

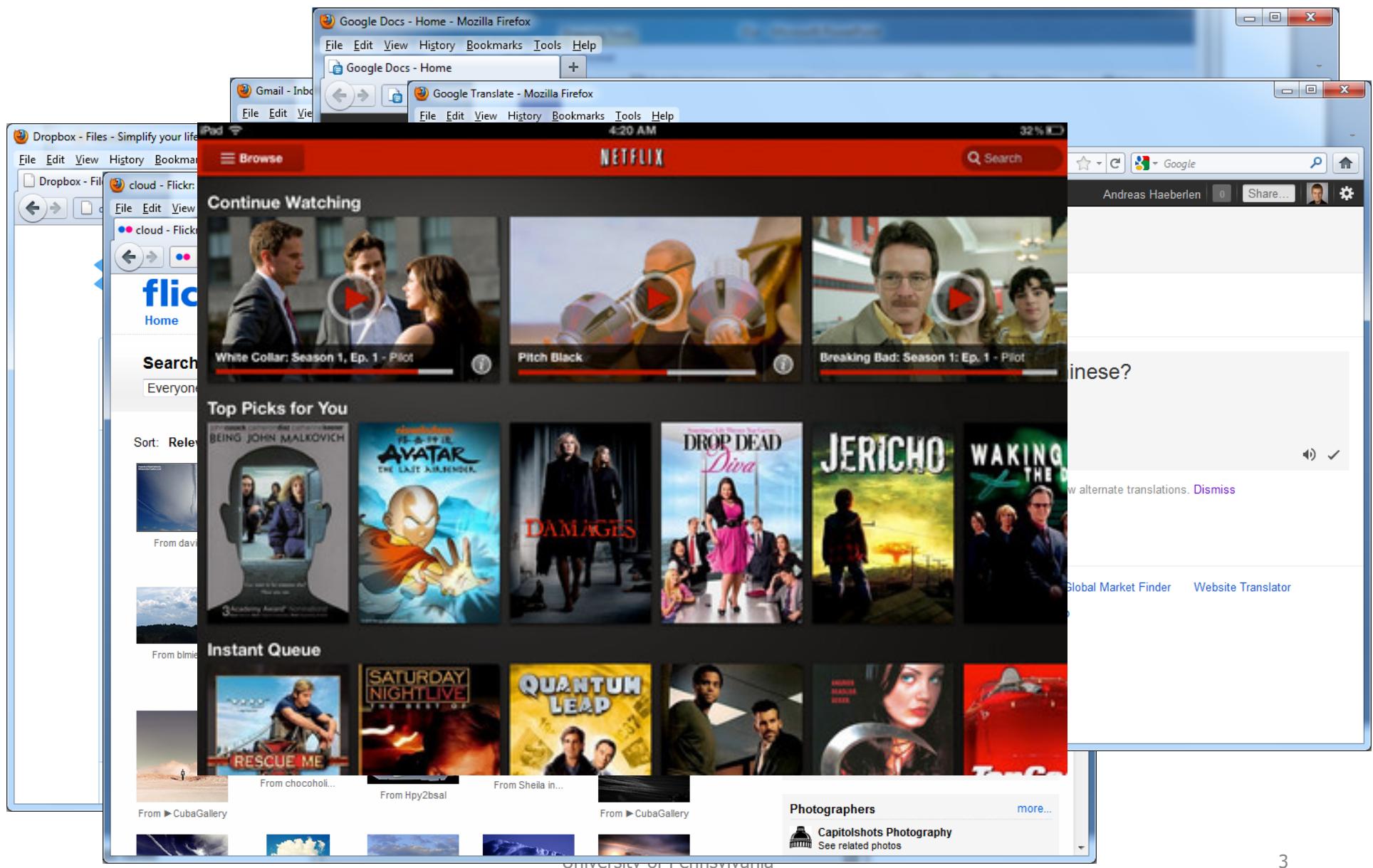
CIT 1307: Introduction to Information Technology

Cloud Computing

Overview

- Computing at Scale
 - Need for scalability
 - From PCs to Data Centers
- Cloud Computing
 - What kinds of clouds exist today?
 - What kinds of applications run on the cloud?
 - Virtualization: How clouds work 'under the hood'

Have you used these before?



What is so special about them?

- The key challenge is **scale!**
 - Lots of data, lots of users everywhere on the planet
 - Hosted on massive shared infrastructure (data centers – think computers the size of a football field!)
- Scale brings new challenges
 - Many algorithms do not work at these scales
 - Need special solutions for security, performance, ...
 - "**Big Data**": Data analysis at scale

How many users and objects?

- Flickr has >6 billion photos
- Facebook has 1.7 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 (1 million GB or a thousand TB)
 - 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 (1B GB) of storage
 - NSA Utah Data Center is said to have 5 zettabyte (thousand Exabyte))
- How much is a zettabyte?
 - 1,000,000,000,000,000,000 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!
 - Need parallel processing!
- 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.



Scaling up



PC



Server



Cluster

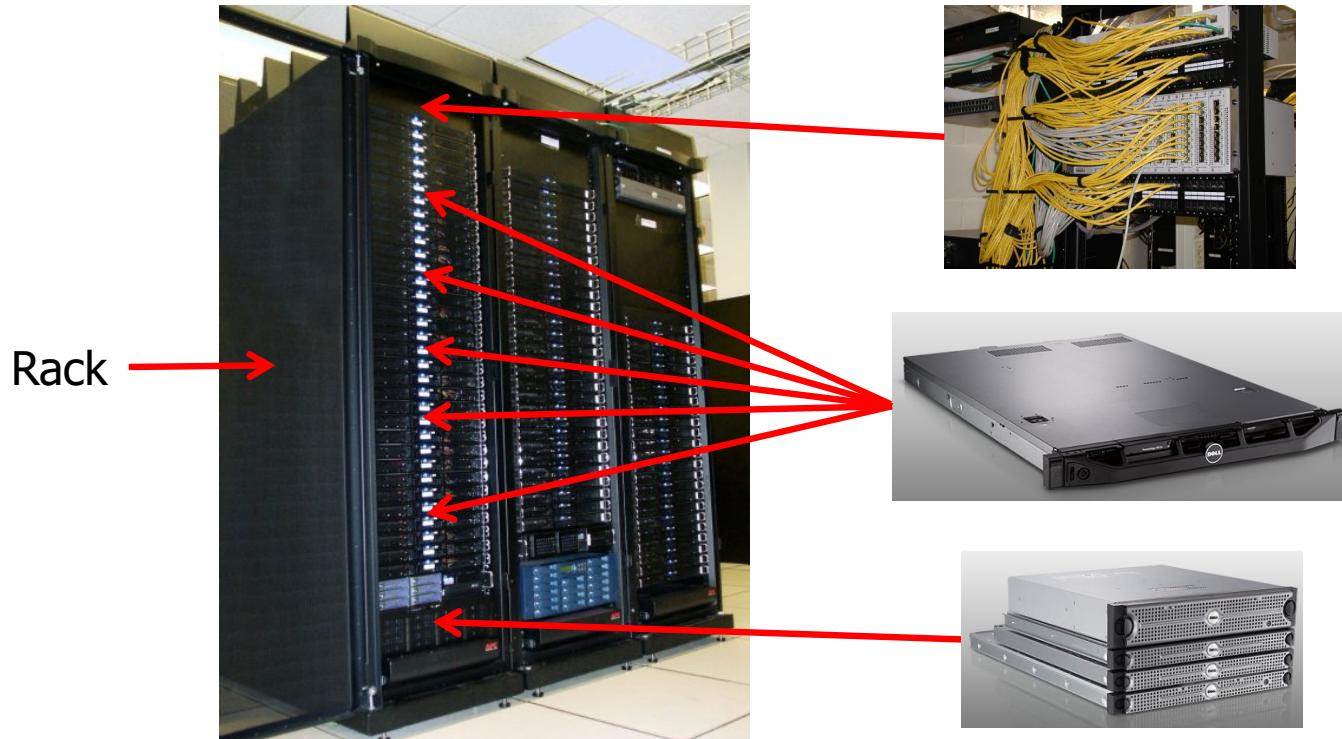


Data center



Network of data centers

Clusters



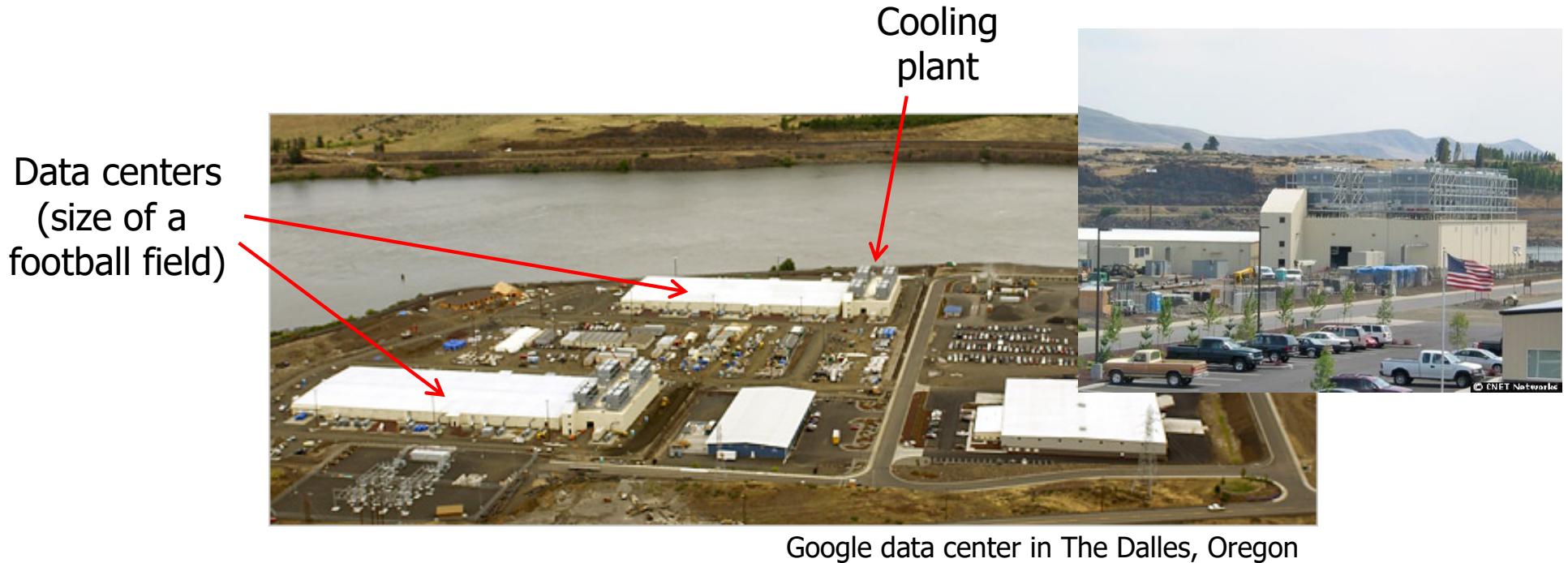
Network **switch**
(connects nodes with each other and with other racks)

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
 - Needs lot of power and massive cooling.

What does a data center look like?



- A separate building that holds the clusters and have lots of cooling and power
- 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)

Global distribution of Data Centers



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

Cloud Computing

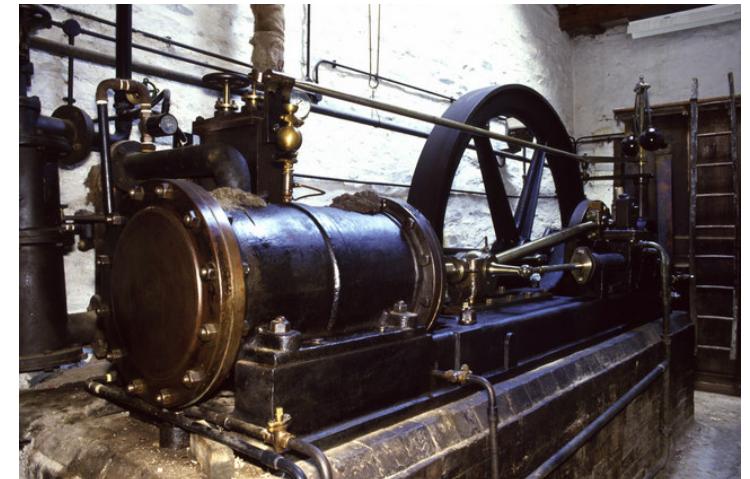
The vision...



The power plant analogy



Waterwheel at the Neuhausen ob Eck Open-Air Museum



Steam engine at Stott Park Bobbin Mill

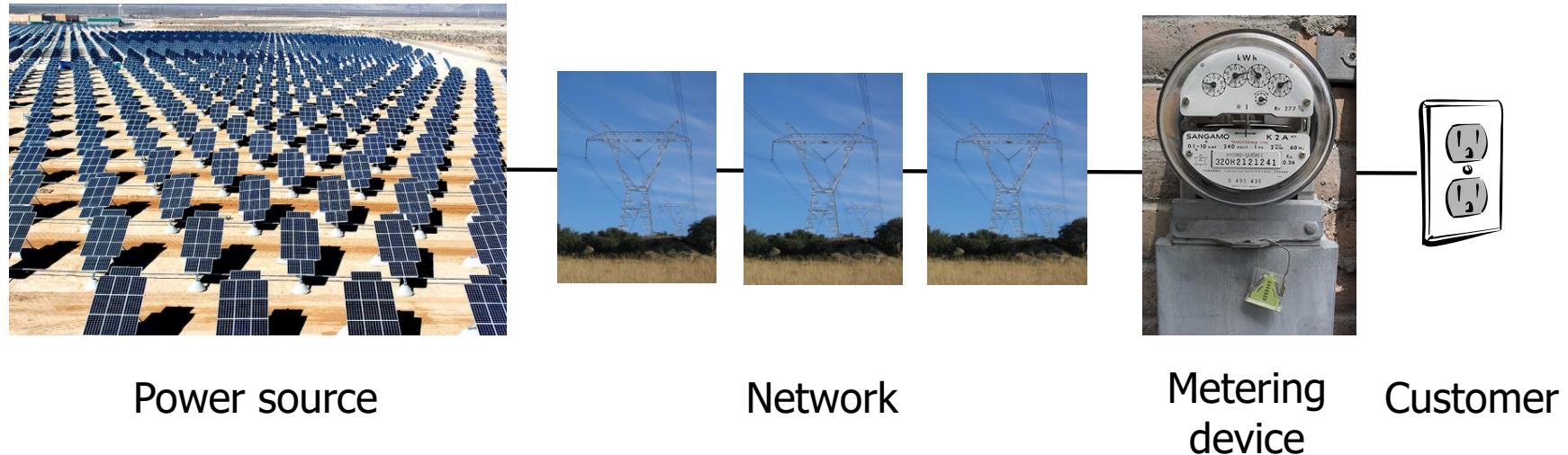
- It used to be that everyone had their own power source
 - Challenges are similar to the cluster: Needs large up-front investment, expertise to operate, difficult to scale up/down...

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

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

Computing

Cheaper to run one big data center than many small ones

High utilization!

No investment in data center; pay-as-you-go model

Thousands of computers available on demand; add more within seconds

So what is Cloud Computing?

According to *Buyya et al*

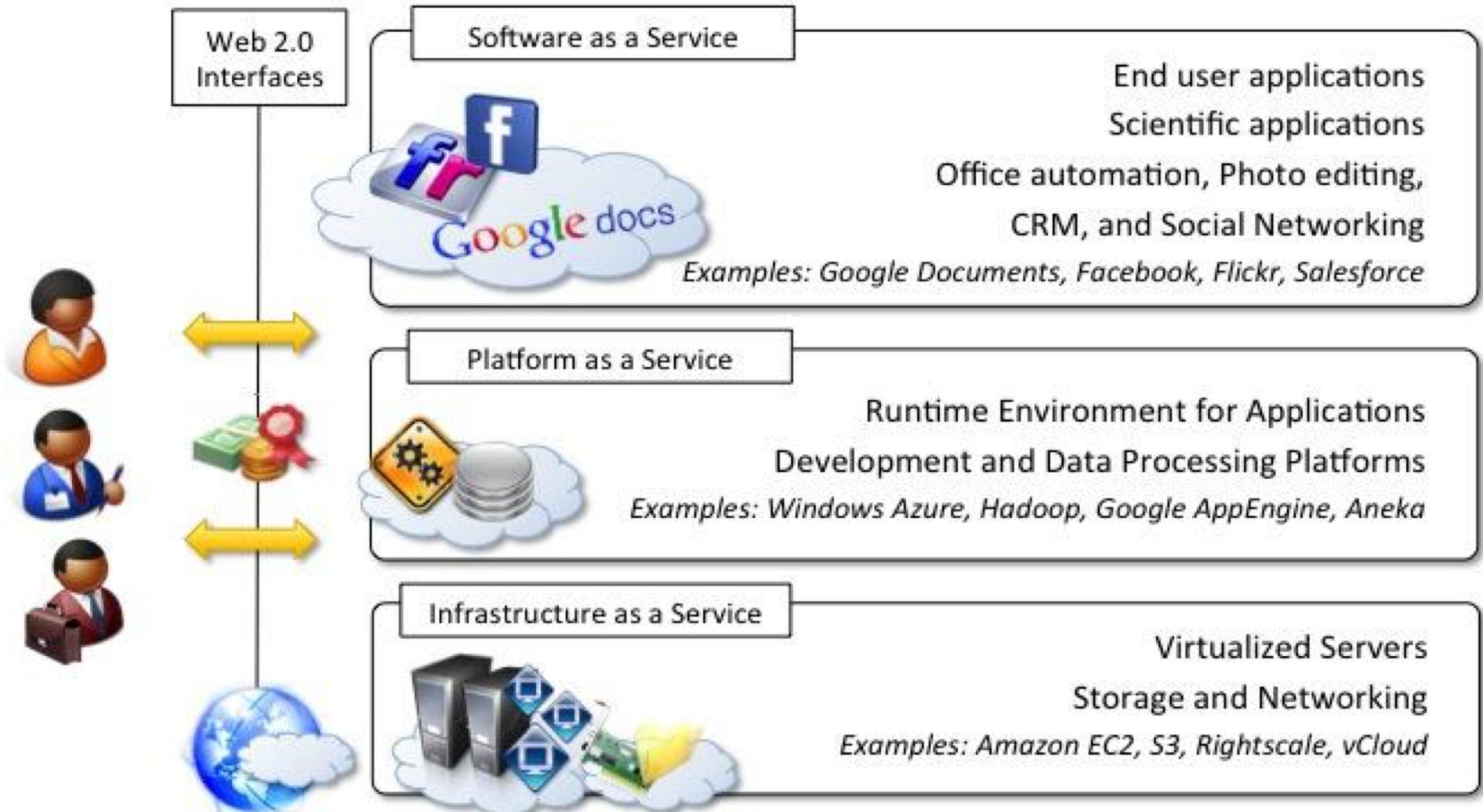
A cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers.

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

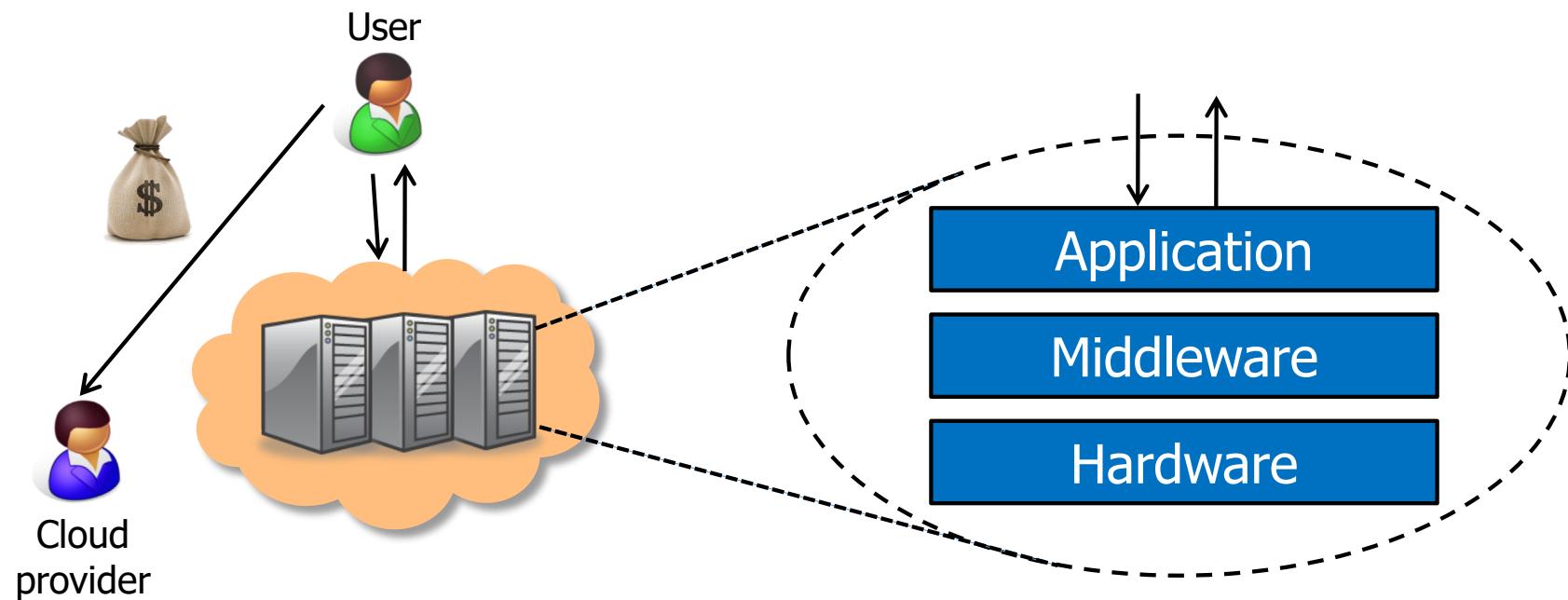
Everything as a Service

- 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

Reference Models

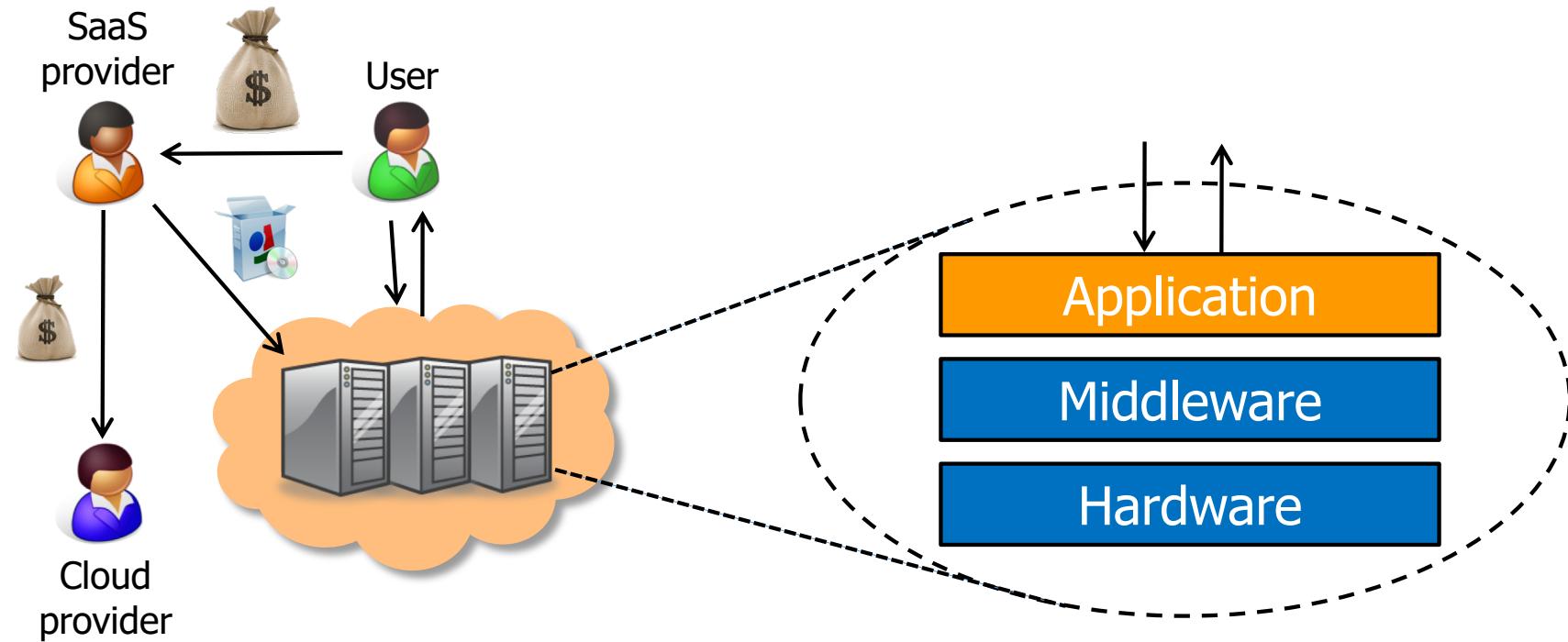


Software as a Service (SaaS)



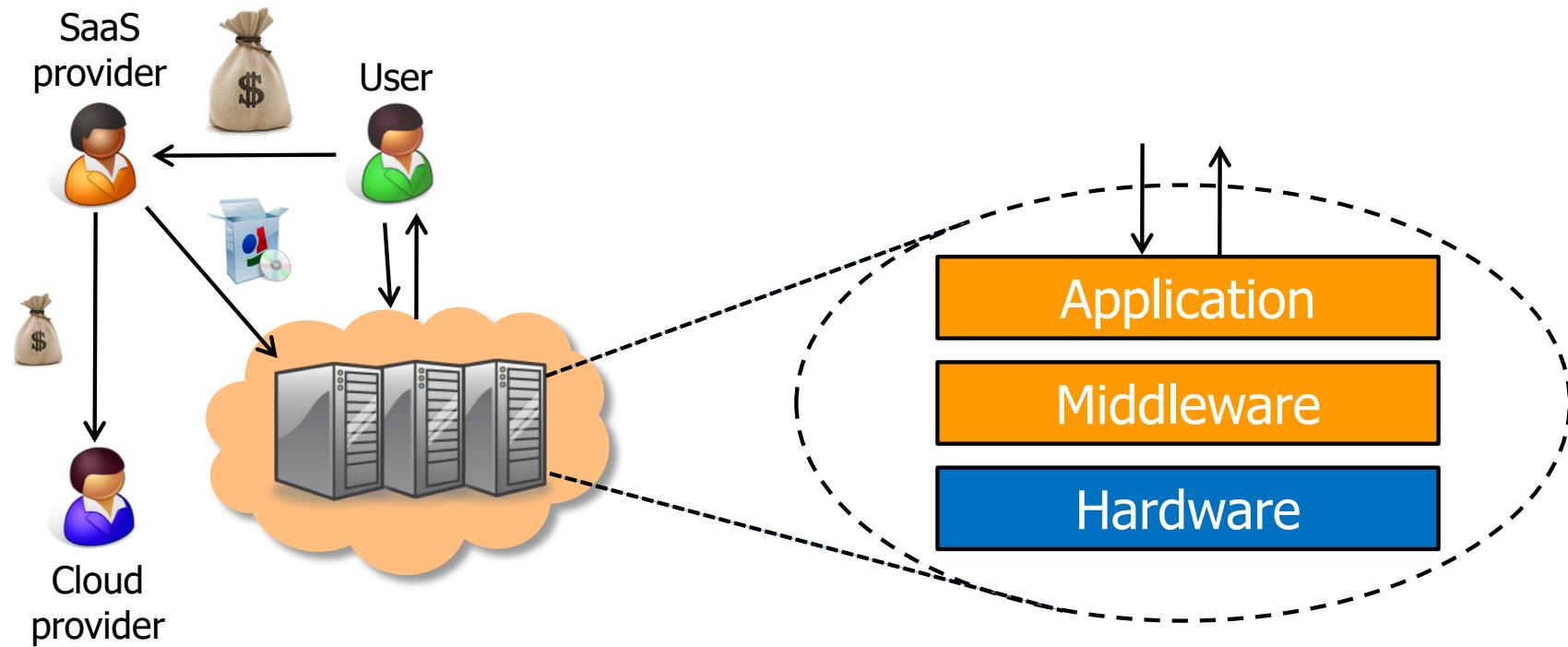
- Cloud provides an entire application
 - Word processor, spreadsheet, CRM software, calendar...
 - Customer pays cloud provider
 - Example: Google Apps, Salesforce.com

Platform as a Service (PaaS)



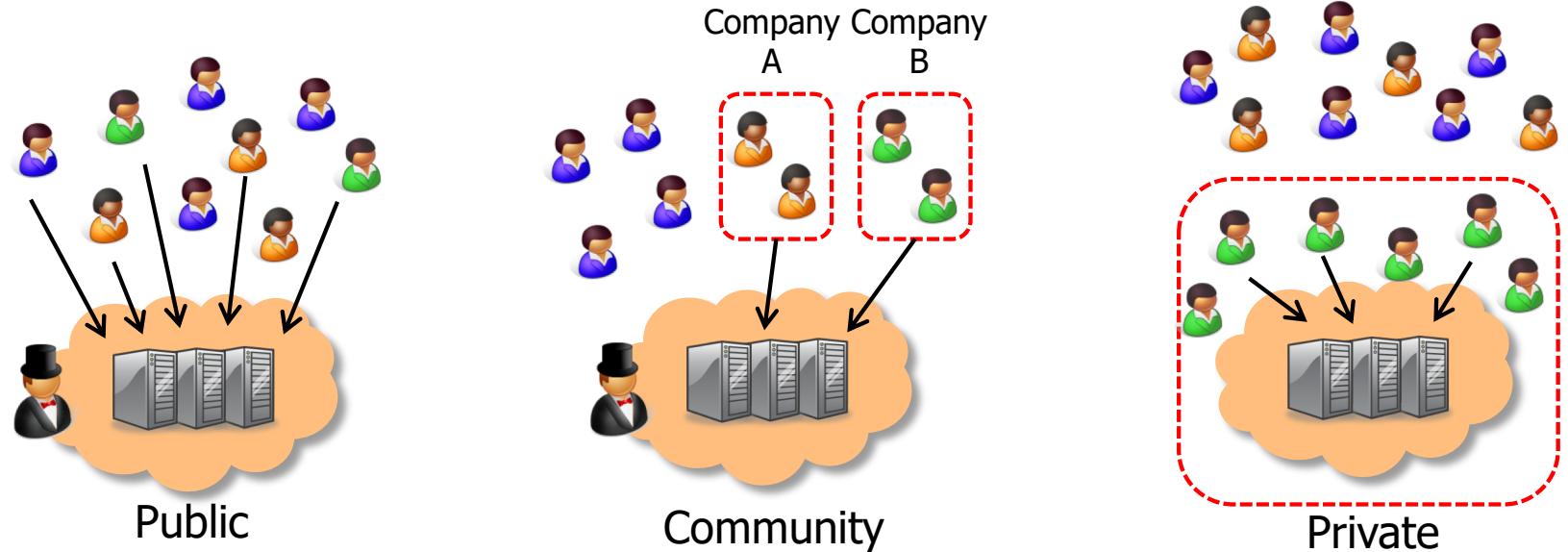
- Cloud provides middleware/infrastructure
 - For example, Microsoft Common Language Runtime (CLR) manages the execution of .NET programs.
 - Customer pays SaaS provider for the service; SaaS provider pays the cloud for the infrastructure
 - Example: Windows Azure, Google App Engine

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: Amazon's "GovCloud"
 - **Private cloud:** Shared within a single organization. Example: Internal datacenter of a large company.

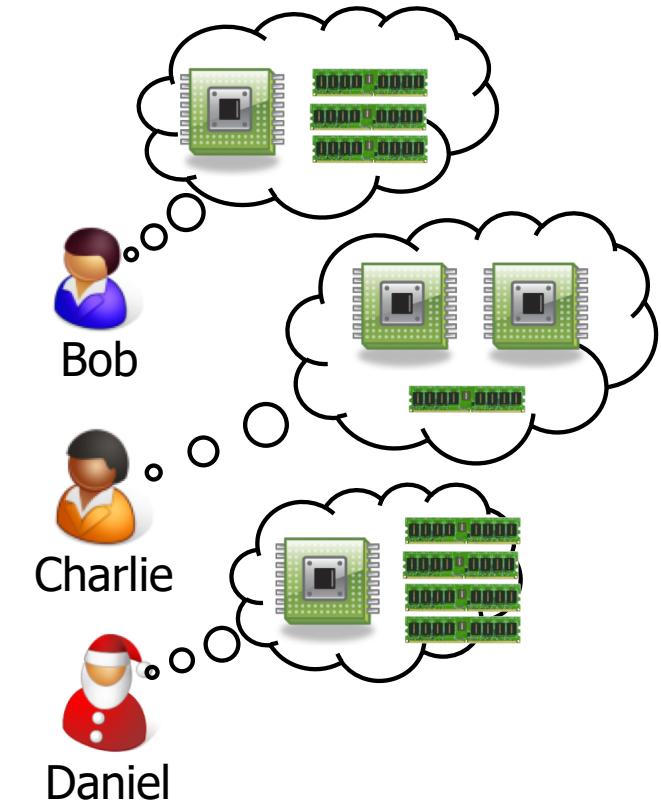
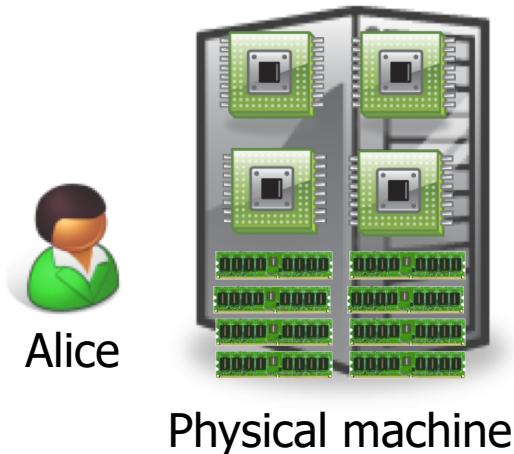
Examples of cloud applications

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

Case study: The Washington Post

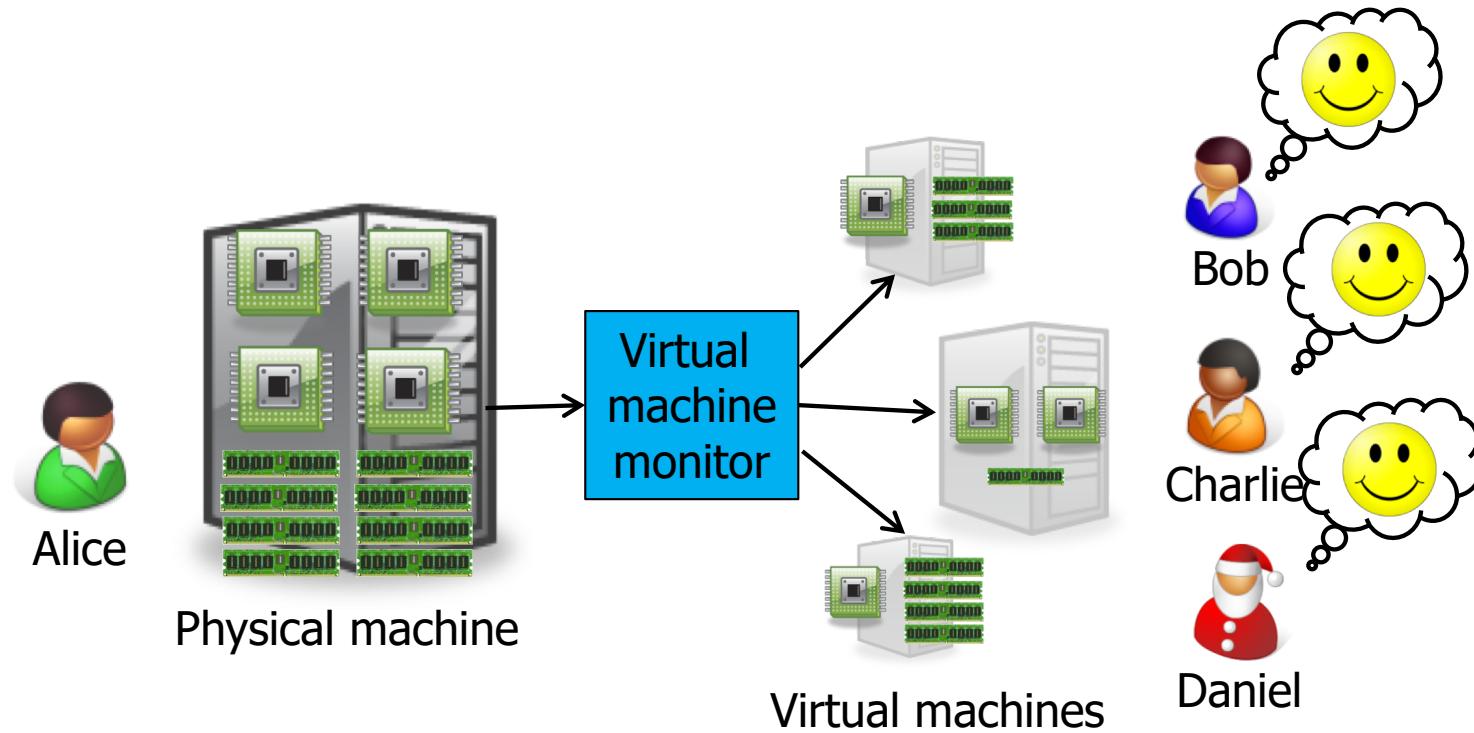
- March 19, 2008: Hillary Clinton's official White House schedule released to the public
 - 17,481 pages of non-searchable, low-quality PDF
 - Very interesting to journalists, but would have required hundreds of man-hours to evaluate
 - Peter Harkins, Senior Engineer at The Washington Post:
Can we make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

What is virtualization?



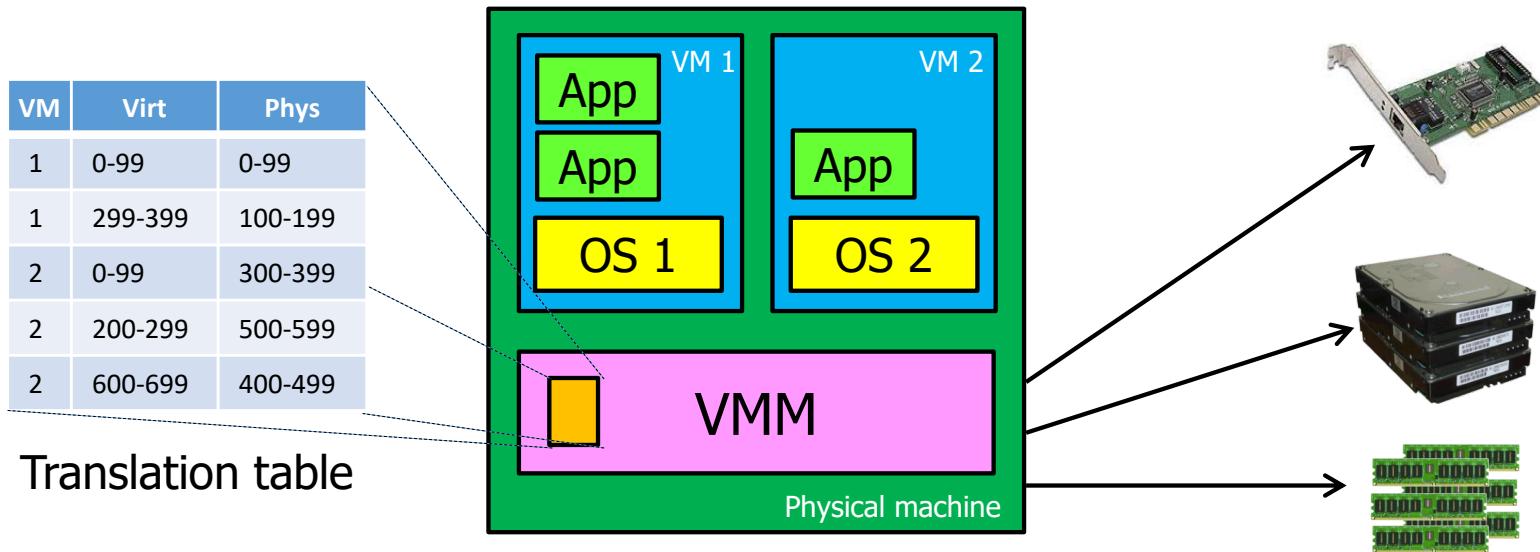
- Suppose Alice has a machine with 4 CPUs and 8 GB of memory, and three customers:
 - Bob wants a machine with 1 CPU and 3GB of memory
 - Charlie wants 2 CPUs and 1GB of memory
 - Daniel wants 1 CPU and 4GB of memory
- What should Alice do?

What is virtualization?



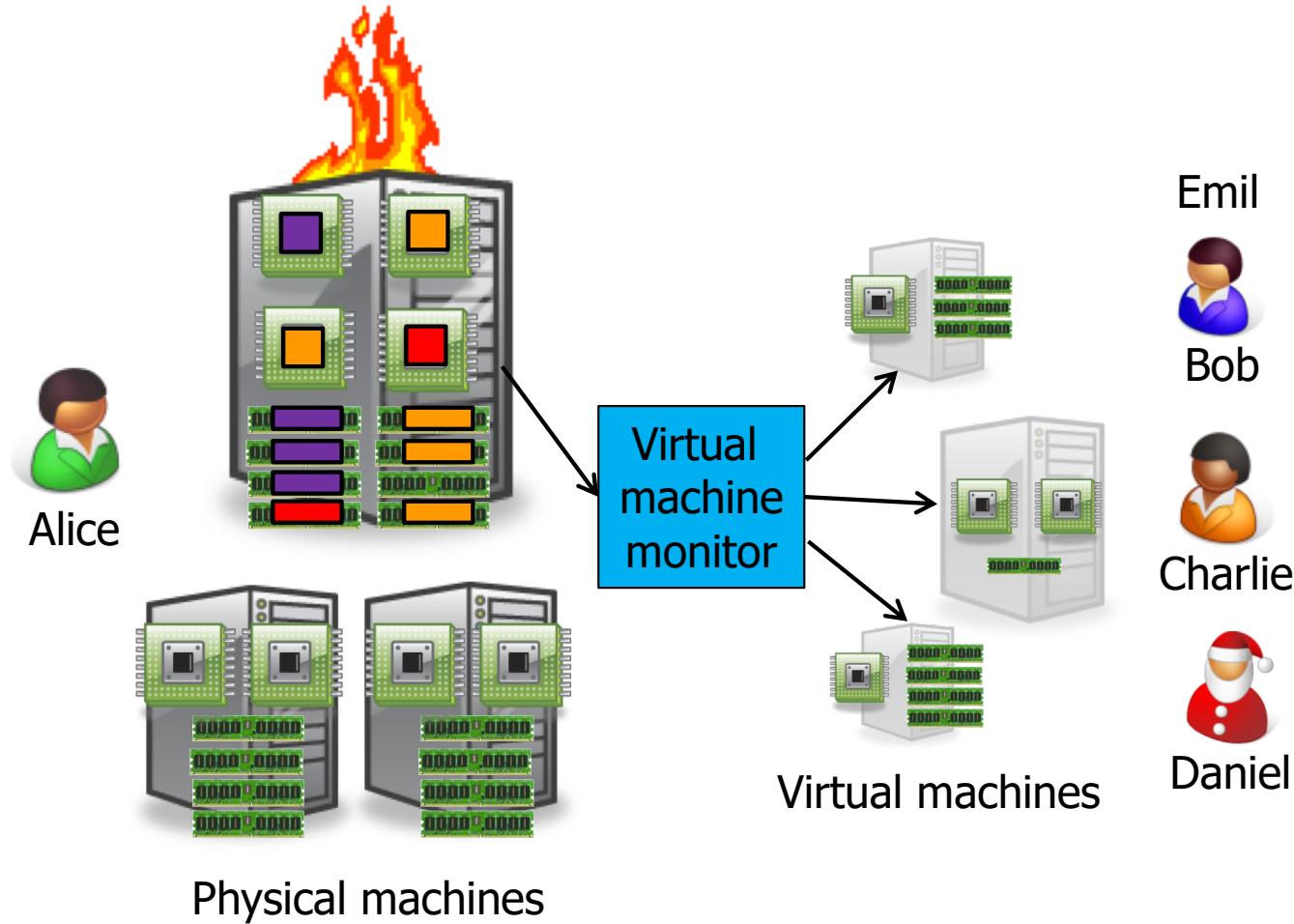
- Alice can sell each customer a **virtual machine** (VM) with the requested resources
 - From each customer's perspective, it appears as if they had a physical machine all by themselves (**isolation**)

How does it work?



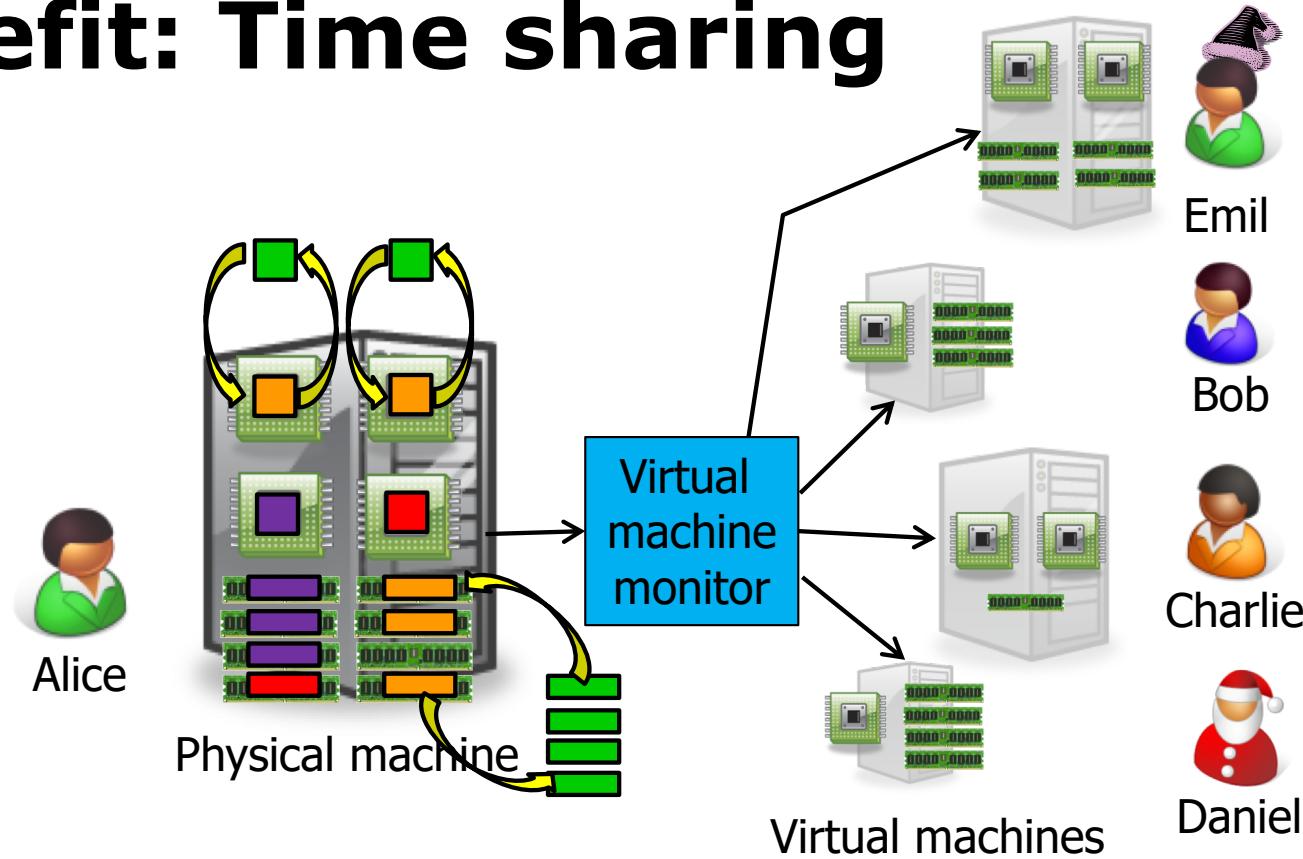
- Resources (CPU, memory, ...) are virtualized
 - VMM ("Hypervisor") has translation tables that map requests for virtual resources to physical resources
 - Example: VM 1 accesses memory cell #323; VMM maps this to memory cell 123.

Benefit: Migration



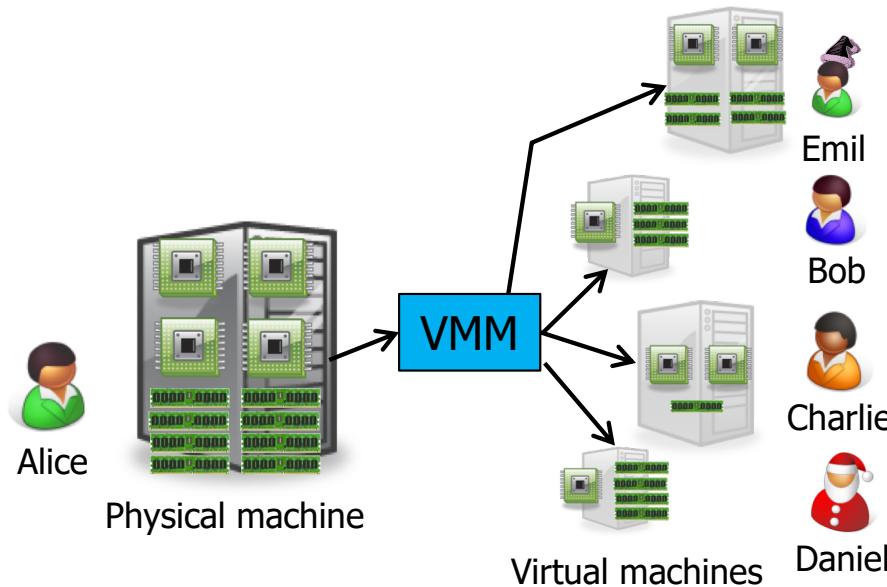
- What if the machine needs to be shut down?
 - e.g., for maintenance, consolidation, ...
 - Alice can **migrate** the VMs to different physical machines without any customers noticing

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 machine 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

Cloud Computing Challenges

1. Availability

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

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?

| 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 prominent cloud outages

Cloud Computing Challenges

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

What are the benefits to a (regular) business?

- No up-front commitments
- On-demand access
- Nice pricing (capital costs -> utility costs, no depreciation)
- Simplified app acceleration and scalability
- Efficient resource allocation
- Energy efficiency??
- Seamless creation and use of third-party services

Benefits for a software company going to SaaS?

- NO deployment issues (CDs, downloads, etc)
- No need to support multiple OSs
- Faster to market
- A/B testing of features
- Efficiency and reliability now key
- More efficient developers, just try it!

Problems?

- Security
 - Confidentiality, Secrecy, Protection
- Legal
 - Google/Facebook privacy
 - Differing viewing laws
- Performance & Data Location

Acknowledgement

- A. Haeberlen, Upenn NETS212
- Rajkumar Buyya, “Mastering Cloud Computing Foundations and Applications Programming”