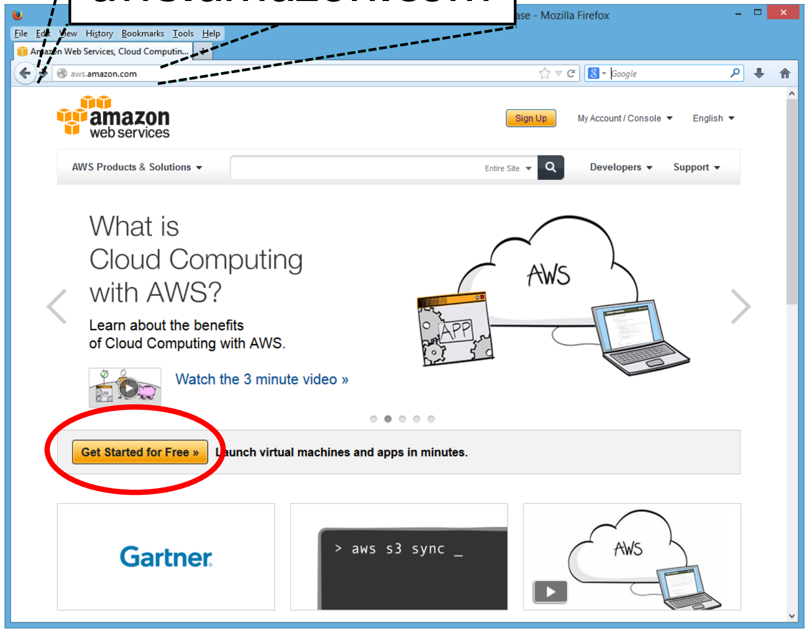
What is Amazon AWS?

- Amazon Web Services (AWS) provides a number of different services, including:
  - Amazon Elastic Compute Cloud (EC2)
    Virtual machines for running custom software
  - Amazon Simple Storage Service (S3)
    Simple key-value store, accessible as a web service
  - Amazon SimpleDB
    Simple distributed database
  - Amazon Elastic MapReduce
    Scalable MapReduce computation
  - Amazon Mechanical Turk (MTurk)
    A 'marketplace for work'
  - Amazon CloudFront
    Content delivery network
  - ...

Used for the projects

Setting up an AWS account

aws.amazon.com

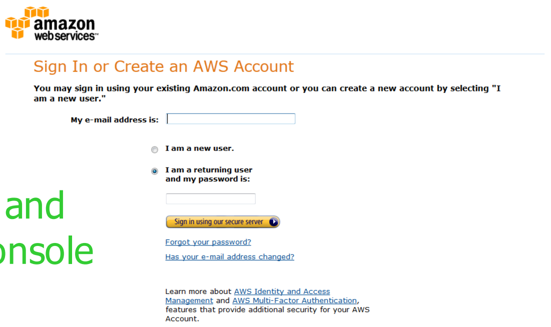

aws.amazon.com
Sign Up
My Account / Console
English
AWS Products & Solutions
Developers
Support
What is Cloud Computing with AWS?
Learn about the benefits of Cloud Computing with AWS.
Watch the 3 minute video »
Get Started for Free »
Launch virtual machines and apps in minutes.
Gartner
> aws s3 sync _
AWS

- Sign up for an account on aws.amazon.com
  - You need to choose an username and a password
  - These are for the management interface only
  - Your programs will use other credentials (RSA keypairs, access keys, ...) to interact with AWS

AWS credentials

Command-line tools
SOAP APIs

AWS web site and management console



Sign In or Create an AWS Account

You may sign in using your existing Amazon.com account or you can create a new account by selecting "I am a new user."

My e-mail address is:

( ) I am a new user.

(•) I am a returning user and my password is:

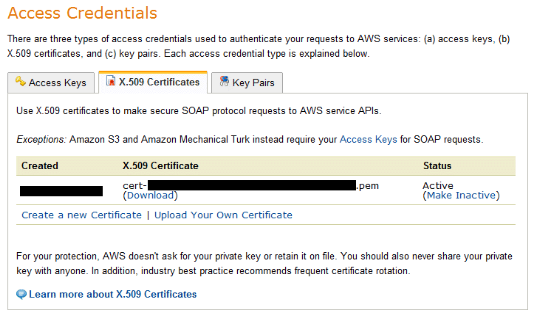
Sign in using our secure server

Forgot your password?

Has your e-mail address changed?

Learn more about AWS Identity and Access Management and AWS Multi-Factor Authentication, features that provide additional security for your AWS Account.

Sign-in credentials



Access Credentials

There are three types of access credentials used to authenticate your requests to AWS services: (a) access keys, (b) X.509 certificates, and (c) key pairs. Each access credential type is explained below.

Access Keys | X.509 Certificates | Key Pairs

Use X.509 certificates to make secure SOAP protocol requests to AWS service APIs.

Exceptions: Amazon S3 and Amazon Mechanical Turk instead require your Access Keys for SOAP requests.

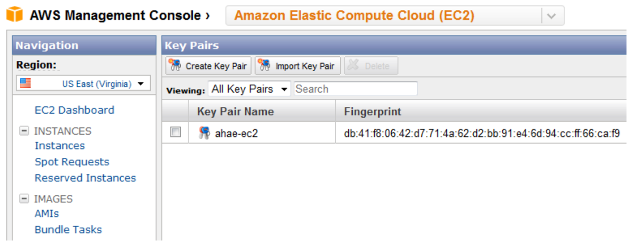
Created	X.509 Certificate	Status
[redacted]	cert-[redacted].pem (Download)	Active (Make Inactive)

Create a new Certificate | Upload Your Own Certificate

For your protection, AWS doesn't ask for your private key or retain it on file. You should also never share your private key with anyone. In addition, industry best practice recommends frequent certificate rotation.

Learn more about X.509 Certificates

X.509 certificates



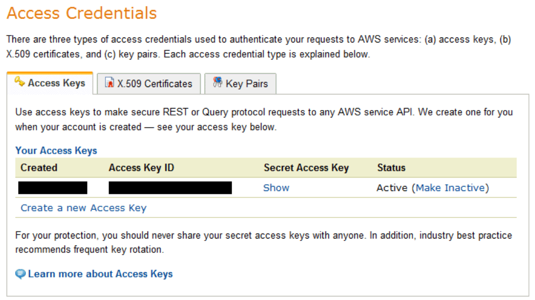
AWS Management Console › Amazon Elastic Compute Cloud (EC2)

Viewing: All Key Pairs | Search

	Key Pair Name	Fingerprint
☐	ahae-ec2	db:41:f8:06:42:d7:71:4a:62:d2:bb:91:e4:6d:94:cc:ff:66:ca:f9

Connecting to an instance (e.g., via ssh)

EC2 key pairs



Access Credentials

There are three types of access credentials used to authenticate your requests to AWS services: (a) access keys, (b) X.509 certificates, and (c) key pairs. Each access credential type is explained below.

Access Keys | X.509 Certificates | Key Pairs

Use access keys to make secure REST or Query protocol requests to any AWS service API. We create one for you when your account is created — see your access key below.

Your Access Keys

Created	Access Key ID	Secret Access Key	Status
[redacted]	[redacted]	Show	Active (Make Inactive)

Create a new Access Key

For your protection, you should never share your secret access keys with anyone. In addition, industry best practice recommends frequent key rotation.

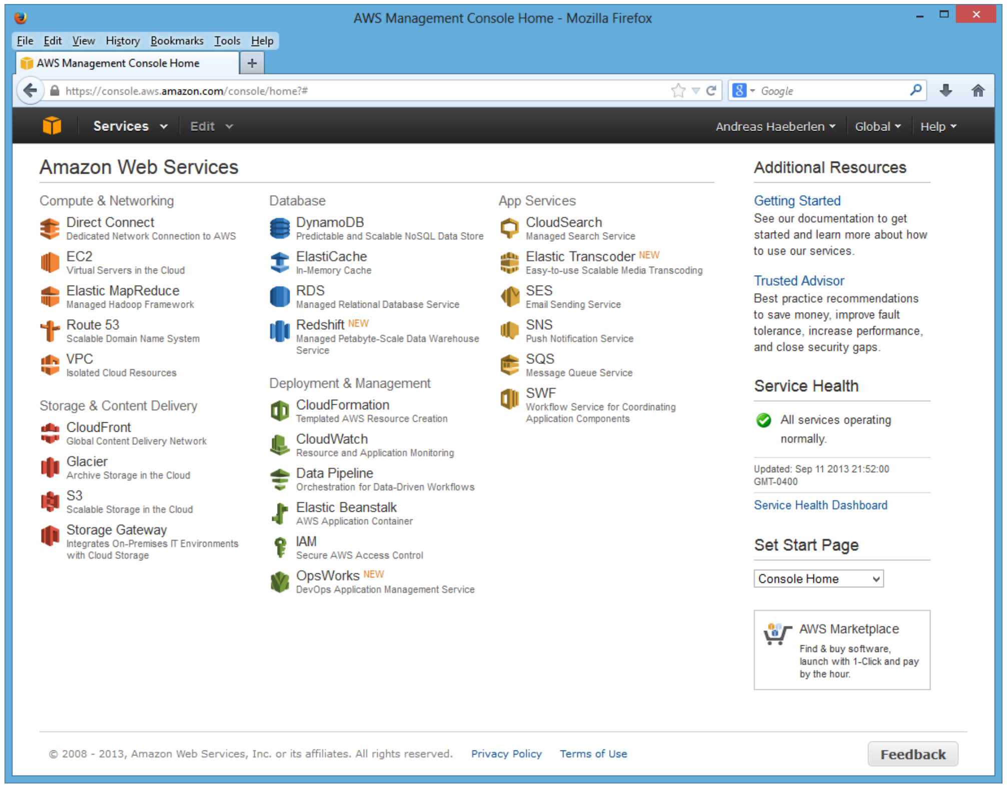
Learn more about Access Keys

Access keys

REST APIs

- Why so many different types of credentials?

The AWS management console



AWS Management Console Home - Mozilla Firefox

File Edit View History Bookmarks Tools Help

AWS Management Console Home

https://console.aws.amazon.com/console/home?#

Services Edit Andreas Haeberlen Global Help

Amazon Web Services

Compute & Networking

- Direct Connect — Dedicated Network Connection to AWS
- EC2 — Virtual Servers in the Cloud
- Elastic MapReduce — Managed Hadoop Framework
- Route 53 — Scalable Domain Name System
- VPC — Isolated Cloud Resources

Storage & Content Delivery

- CloudFront — Global Content Delivery Network
- Glacier — Archive Storage in the Cloud
- S3 — Scalable Storage in the Cloud
- Storage Gateway — Integrates On-Premises IT Environments with Cloud Storage

Database

- DynamoDB — Predictable and Scalable NoSQL Data Store
- ElastiCache — In-Memory Cache
- RDS — Managed Relational Database Service
- Redshift NEW — Managed Petabyte-Scale Data Warehouse Service

Deployment & Management

- CloudFormation — Templated AWS Resource Creation
- CloudWatch — Resource and Application Monitoring
- Data Pipeline — Orchestration for Data-Driven Workflows
- Elastic Beanstalk — AWS Application Container
- IAM — Secure AWS Access Control
- OpsWorks NEW — DevOps Application Management Service

App Services

- CloudSearch — Managed Search Service
- Elastic Transcoder NEW — Easy-to-use Scalable Media Transcoding
- SES — Email Sending Service
- SNS — Push Notification Service
- SQS — Message Queue Service
- SWF — Workflow Service for Coordinating Application Components

Additional Resources

Getting Started
See our documentation to get started and learn more about how to use our services.

Trusted Advisor
Best practice recommendations to save money, improve fault tolerance, increase performance, and close security gaps.

Service Health

All services operating normally.

Updated: Sep 11 2013 21:52:00 GMT-0400

Service Health Dashboard

Set Start Page

Console Home

AWS Marketplace
Find & buy software, launch with 1-Click and pay by the hour.

© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Feedback

- Used to control many AWS services:
  - For example, start/stop EC2 instances, create S3 buckets...

REST and SOAP

- How do your programs access AWS?
  - Via the REST or SOAP protocols
  - Example: Launch an EC2 instance, store a value in S3, ...
- Simple Object Access protocol (SOAP)
  - Not as simple as the name suggests
  - XML-based, extensible, general, standardized, but also somewhat heavyweight and verbose
  - Increasingly deprecated (e.g., for SimpleDB and EC2)
- Representational State Transfer (REST)
  - Much simpler to develop than SOAP
  - Web-specific; lack of standards

Example: REST

Invoked method

Response elements

Parameters

Credentials

```
https://sdb.amazonaws.com/?Action=PutAttributes
&DomainName=MyDomain
&ItemName=Item123
&Attribute.1.Name=Color&Attribute.1.Value=Blue
&Attribute.2.Name=Size&Attribute.2.Value=Med
&Attribute.3.Name=Price&Attribute.3.Value=0014.99
&AWSAccessKeyId=<valid_access_key>
&Version=2009-04-15
&Signature=[valid signature]
&SignatureVersion=2
&SignatureMethod=HmacSHA256
&Timestamp=2010-01-25T15%3A01%3A28-07%3A00
```

Sample request

```
<PutAttributesResponse>
<ResponseMetadata>
<StatusCode>Success</StatusCode>
<RequestId>f6820318-9658-4a9d-89f8-
b067c90904fc</RequestId>
<BoxUsage>0.0000219907</BoxUsage>
</ResponseMetadata>
</PutAttributesResponse>
```

Sample response

Source: http://awsdocs.s3.amazonaws.com/SDB/latest/sdb-dg.pdf

Example: SOAP

```
<?xml version='1.0' encoding='UTF-8'?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV='http://schemas.xmlsoap.org/soap/envelope/'
xmlns:SOAP-ENC='http://schemas.xmlsoap.org/soap/encoding/'
xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xmlns:xsd='http://www.w3.org/2001/XMLSchema'>
<SOAP-ENV:Body>
<PutAttributesRequest xmlns='http://sdb.amazonaws.com/doc/
2009-04-15'>
<Attribute><Name>a1</Name><Value>2</Value></Attribute>
<Attribute><Name>a2</Name><Value>4</Value></Attribute>
<DomainName>domain1</DomainName>
<ItemName>eID001</ItemName>
<Version>2009-04-15</Version>
</PutAttributesRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Sample request

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
<PutAttributesResponse>
<ResponseMetadata>
<RequestId>4c68e051-fe45-43b2-992a-
a24017ffe7ab</RequestId>
<BoxUsage>0.0000219907</BoxUsage>
</ResponseMetadata>
</PutAttributesResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

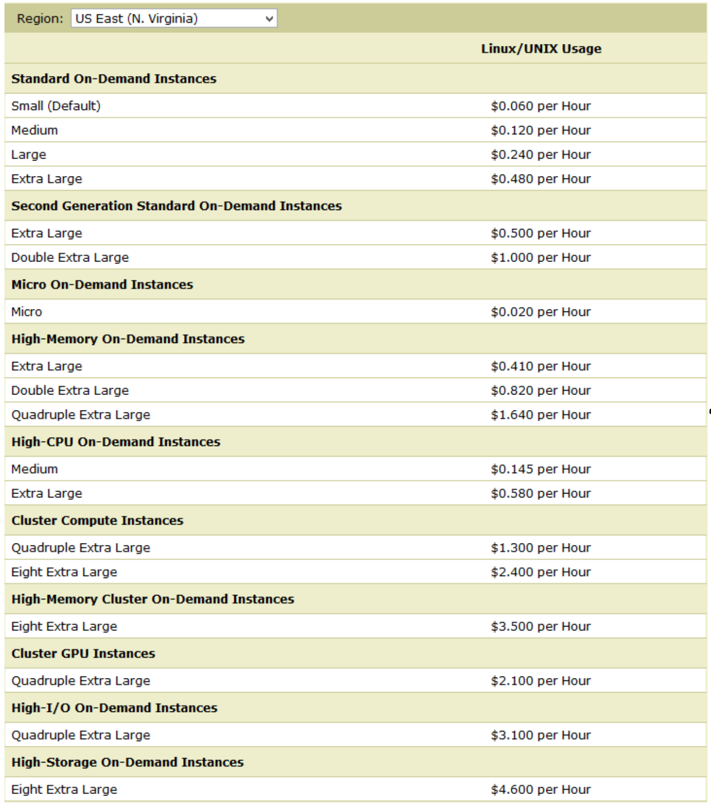
Sample response

Source: http://awsdocs.s3.amazonaws.com/SDB/latest/sdb-dg.pdf

Plan for today

- A brief history of cloud computing ✓
- Introduce one specific commercial cloud
  - Amazon Web Services (AWS) ✓
  - Elastic Compute Cloud (EC2) ← NEXT
  - Elastic Block Storage (EBS)
  - Other services: Mechanical Turk, CloudFront, ...
  - Next time: S3 and SimpleDB

What is Amazon EC2?



Region: US East (N. Virginia)

	Linux/UNIX Usage
Standard On-Demand Instances	
Small (Default)	$0.060 per Hour
Medium	$0.120 per Hour
Large	$0.240 per Hour
Extra Large	$0.480 per Hour
Second Generation Standard On-Demand Instances	
Extra Large	$0.500 per Hour
Double Extra Large	$1.000 per Hour
Micro On-Demand Instances	
Micro	$0.020 per Hour
High-Memory On-Demand Instances	
Extra Large	$0.410 per Hour
Double Extra Large	$0.820 per Hour
Quadruple Extra Large	$1.640 per Hour
High-CPU On-Demand Instances	
Medium	$0.145 per Hour
Extra Large	$0.580 per Hour
Cluster Compute Instances	
Quadruple Extra Large	$1.300 per Hour
Eight Extra Large	$2.400 per Hour
High-Memory Cluster On-Demand Instances	
Eight Extra Large	$3.500 per Hour
Cluster GPU Instances	
Quadruple Extra Large	$2.100 per Hour
High-I/O On-Demand Instances	
Quadruple Extra Large	$3.100 per Hour
High-Storage On-Demand Instances	
Eight Extra Large	$4.600 per Hour

1.7 GB memory
1 virtual core
(1 CU each)
160GB storage
'moderate' I/O

68.4 GB memory
8 virtual cores
(3.25 CU each)
1690 GB storage
'high' I/O

http://aws.amazon.com/ec2/#pricing (9/11/2013)

- Infrastructure-as-a-Service (IaaS)
  - You can rent various types of virtual machines by the hour
  - In your VMs, you can run your own (Linux/Windows) programs
    - Examples: Web server, search engine, movie renderer, ...

Oh no - where has my data gone?

- EC2 instances do not have persistent storage
  - Data survives stops & reboots, but not termination

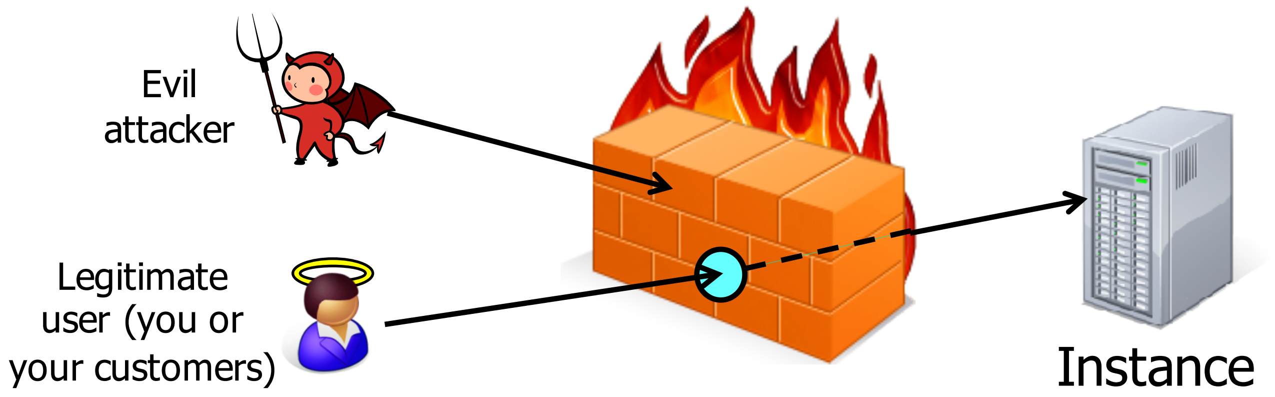
If you store data on the virtual hard disk of your instance
and the instance fails or you terminate it,
your data WILL be lost!

- So where should I put persistent data?
  - Elastic Block Store (EBS) - in a few slides
  - Ideally, use an AMI with an EBS root (Amzon's default AMI has this property)

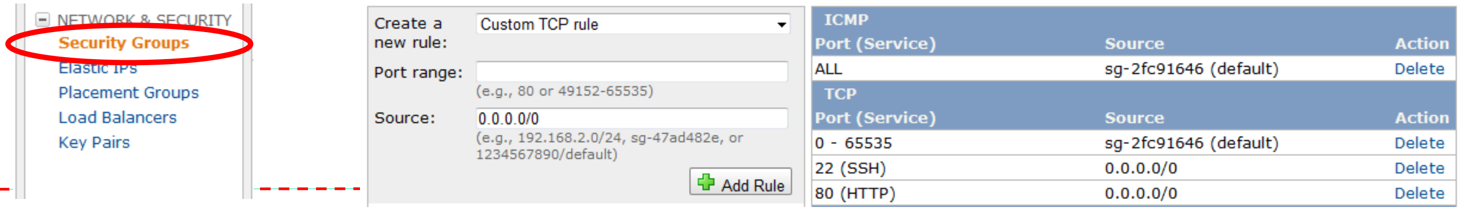
Amazon Machine Images

- When I launch an instance, what software will be installed on it?
  - Software is taken from an Amazon Machine Image (AMI)
  - Selected when you launch an instance
  - Essentially a file system that contains the operating system, applications, and potentially other data
  - Lives in S3
- How do I get an AMI?
  - Amazon provides several generic ones, e.g., Amazon Linux, Fedora Core, Windows Server, ...
  - You can make your own
    - You can even run your own custom kernel (with some restrictions)

Security Groups



- Basically, a set of firewall rules
  - Can be applied to groups of EC2 instances
  - Each rule specifies a protocol, port numbers, etc...
  - Only traffic matching one of the rules is allowed through
- Sometimes need to explicitly open ports



Create a new rule: Custom TCP rule

Port range:
(e.g., 80 or 49152-65535)

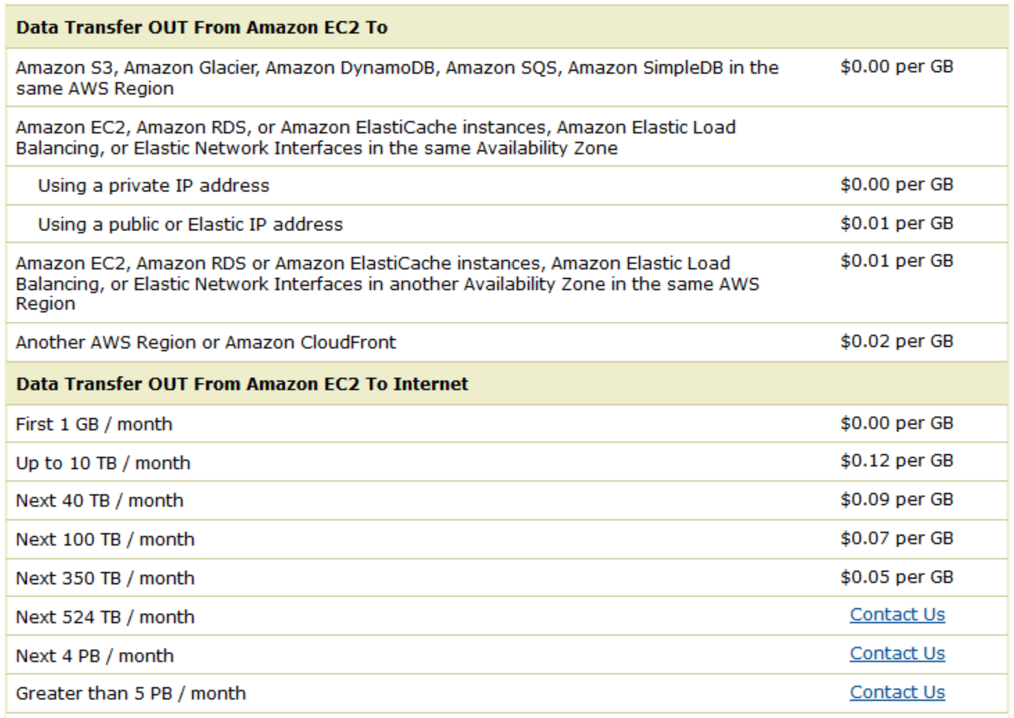
Source: 0.0.0.0/0
(e.g., 192.168.2.0/24, sg-47ad482e, or 1234567890/default)

ICMP		
Port (Service)	Source	Action
ALL	sg-2fc91646 (default)	Delete
TCP		
Port (Service)	Source	Action
0 - 65535	sg-2fc91646 (default)	Delete
22 (SSH)	0.0.0.0/0	Delete
80 (HTTP)	0.0.0.0/0	Delete

Regions and Availability Zones

- Where exactly does my instance run?
  - No easy way to find out - Amazon does not say
- Instances can be assigned to regions
  - Currently 9 availble: US East (Northern Virginia), US West (Northern California), US West (Oregon), EU (Ireland), Asia/Pacific (Singapore), Asia/Pacific (Sydney), Asia/Pacific (Tokyo), South America (Sao Paulo), AWS GovCloud
  - Important, e.g., for reducing latency to customers
- Instances can be assigned to availability zones
  - Purpose: Avoid correlated fault
  - Several availability zones within each region

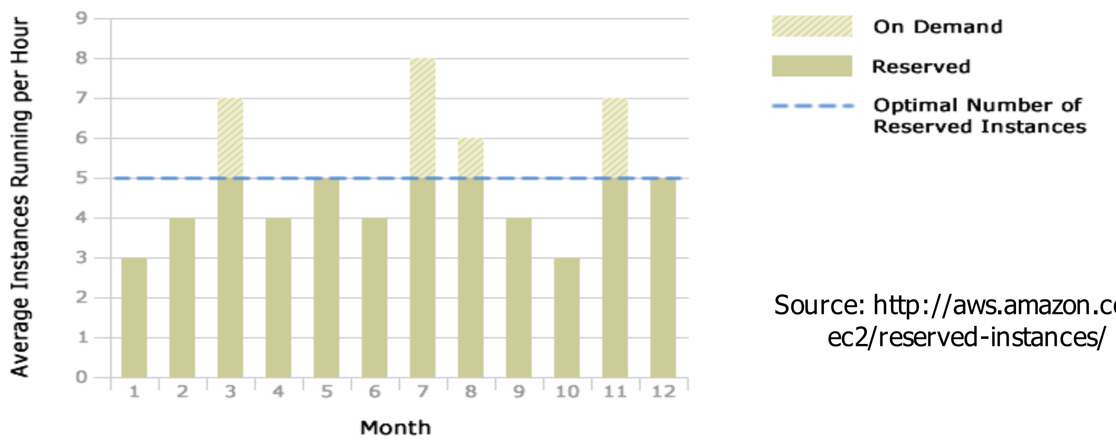
Network pricing



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
Next 524 TB / month	Contact Us
Next 4 PB / month	Contact Us
Greater than 5 PB / month	Contact Us

- AWS does charge for network traffic
  - Price depends on source and destination of traffic
  - Free within EC2 and other AWS svcs in same region (e.g., S3)
  - Remember: ISPs are typically charged for upstream traffic

Instance types

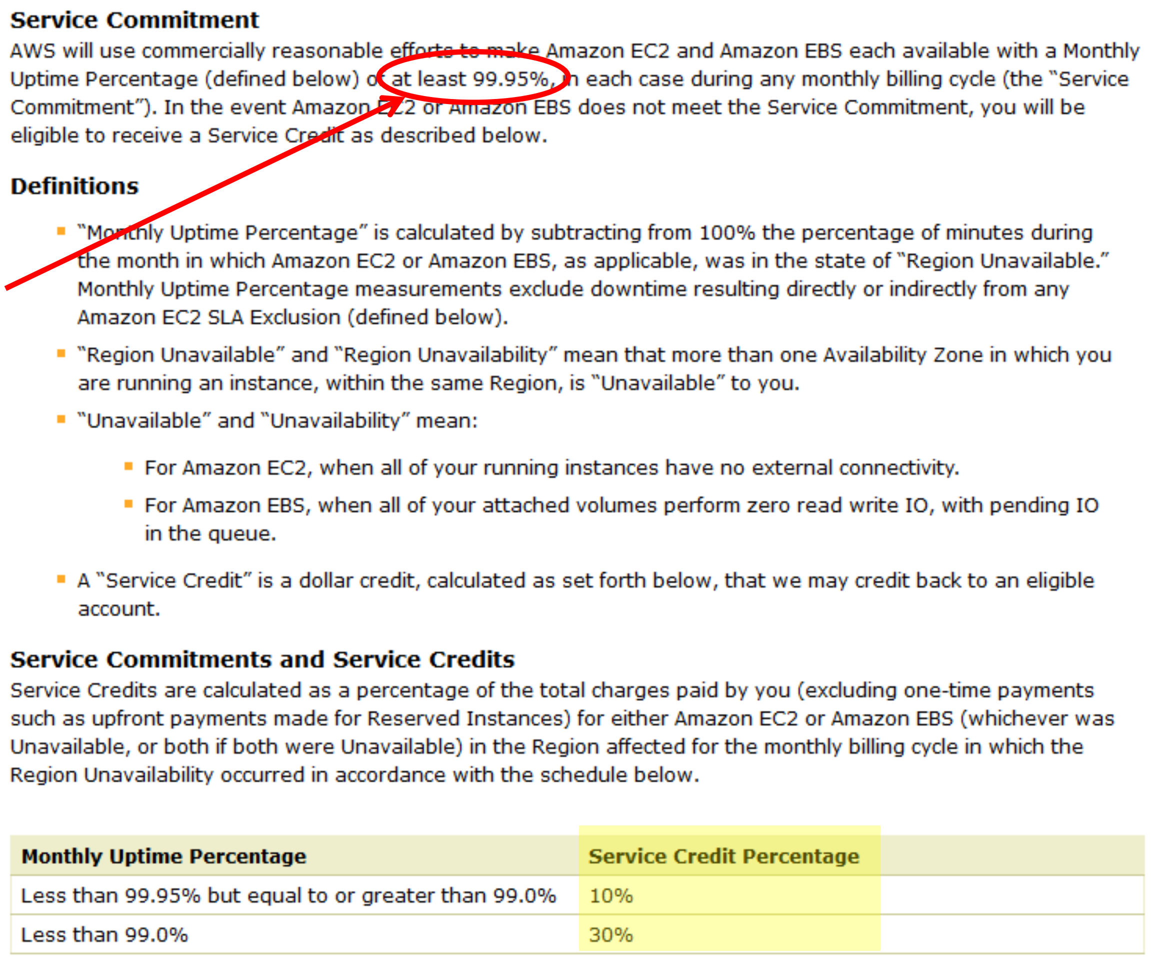


Source: http://aws.amazon.com/ec2/reserved-instances/

- So far: On-demand instances
- Also available: Reserved instances
  - One-time reservation fee to purchase for 1 or 3 years
  - Usage still billed by the hour, but at a considerable discount
- Also available: Spot instances
  - Spot market: Can bid for available capacity
  - Instance continues until terminated or price rises above bid

Service Level Agreement

4.38h downtime per year allowed



Service Commitment

AWS will use commercially reasonable efforts to make Amazon EC2 and Amazon EBS each available with a Monthly Uptime Percentage (defined below) of at least 99.95%, in each case during any monthly billing cycle (the "Service Commitment"). In the event Amazon EC2 or Amazon EBS does not meet the Service Commitment, you will be eligible to receive a Service Credit as described below.

Definitions

- "Monthly Uptime Percentage" is calculated by subtracting from 100% the percentage of minutes during the month in which Amazon EC2 or Amazon EBS, as applicable, was in the state of "Region Unavailable." Monthly Uptime Percentage measurements exclude downtime resulting directly or indirectly from any Amazon EC2 SLA Exclusion (defined below).
- "Region Unavailable" and "Region Unavailability" mean that more than one Availability Zone in which you are running an instance, within the same Region, is "Unavailable" to you.
- "Unavailable" and "Unavailability" mean:
  - For Amazon EC2, when all of your running instances have no external connectivity.
  - For Amazon EBS, when all of your attached volumes perform zero read write IO, with pending IO in the queue.
- A "Service Credit" is a dollar credit, calculated as set forth below, that we may credit back to an eligible account.

Service Commitments and Service Credits

Service Credits are calculated as a percentage of the total charges paid by you (excluding one-time payments such as upfront payments made for Reserved Instances) for either Amazon EC2 or Amazon EBS (whichever was Unavailable, or both if both were Unavailable) in the Region affected for the monthly billing cycle in which the Region Unavailability occurred in accordance with the schedule below.

Monthly Uptime Percentage	Service Credit Percentage
Less than 99.95% but equal to or greater than 99.0%	10%
Less than 99.0%	30%

http://aws.amazon.com/ec2-sla/ (9/11/2013; excerpt)

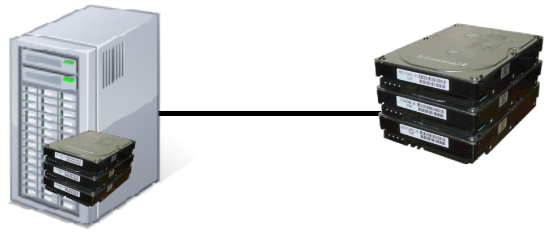
Recap: EC2

- What EC2 is:
  - IaaS service - you can rent virtual machines
  - Various types: Very small to very powerful
- How to use EC2:
  - Ephemeral state - local data is lost when instance terminates
  - AMIs - used to initialize an instance (OS, applications, ...)
  - Security groups - "firewalls" for your instances
  - Regions and availability zones
  - On-demand/reserved/spot instances
  - Service level agreement (SLA)

Plan for today

- A brief history of cloud computing ✓
- Introduce one specific commercial cloud
  - Amazon Web Services (AWS) ✓
  - Elastic Compute Cloud (EC2) ✓
  - Elastic Block Storage (EBS) NEXT
  - Other services: Mechanical Turk, CloudFront, ...
  - Next time: S3 and SimpleDB

What is Elastic Block Store (EBS)?



- Persistent storage
  - Unlike the local instance store, data stored in EBS is not lost when an instance fails or is terminated
- Should I use the instance store or EBS?
  - Typically, instance store is used for temporary data

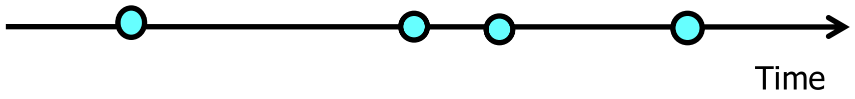
Volumes

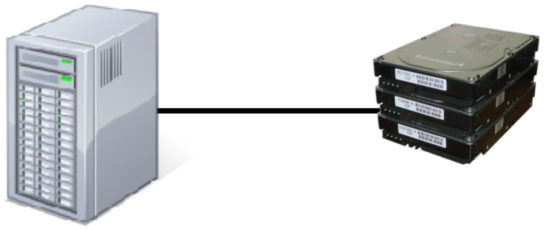
- EBS storage is allocated in volumes
  - A volume is a 'virtual disk' (size: 1GB - 1TB)
  - Basically, a raw block device
  - Can be attached to an instance (but only one at a time)
  - A single instance can access multiple volumes
- Placed in specific availability zones
  - Why is this useful?
  - Be sure to place it near instances (otherwise can't attach)
- Replicated across multiple servers
  - Data is not lost if a single server fails
  - Amazon: Annual failure rate is 0.1-0.5% for a 20GB volume

EC2 instances with EBS roots

- EC2 instances can have an EBS volume as their root device ("EBS boot")
  - Result: Instance data persists independently from the lifetime of the instance
  - You can stop and restart the instance, similar to suspending and resuming a laptop
    - You won't be charged for the instance while it is stopped (only for EBS)
  - You can enable termination protection for the instance
    - Blocks attempts to terminate the instance (e.g., by accident) until termination protection is disabled again
- Alternative: Use instance store as the root
  - You can still store temporary data on it, but it will disappear when you terminate the instance
  - You can still create and mount EBS volumes explicitly

Snapshots





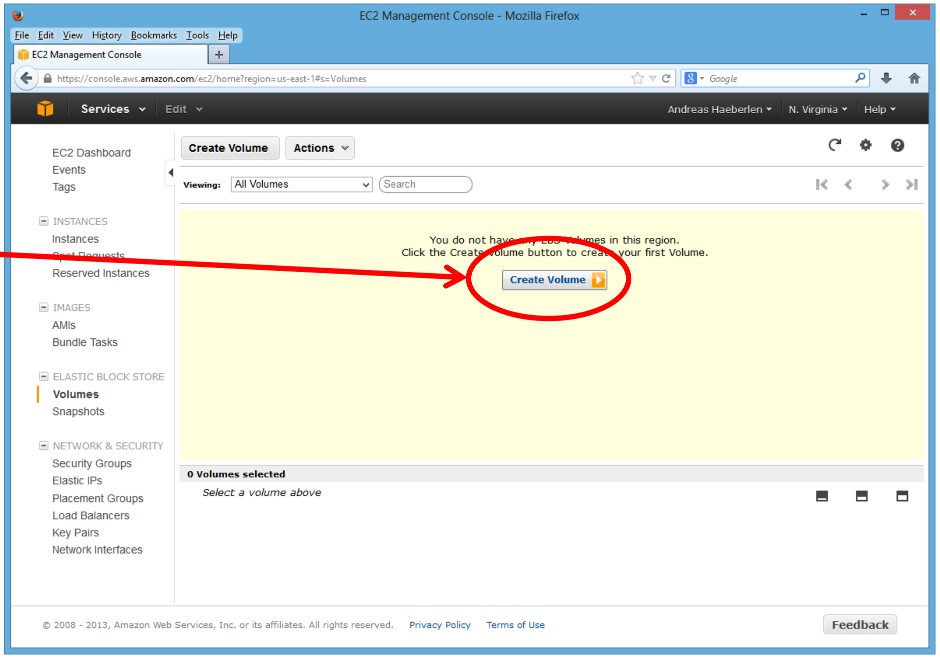
- You can create a snapshot of a volume
  - Copy of data in the volume at the time snapshot was made
  - Only the first snapshot makes a full copy; subsequent snapshots are incremental
- What are snapshots good for?
  - Sharing data with others
    - DBpedia snapshot ID is "snap-882a8ae3"
    - Access control list (specific account numbers) or public access
  - Instantiate new volumes
  - Point-in-time backups

Pricing

- You pay for...
  - Storage space: $0.10 per allocated GB per month
  - I/O requests: $0.10 per million I/O requests
  - S3 operations (GET/PUT)
- Charge is only for actual storage used
  - Empty space does not count

Creating an EBS volume

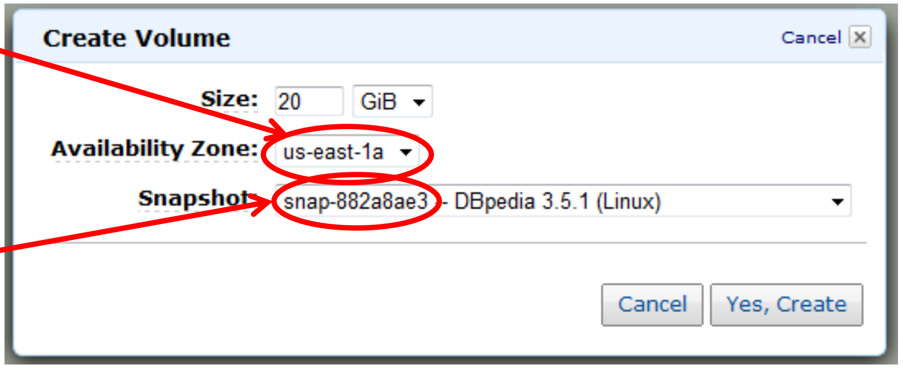
Create volume



You do not have any EBS Volumes in this region.
Click the Create Volume button to create your first Volume.

© 2008 - 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Needs to be in same availability zone as your instance!



Size: 20 GiB
Availability Zone: us-east-1a
Snapshot: snap-882a8ae3 - DBpedia 3.5.1 (Linux)

DBpedia snapshot ID

Mounting an EBS volume

- Step 1: Attach the volume

```
mkse212@vm:~$ ec2-attach-volume -d /dev/sda2 -i i-9bd6eef1 vol-cca68ea5
ATTACHMENT      vol-cca68ea5    i-9bd6eef1      /dev/sda2       attaching
mkse212@vm:~$
```

- Step 2: Mount the volume in the instance

```
mkse212@vm:~$ ssh ec2-user@ec2-50-17-64-130.compute-1.amazonaws.com

       __|  __|_  )  Amazon Linux AMI
       _|  (     /     Beta
      ___|\___|___|

See /usr/share/doc/system-release-2011.02 for latest release notes. :-)
[ec2-user@ip-10-196-82-65 ~]$ sudo mount /dev/sda2 /mnt/
[ec2-user@ip-10-196-82-65 ~]$ ls /mnt/
dbpedia_3.5.1.owl  dbpedia_3.5.1.owl.bz2  en  other_languages
[ec2-user@ip-10-196-82-65 ~]$
```

Detaching an EBS volume

- Step 1: Unmount the volume in the instance

```
[ec2-user@ip-10-196-82-65 ~]$ sudo umount /mnt/
[ec2-user@ip-10-196-82-65 ~]$ exit
mkse212@vm:~$
```

- Step 2: Detach the volume

```
mkse212@vm:~$ ec2-detach-volume vol-cca68ea5
ATTACHMENT      vol-cca68ea5    i-9bd6eef1      /dev/sda2       detaching
mkse212@vm:~$
```

Recap: Elastic Block Store (EBS)

- What EBS is:
  - Basically a virtual hard disk; can be attached to EC2 instances
  - Persistent - state survives termination of EC2 instance
- How to use EBS:
  - Allocate volume - empty or initialized with a snapshot
  - Attach it to EC2 instance and mount it there
  - Can create snapshots for data sharing, backup

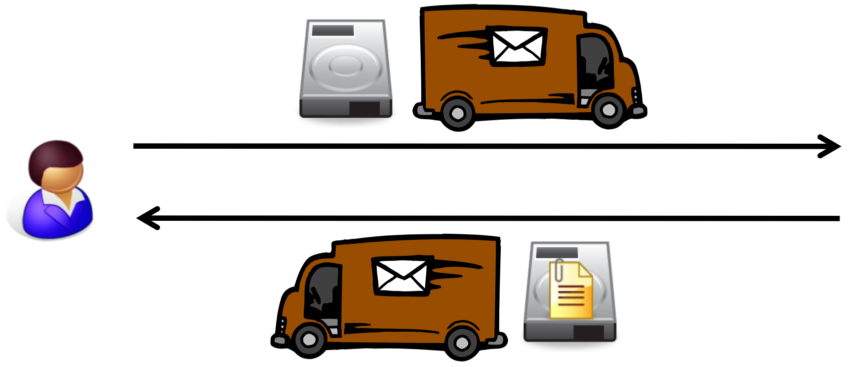
Plan for today

- A brief history of cloud computing ✓
- Introduce one specific commercial cloud ✓
  - Amazon Web Services (AWS) ✓
  - Elastic Compute Cloud (EC2) ✓
  - Elastic Block Storage (EBS) ✓
  - Other services: Mechanical Turk, CloudFront, … ← NEXT

AWS Import/Export

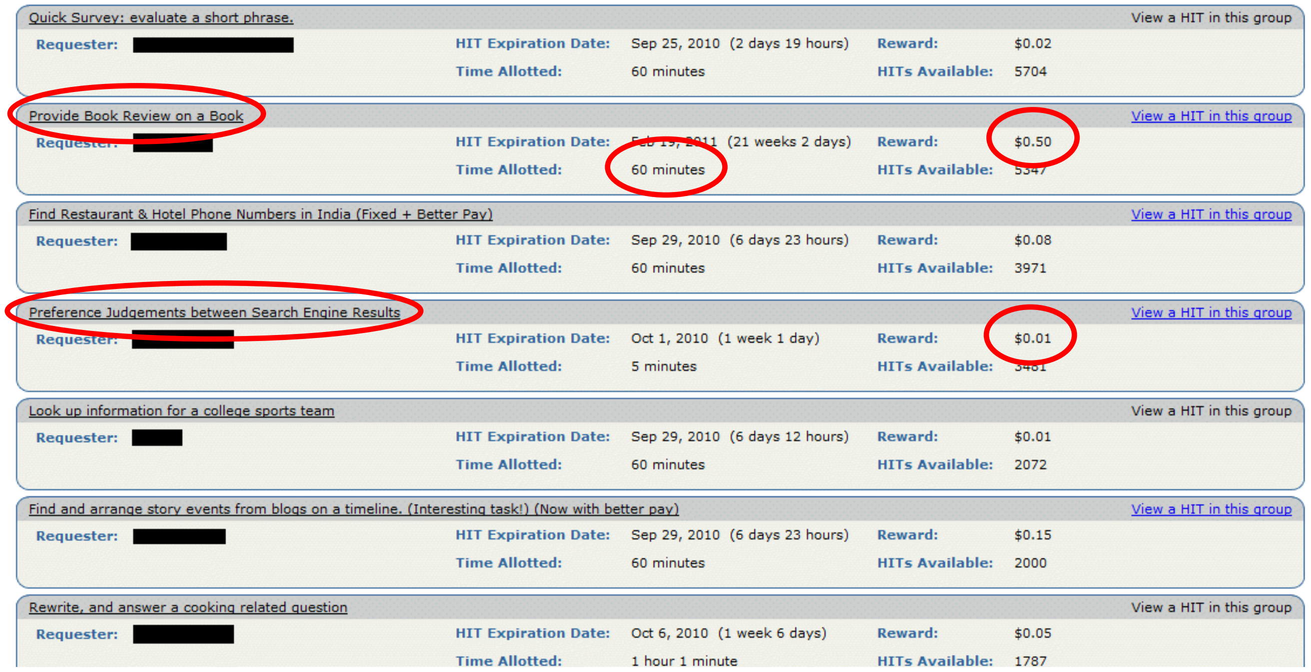
Method	Time
Internet (20Mbps)	45 days
FedEx	1 day

Time to transfer 10TB [AF10]



- Import/export large amounts of data to/from S3 buckets via physical storage device
  - Mail an actual hard disk to Amazon (power adapter, cables!)
  - Signature file for authentication
  - Discussion: Is this the Right Way to be shipping data, or should we rather be using a network?

Mechanical Turk (MTurk)

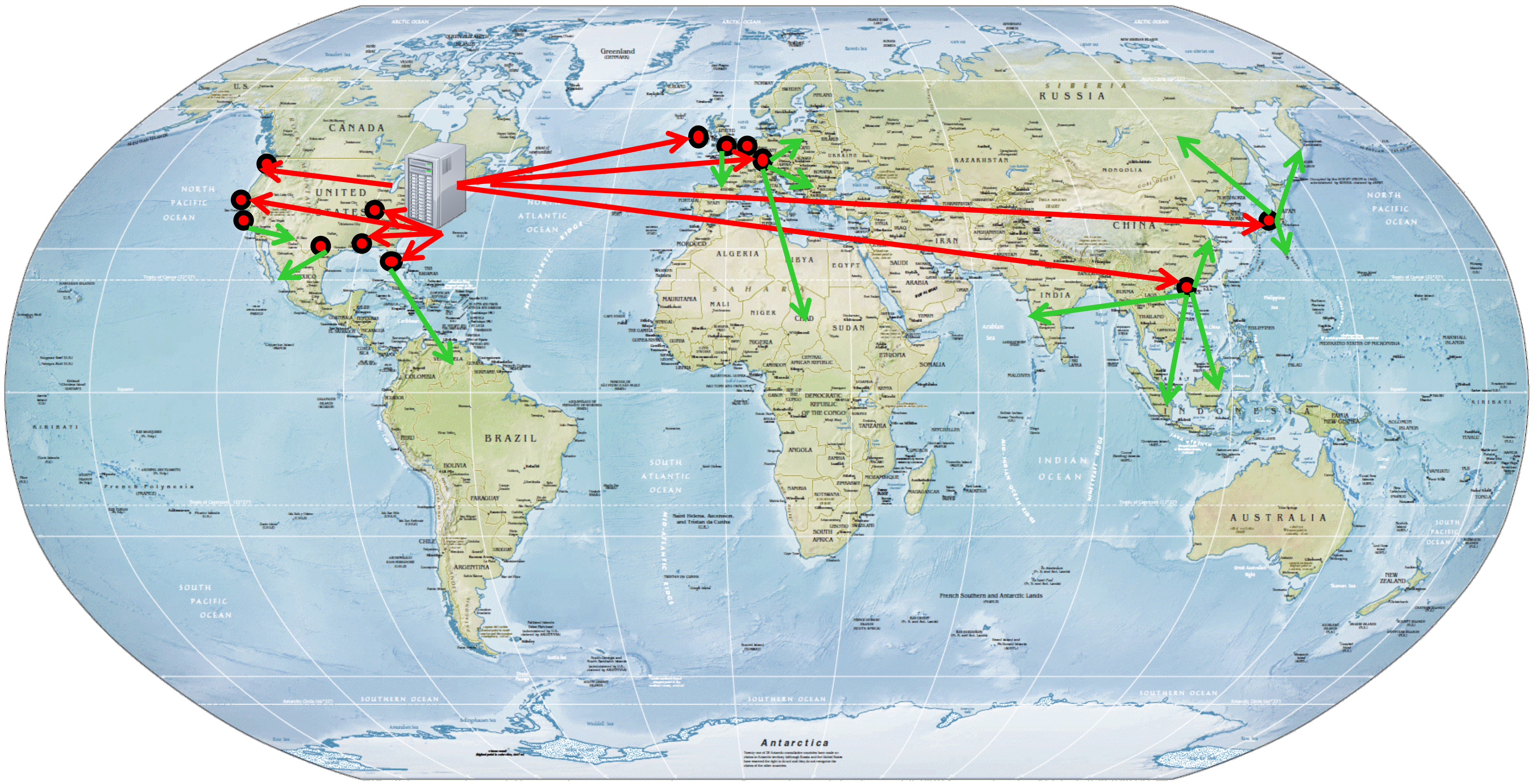


HIT	Requester:	HIT Expiration Date:	Time Allotted:	Reward:	HITs Available:	
Quick Survey: evaluate a short phrase.	[redacted]	Sep 25, 2010 (2 days 19 hours)	60 minutes	$0.02	5704	View a HIT in this group
Provide Book Review on a Book	[redacted]	Feb 19, 2011 (21 weeks 2 days)	60 minutes	$0.50	5347	View a HIT in this group
Find Restaurant & Hotel Phone Numbers in India (Fixed + Better Pay)	[redacted]	Sep 29, 2010 (6 days 23 hours)	60 minutes	$0.08	3971	View a HIT in this group
Preference Judgements between Search Engine Results	[redacted]	Oct 1, 2010 (1 week 1 day)	5 minutes	$0.01	3481	View a HIT in this group
Look up information for a college sports team	[redacted]	Sep 29, 2010 (6 days 12 hours)	60 minutes	$0.01	2072	View a HIT in this group
Find and arrange story events from blogs on a timeline. (Interesting task!) (Now with better pay)	[redacted]	Sep 29, 2010 (6 days 23 hours)	60 minutes	$0.15	2000	View a HIT in this group
Rewrite, and answer a cooking related question	[redacted]	Oct 6, 2010 (1 week 6 days)	1 hour 1 minute	$0.05	1787	View a HIT in this group

https://www.mturk.com/mturk/ (9/23/2010 1:58am)

- A crowdsourcing marketplace
  - Requesters post small jobs (HIT - Human Intelligence Task), offer small rewards ($0.01-$0.10)

CloudFront



- Content distribution network
  - Caches S3 content at edge locations for low-latency delivery
  - Some similarities to other CDNs like Akamai, Limelight, ...

Case studies

Plan for today

- Recap: The cloud NEXT
  - Types of clouds, key benefits of cloud services
  - Major cloud providers
- SaaS case study: Salesforce.com
- PaaS case study: Facebook
- IaaS case study: Netflix
- Discussion

Recap: Public vs. Private Clouds

- As discussed previously, “cloud” is a broad term but comprises:
  - Very large data centers with thousands of commodity machines
  - Multiple, geographically distributed sites
  - Common management infrastructure
  - Common programming infrastructure that automatically allocates requests and/or jobs toavailable machines
- Difference between public and private clouds?
  - Public clouds sub-contract out to multiple clients; private clouds are controlled by one organization

Recap: Who uses the Cloud?

- Virtually all the major Web players can be considered to use Cloud capabilities
  - "Private" clouds: Amazon, eBay, Bing, Google, Salesforce, Facebook, …
  - "Public" clouds: Netflix, Jungle Disk, many companies' internal infrastructure
  - We'll discuss some examples today

Recap: Why use the Cloud?

- Main reason: cost savings due to elasticity
  - Commodity machines – easy to add, replace, expand
  - On-demand resources – pay as you need them, where you need them
- Especially true of public clouds
- But partially true of private clouds, where infrastructure might be shared among multiple divisions, tasks, etc.
- Also in some cases: geographic distribution

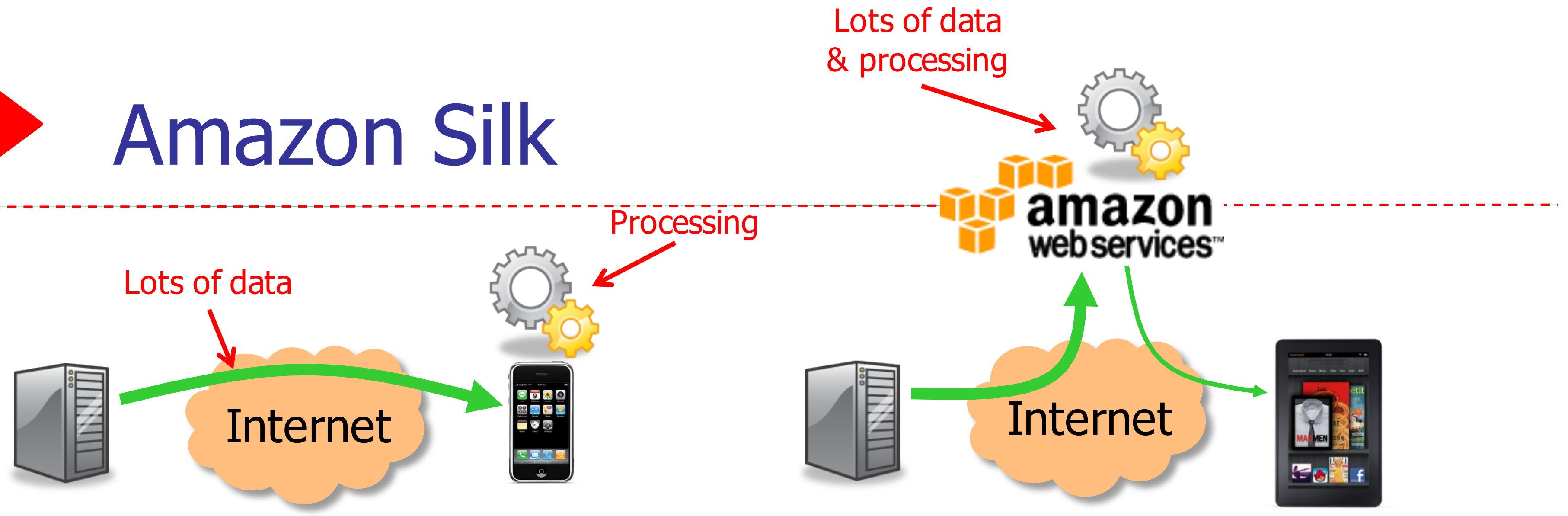
Recap: Types of clouds

- Software as a Service (SaaS): cloud-hosted apps
  - think Hotmail, GMail, Google Docs, Office Web, …
  - where Microsoft, etc. want to go – subscriptions & ads
- Platform as a Service (PaaS): programming layer and services over the cloud
  - think Hadoop, MS Azure, extensible apps, Google Maps
- Infrastructure as a Service (IaaS): virtual machines, virtualized networks and disks
  - think Amazon EC2
  - includes Storage as a Service: Amazon S3, SimpleDB, etc.
  - also some variants like content delivery networks

The major public Cloud providers

- Amazon is the big player
  - Multiple services: infrastructure as a service, platform as a service (incl. Hadoop), storage as a service
- But there are many others:
  - Microsoft Azure – in many ways has similar services to Amazon, with an emphasis on .Net programming model
  - Google App Engine + GWT + services – offers servlet-level programming interface, Hadoop, …
    - Also software as a service: GMail, Docs, etc.
  - IBM, HP, Yahoo – seem to focus mostly on enterprise (often private) cloud apps (not small business-level)
  - Rackspace, Terremark – mostly infrastructure as a service

Amazon Silk



- Idea: Use the cloud to make browsers faster
  - Page rendering is split between the user's device & the cloud
  - Cloud performs 'heavy lifting' (rendering, script execution, ...)
  - Device just has to show the resulting page, so it doesn't need much bandwidth or processing power (compare: Opera Mini)
- Many opportunities for optimizations
  - Smart caching, on-the-fly optimizations
  - Learn about traffic patterns and pre-fetch pages

Plan for today

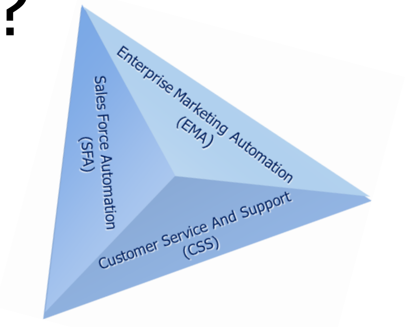
- Recap: The cloud ✓
  - Types of clouds, key benefits of cloud services ✓
  - Major cloud providers ✓
- SaaS case study: Salesforce.com ⇦ NEXT
- PaaS case study: Facebook
- IaaS case study: Netflix
- Discussion

Software as a Service

- We'll look at three successful SaaS services hosted on companies' private clouds, all of which use AJAX-based Web interfaces:
  - Salesforce.com (also similar: NetSuite; Quicken's Web apps; TurboTax Web; etc.)
  - GMail (also similar: Hotmail, Yahoo Mail)
  - Google Docs (also similar: Office Web)
- In some sense, your HWs and projects are along this vein!

Salesforce.com

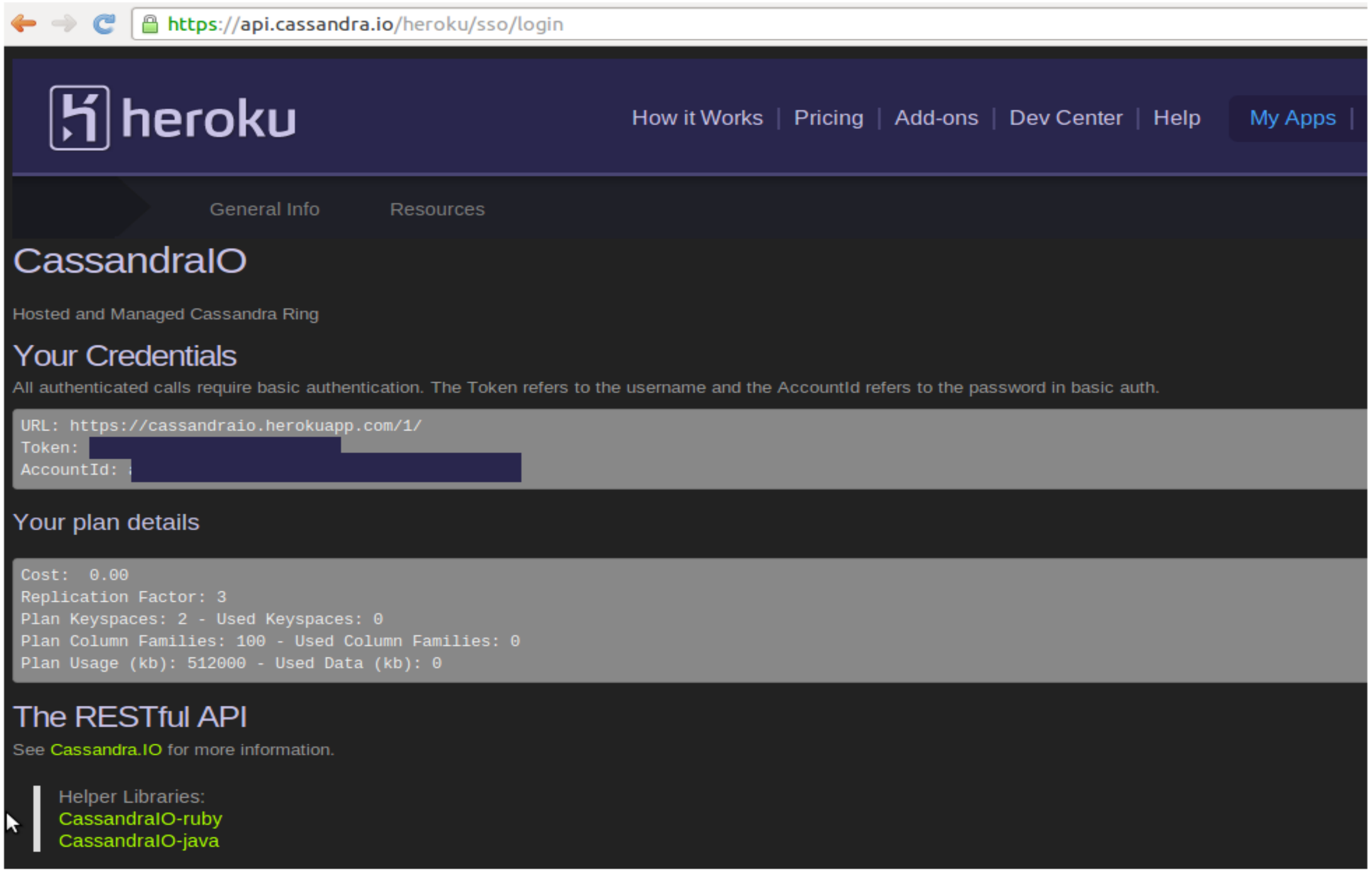
- Perhaps the first truly successful "software as a service" platform
  - Predated the term "cloud" (founded in 1999) – and was initially met with skepticism
  - Now the IBMs, MSs of the world want to be like them: a constant revenue stream, unlike shrink-wrapped software
- What is the software being provided?
  - "Customer Relationship Management" – tools for sales people to find customers, keep in contact with them
  - Gives a bird's-eye view of customers' status, in-flight orders, order history, leads, approvals, etc.


Enterprise Marketing Automation (EMA)
Sales Force Automation (SFA)
Customer Service And Support (CSS)

Salesforce.com: A Timeline

- Founded in 1999: first proponents of the term 'cloud', with support from Larry Ellison (Oracle)
- First CRM offered as a SAAS (Software as a service)
- 2005: offered Force.com as a platform for apps
- 2010: Chatter Launched, Heroku acquired
- 2011: Radian 6 acquired, more than 90,000 customers

What does it look like?


https://api.cassandra.io/heroku/sso/login
heroku
How it Works | Pricing | Add-ons | Dev Center | Help | My Apps
General Info
Resources

CassandraIO

Hosted and Managed Cassandra Ring

Your Credentials

All authenticated calls require basic authentication. The Token refers to the username and the AccountId refers to the password in basic auth.

```
URL: https://cassandraio.herokuapp.com/1/
Token: [redacted]
AccountId: [redacted]
```

Your plan details

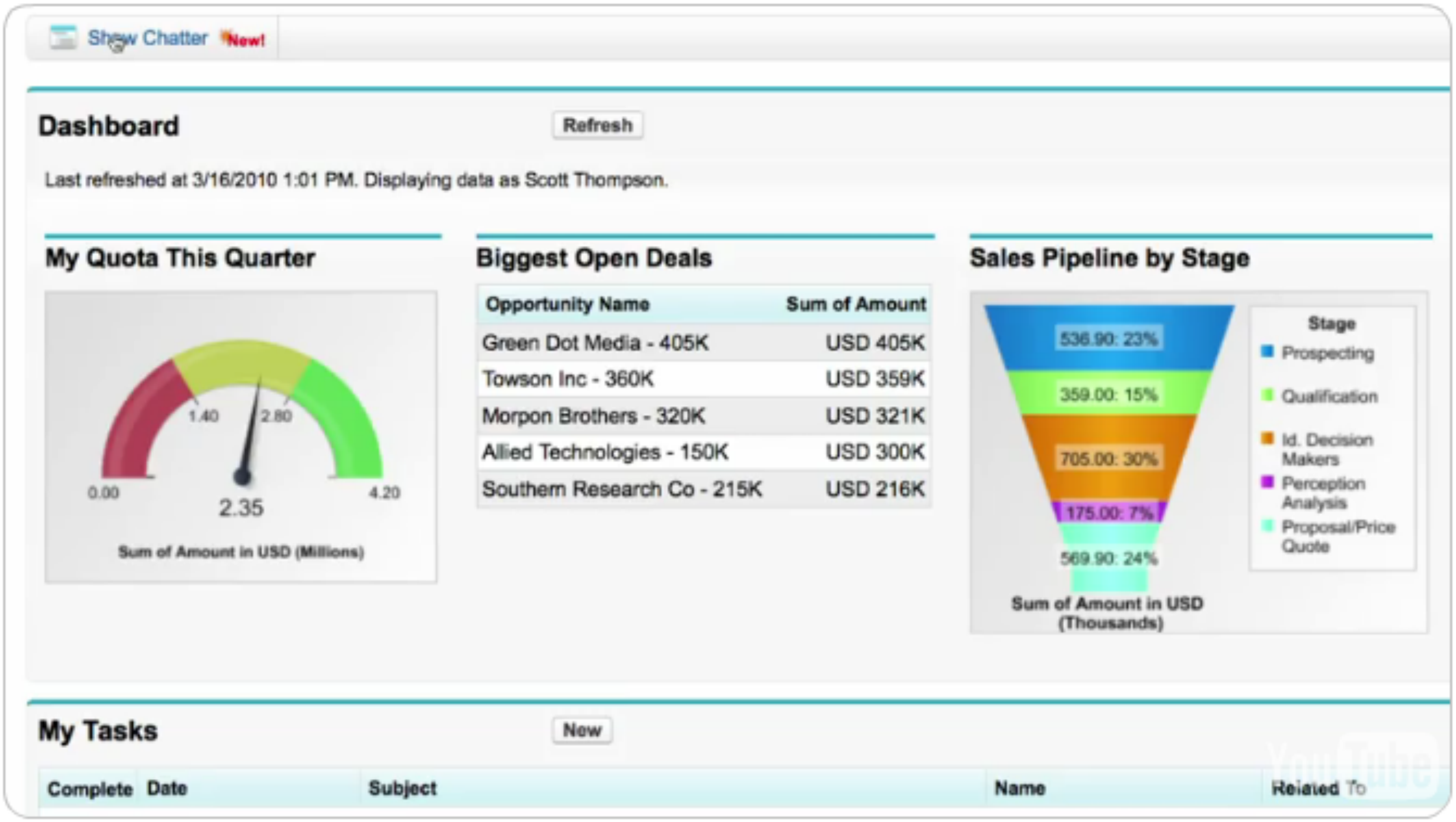
```
Cost:  0.00
Replication Factor: 3
Plan Keyspaces: 2 - Used Keyspaces: 0
Plan Column Families: 100 - Used Column Families: 0
Plan Usage (kb): 512000 - Used Data (kb): 0
```

The RESTful API

See Cassandra.IO for more information.

> Helper Libraries:
> CassandraIO-ruby
> CassandraIO-java

Example Salesforce “Dashboard”



Dashboard

Last refreshed at 3/16/2010 1:01 PM. Displaying data as Scott Thompson.

Biggest Open Deals

Opportunity Name	Sum of Amount
Green Dot Media - 405K	USD 405K
Towson Inc - 360K	USD 359K
Morpon Brothers - 320K	USD 321K
Allied Technologies - 150K	USD 300K
Southern Research Co - 215K	USD 216K

Stage
🟦 Prospecting
🟩 Qualification
🟧 Id. Decision Makers
🟪 Perception Analysis
🟩 Proposal/Price Quote

My Tasks

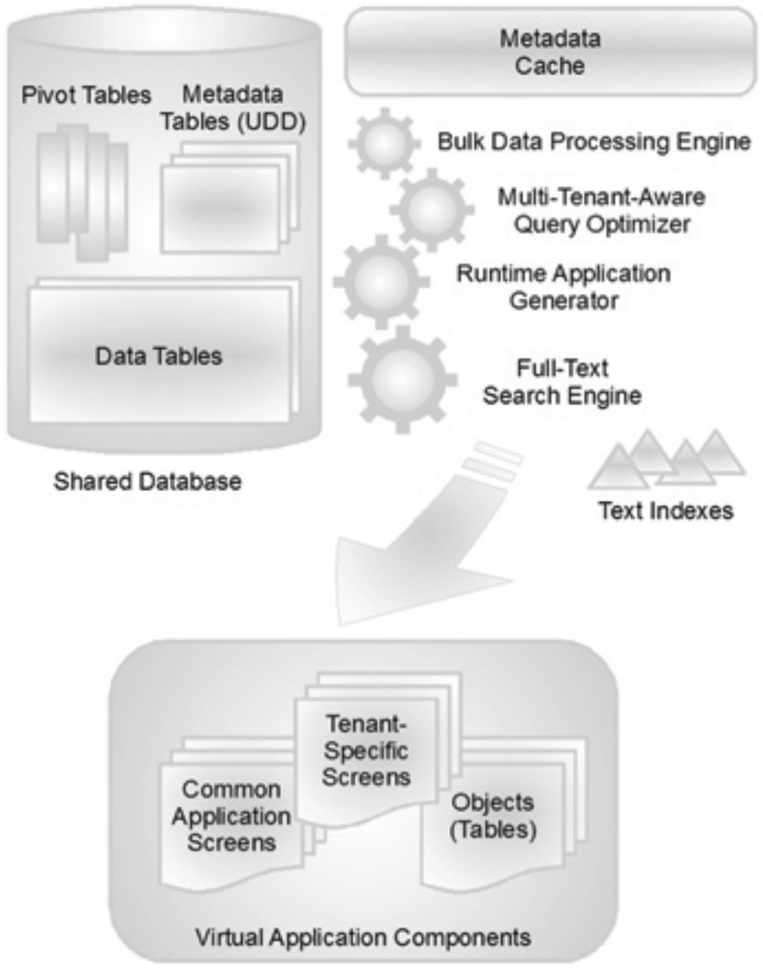
Complete	Date	Subject	Name	Related To

How Salesforce.com works

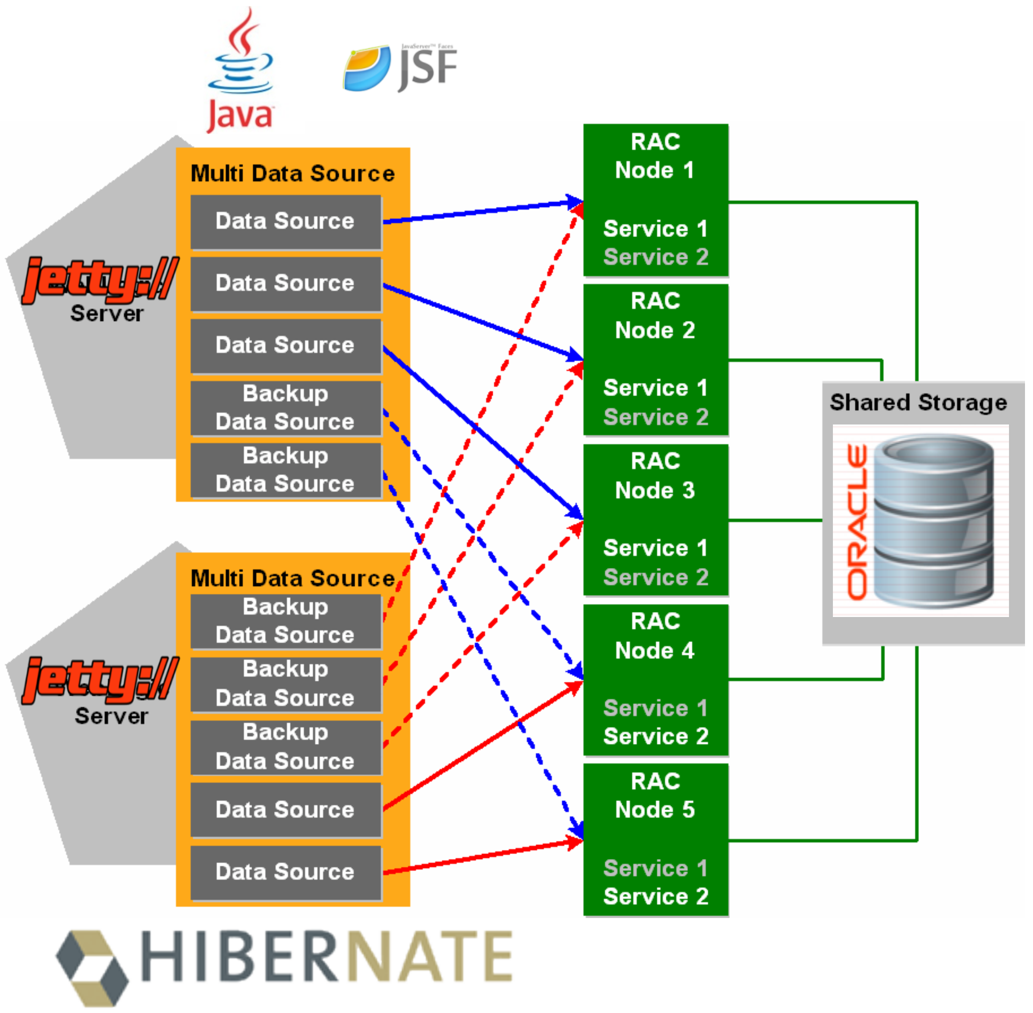
- Basic architecture as of Mar 2009:
  - 'Only' about 1000 mirrored machines for
    - 55K enterprise customers, 1.5M subscribers
  - 10 Oracle databases across 50 servers
    - About 20 predefined tables / schemas, shared across all customers, 100s of TB
    - Sophisticated, proprietary query optimization and indexing
  - AJAX Web interface with various communication services
    - Tracking for Twitter, collaborative tools, etc.
  - Easy "tunnels" for sharing across customers
- Plug-ins for extensions via Platform-as-a-Service "force.com" – 30M lines of 3rd party code

Salesforce.com Architecture

- Multi-tenant: Each datacenter contains servers shared across customers
- Performance maintained by limits
- App logic separation
- Scales vertically (adding more cores, improving index strategies)



Salesforce.com Technology Stack



- Consist of Oracle RAC (Real Application Clusters) nodes
- Allow transparent access of single database instance by multiple clients
- Largest standing Oracle installation in the world

Why Salesforce is so effective

- Their value proposition: outsource your main corporate IT to them
  - They bill per month – force.com $15/user/month
- They can offer it cheaper than corporate IT:
  - Leverage the same infrastructure, design, and support across many companies at the same time – “multi-tenancy”
- Some customers:
  - Dell, AMD, SunTrust, Spring, Computer Associates, Kaiser Permanente

Outsourcing your e-mail: Gmail

- (and, to a lesser extent, Yahoo Mail, Hotmail)
- Basic architecture:
  - Distributed, replicated message store in BigTable – a key-value store like Amazon SimpleDB
    - "Multihomed" model – if one site crashes, user gets forwarded to another
    - Weak consistency model for some operations – "message read"
    - Stronger consistency for others – "send message"
- We all know Gmail: what is it that makes it special?
- What is the business model?

Outsourcing your documents: Google Docs

- The idea:
  - instead of buying software, worrying about security and administration…
  - simply put your docs on the Web and let Google do the rest!
- Today: much remains to be proven
  - Features? [right now, very limited vs. MS Office]
  - Security? [cf. hackers' attack on Google]
- But some benefits
  - Sharing and collaboration are much easier

Plan for today

- Recap: The cloud ✓
  - Types of clouds, key benefits of cloud services ✓
  - Major cloud providers ✓
- SaaS case study: Salesforce.com ✓
- PaaS case study: Facebook ← NEXT
- IaaS case study: Netflix
- Discussion

Users of Platform as a Service

- Facebook provides some PaaS capabilities to application developers
  - Web services – remote APIs – that allow access to social network properties, data, "Like" button, etc.
  - Many third-parties run their apps off Amazon EC2, and interface to Facebook via its APIs – PaaS + IaaS
- Facebook itself makes heavy use of PaaS services for their own private cloud
  - Key problems: how to analyze logs, make suggestions, determine which ads to place

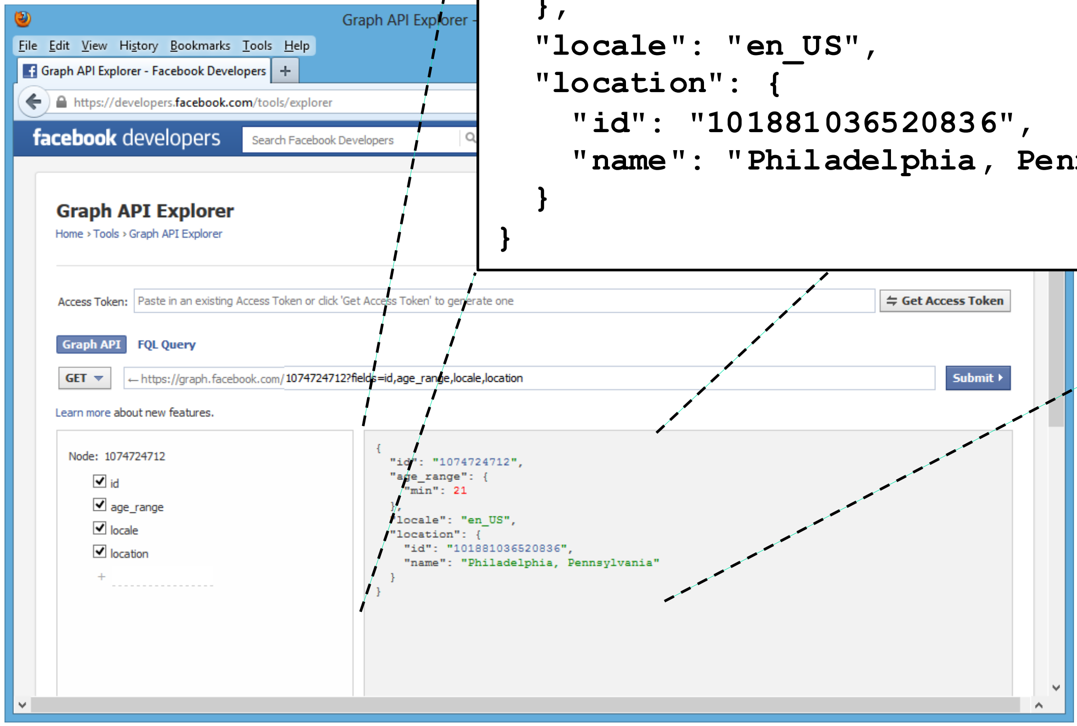
Facebook API: Overview

- What you can do:
  - Read data from profiles and pages
  - Navigate the graph (e.g., via friends lists)
  - Issue queries (for posts, people, pages, ...)
  - Add or modify data (e.g., create new posts)
  - Get real-time updates, issue batch requests, ...

- How you can access it:
  - Graph API
  - FQL
  - Legacy REST API

Facebook API: The

```
{
  "id": "1074724712",
  "age_range": {
    "min": 21
  },
  "locale": "en_US",
  "location": {
    "id": "101881036520836",
    "name": "Philadelphia, Pennsylvania"
  }
}
```


Graph API Explorer
File Edit View History Bookmarks Tools Help
Graph API Explorer - Facebook Developers
https://developers.facebook.com/tools/explorer
facebook developers
Search Facebook Developers

Graph API Explorer

Home › Tools › Graph API Explorer

Access Token: Paste in an existing Access Token or click 'Get Access Token' to generate one

Get Access Token

Graph API FQL Query

GET https://graph.facebook.com/1074724712?fields=id,age_range,locale,location Submit

Learn more about new features.

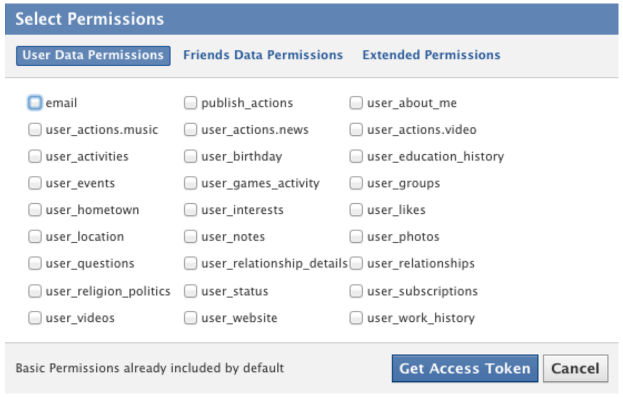
Node: 1074724712
- ☑ id
- ☑ age_range
- ☑ locale
- ☑ location

```
{
  "id": "1074724712",
  "age_range": {
    "min": 21
  },
  "locale": "en_US",
  "location": {
    "id": "101881036520836",
    "name": "Philadelphia, Pennsylvania"
  }
}
```

- Requests are mapped directly to HTTP:
  - https://graph.facebook.com/(identifier)?fields=(fieldList)
- Response is in JSON

Facebook API: The Graph API (2/2)

- Uses several HTTP methods:
  - GET for reading
  - POST for adding or modifying
  - DELETE for removing
- IDs can be numeric or names
  - /1074724712 or /andreas.haeberlen
  - Pages also have IDs



Select Permissions

User Data Permissions | Friends Data Permissions | Extended Permissions

☐ email	☐ publish_actions	☐ user_about_me
☐ user_actions.music	☐ user_actions.news	☐ user_actions.video
☐ user_activities	☐ user_birthday	☐ user_education_history
☐ user_events	☐ user_games_activity	☐ user_groups
☐ user_hometown	☐ user_interests	☐ user_likes
☐ user_location	☐ user_notes	☐ user_photos
☐ user_questions	☐ user_relationship_details	☐ user_relationships
☐ user_religion_politics	☐ user_status	☐ user_subscriptions
☐ user_videos	☐ user_website	☐ user_work_history

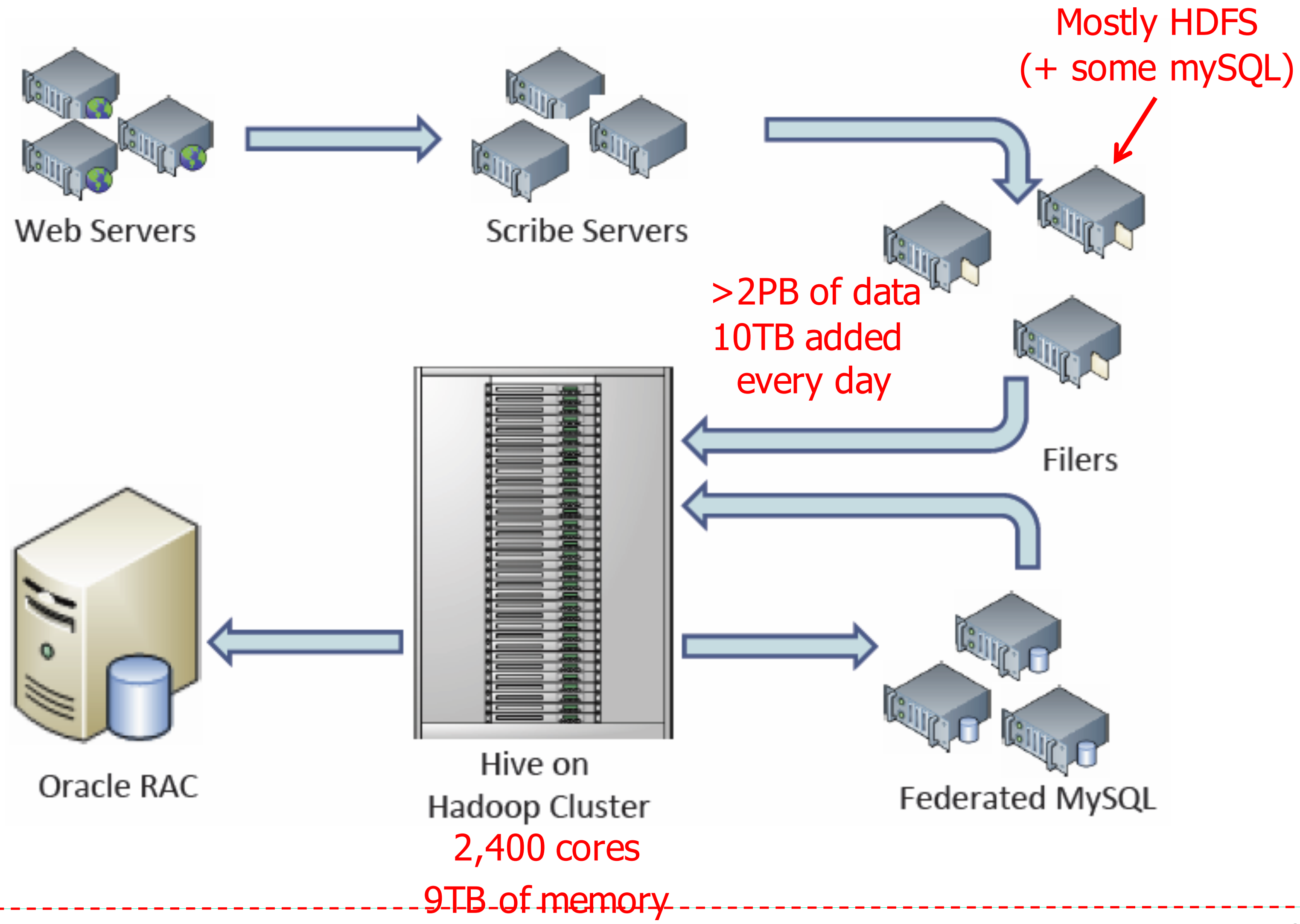
Basic Permissions already included by default

- Authorization is via 'access tokens'
  - Opaque string; encodes specific permissions (access user location, but not interests, etc.)
  - Has an expiration date, so may need to be refreshed

Facebook Data Management / Warehousing Tasks

- Main tasks for “cloud” infrastructure:
  - Summarization (daily, hourly)
    - to help guide development on different components
    - to report on ad performance
    - recommendations
  - Ad hoc analysis:
    - Answer questions on historical data – to help with managerial decisions
  - Archival of logs
  - Spam detection
  - Ad optimization
  - ...
- Initially used Oracle DBMS for this
  - But eventually hit scalability, cost, performance bottlenecks
  - ... just like Salesforce does now

Data Warehousing at Facebook



http://sites.ieee.org/scv-cs/files/2011/03/Facebook-Hive-by-Ashish-Thusoo.pdf

PaaS at Facebook

- Scribe – open source logging, actually records the data that will be analyzed by Hadoop
- Hadoop (MapReduce – discussed next time) as batch processing engine for data analysis
  - As of 2009: 2nd largest Hadoop cluster in the world, 2400 cores, > 2PB data with > 10TB added every day
- Hive – SQL over Hadoop, used to write the data analysis queries
- Federated MySQL, Oracle – multi-machine DBMSs to store query results

Example Use Case 1: Ad Details

- Advertisers need to see how their ads are performing
  - Cost-per-click (CPC), cost-per-1000-impressions (CPM)
  - Social ads – include info from friends
  - Engagement ads – interactive with video
- Performance numbers given:
  - Number unique users, clicks, video views, …
- Main axes:
  - Account, campaign, ad
  - Time period
  - Type of interaction
  - Users
- Summaries are computed using Hadoop via Hive

Use Case 2: Ad Hoc analysis, feedback

- Engineers, product managers may need to understand what is going on
  - e.g., impact of a new change on some sub-population
- Again, Hive-based, i.e., queries are in SQL with database joins
  - Combine data from several tables, e.g., click-through rate = views combined with clicks
- Sometimes requires custom analysis code with sampling

Plan for today

- Recap: The cloud ✓
  - Types of clouds, key benefits of cloud services ✓
  - Major cloud providers ✓
- SaaS case study: Salesforce.com ✓
- PaaS case study: Facebook ✓
- IaaS case study: Netflix NEXT
- Discussion

IaaS example: Netflix

- Perhaps Amazon’s highest-profile customer
  - In 12/2010, most of their traffic was served from AWS
  - A year earlier, none of it was
- Why did Netflix take this step?
  - Needed to re-architect after a phase of growth
    → Ability to question everything
  - Focus on their core competence (content); leave the 'heavy lifting' (datacenter operation) to Amazon
  - Customer growth & device engagement hard to predict
    → With the cloud, they don't have to
  - Belief that cloud computing is the future
    → Gain experience with an increasingly important technology

How Netflix uses AWS

- Streaming movie retrieval and playback
  - Media files stored in S3
  - “Transcoding” to target devices (Wii, iPad, etc.) using EC2
- Web site modules
  - Movie lists and search – app hosted by Amazon Web Services
- Recommendations
  - Analysis of streaming sessions, business metrics – using Elastic MapReduce

Netflix: 5 Lessons learned using AWS

- Dorothy, you're not in Kansas anymore
  - Be prepared to unlearn a lot of what you know
  - Example: Assumptions about network capacity, hw reliability
- Co-tenancy is hard
  - Throughput variance can occur at any level in the stack
- Best way to avoid failure: Fail constantly
  - Design for failure independence; use the 'Chaos Monkey'
- Learn with real scale, not toy models
  - Only full-scale traffic shows where the real bottlenecks are
- Commit yourself

Plan for today

- Recap: The cloud ✓
  - Types of clouds, key benefits of cloud services ✓
  - Major cloud providers ✓
- SaaS case study: Salesforce.com ✓
- PaaS case study: Facebook ✓
- IaaS case study: Netflix ✓
- Discussion NEXT

Other users, and the future

- Startups, especially, are making great use of EC2, Rackspace, etc. for their hosting needs
  - compare to 10 years ago – dot-com boom – where you started by buying a cluster of SPARC machines
- Government, health care, science, many enterprises have great interest in cost savings of the cloud
  - But concerns remain – esp. with respect to security, privacy, availability
- … And moreover: the last word has not been written on how to *program* the cloud

Slides adapted (under permission) from

Andreas Haeberlen

(NETS 212: Scalable and Cloud Computing)

CIS Department, Penn-State University