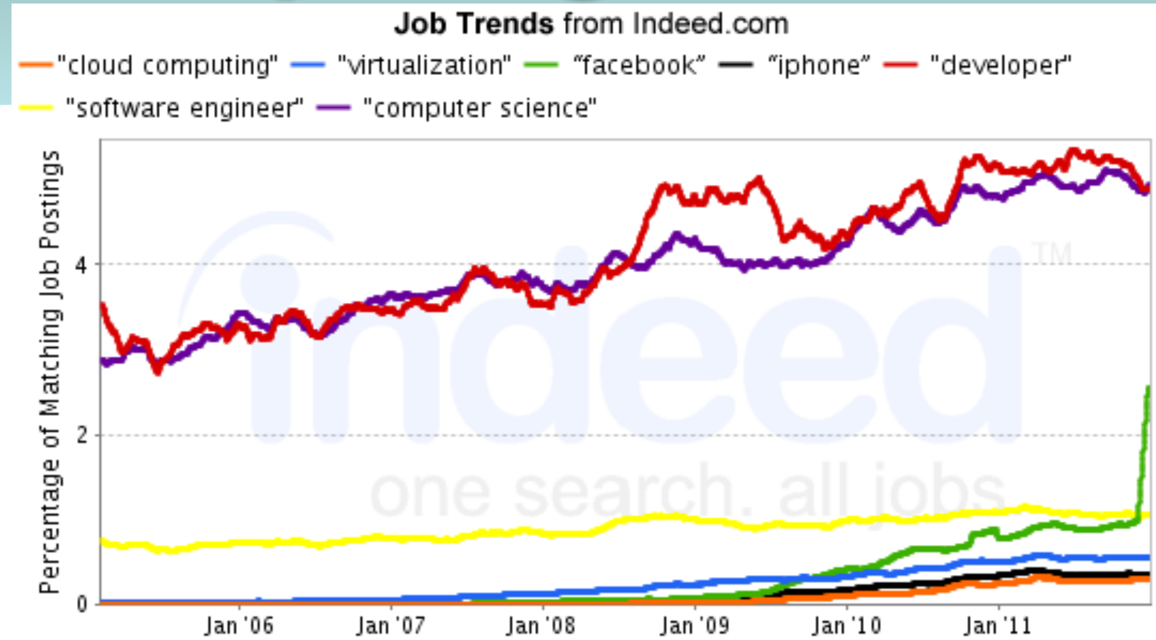
Design of Cloud Computing Platforms

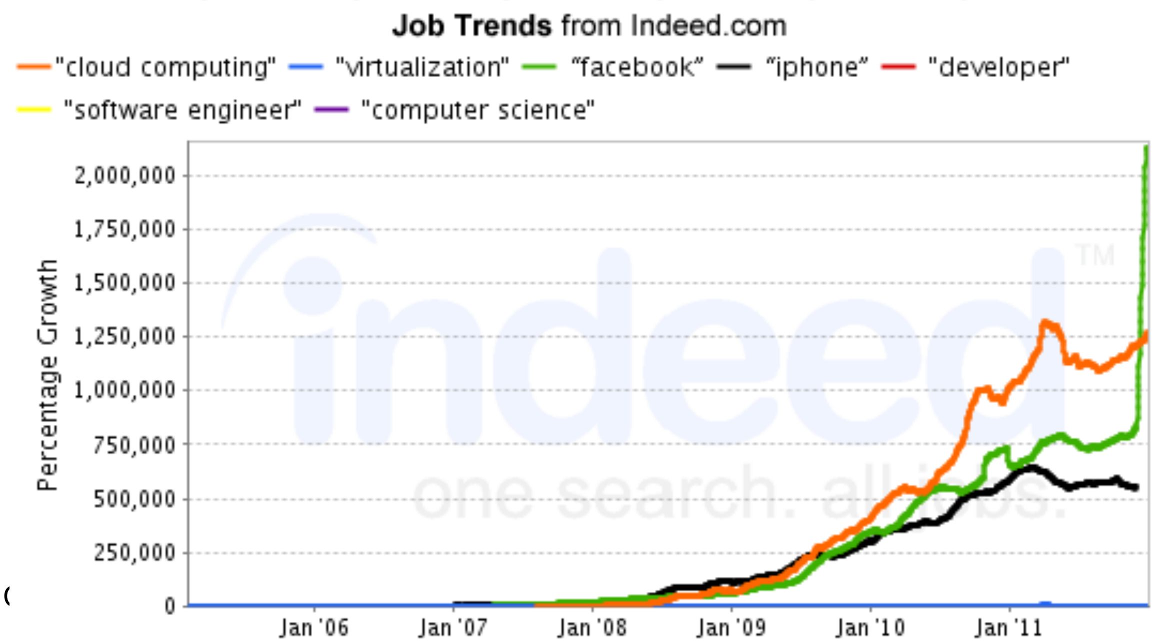
Ioan Raicu
Computer Science Department
Illinois Institute of Technology

CS 553: Cloud Computing
April 2nd, 2018

Cloud Computing

- Value of Cloud computing services:
  - 2008 : $46.41 billion
  - 2009 : $56.30 billion
  - 2013 : $150.1 billion





Cloud Computing

- Cluster, grid, and utility computing ➔ Cloud Computing
  - Cluster and grid computing leverage the use of many computers in parallel to solve problems of any size
  - Utility and Software as a Service (SaaS) provide computing resources as a service with the notion of pay per use
  - Cloud computing leverages dynamic resources to deliver large numbers of services to end users

Public, Private, and Hybrid Clouds

- Public Cloud:
  - Built over the Internet and can be accessed by any user who has paid for the service
  - Owned by service providers and are accessible through a subscription, typically offered on a flexible price-per-use basis
  - Examples: Google App Engine (GAE), Amazon Web Services (AWS), Microsoft Azure, IBM Blue Cloud, and Salesforce.com's Force.com
  - Promote standardization, preserve capital investment, and offer application flexibility
- Private Cloud:
- Hybrid Cloud:

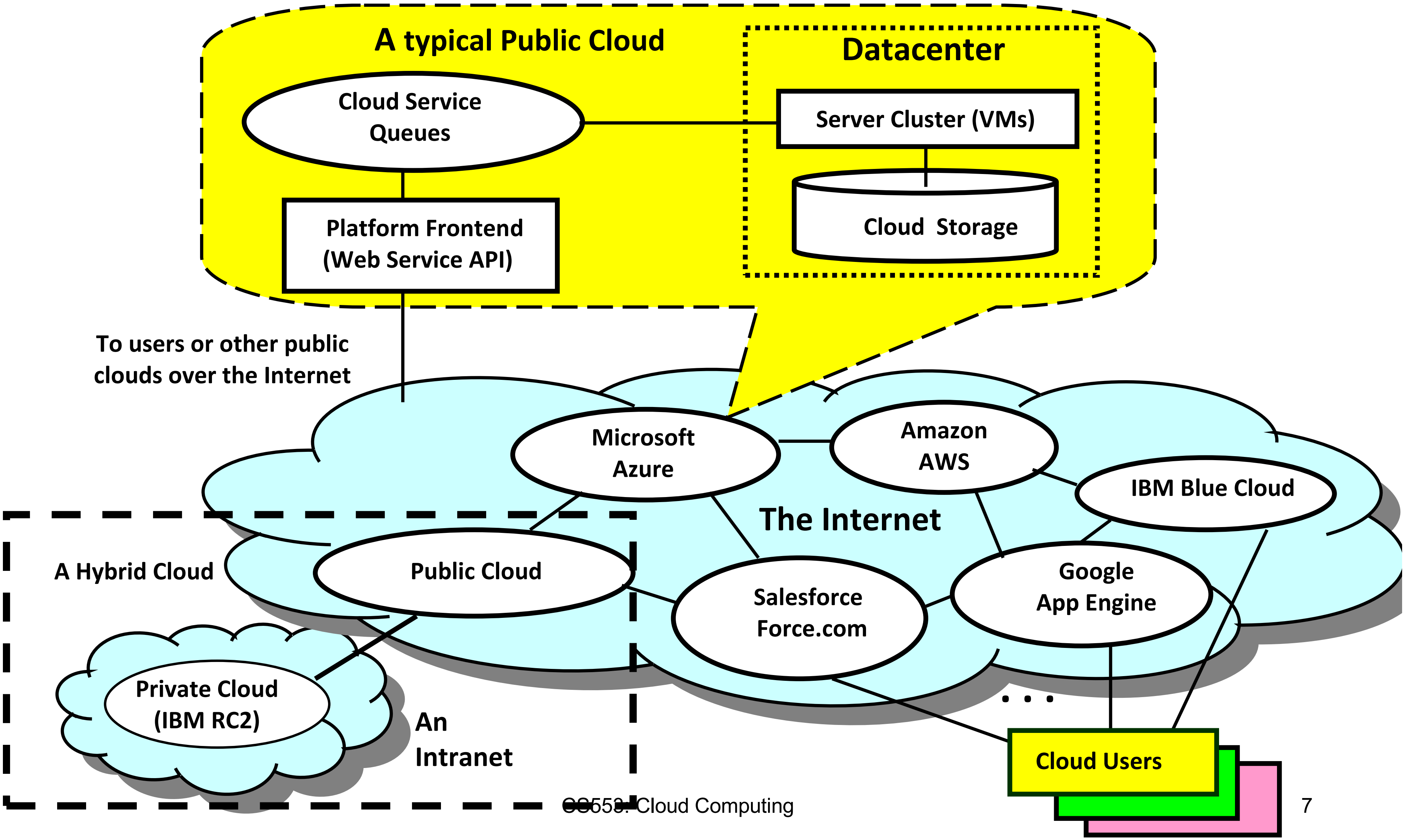
Public, Private, and Hybrid Clouds

- Public Cloud:
- Private Cloud:
  - Built within the domain of an intranet owned by a single organization
  - It is client owned and managed, and its access is limited to the owning clients and their partners
  - Its deployment was not meant to sell capacity over the Internet through publicly accessible interfaces
  - Private clouds give local users a flexible and agile private infrastructure to run service workloads within their administrative domains
  - Aimed to deliver more efficient and convenient cloud services
  - Example: *Research Compute Cloud* (RC2)
    - Built by IBM interconnecting the computing and IT resources at eight IBM Research Centers scattered throughout the United States, Europe, and Asia
  - Attempt to achieve customization and offer higher efficiency, resiliency, security, and privacy
- Hybrid Cloud:

Public, Private, and Hybrid Clouds

- Public Cloud:
- Private Cloud:
- Hybrid Cloud:
  - Built with both public and private clouds
  - Private clouds can also support a hybrid cloud model by supplementing local infrastructure with computing capacity from an external public cloud
  - Provides access to clients, the partner network, and third parties
  - Operate in the middle between private and public clouds, with many compromises in terms of resource sharing

Public, Private, and Hybrid Clouds



Cloud Ecosystem and Enabling Technologies

Classical Computing
(Repeat the following cycle every 18 months)

Buy and own
Hardware, system software, applications to meet peak needs
Install, configure, test, verify, evaluate, manage
- - - -
Use (Finally)
- - - -
Pay $$$$$ (High cost)

Cloud Computing
(Pay as you go per each service provided)

Subscribe
- - - -

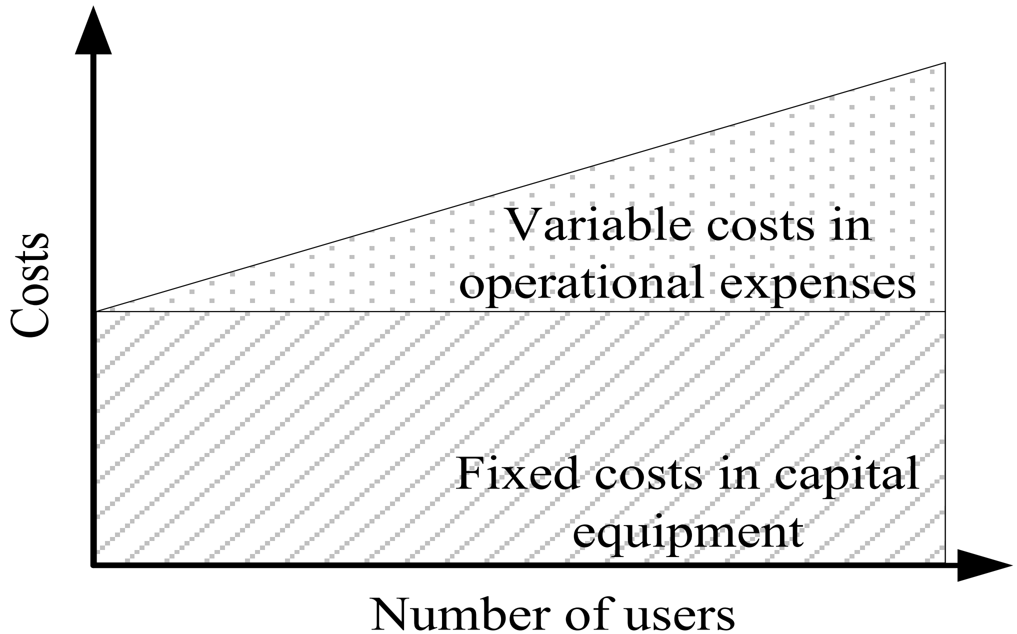
Use (Save about 80-15% of the total cost)
- - - -

$ - Pay for what you use
based on the QoS

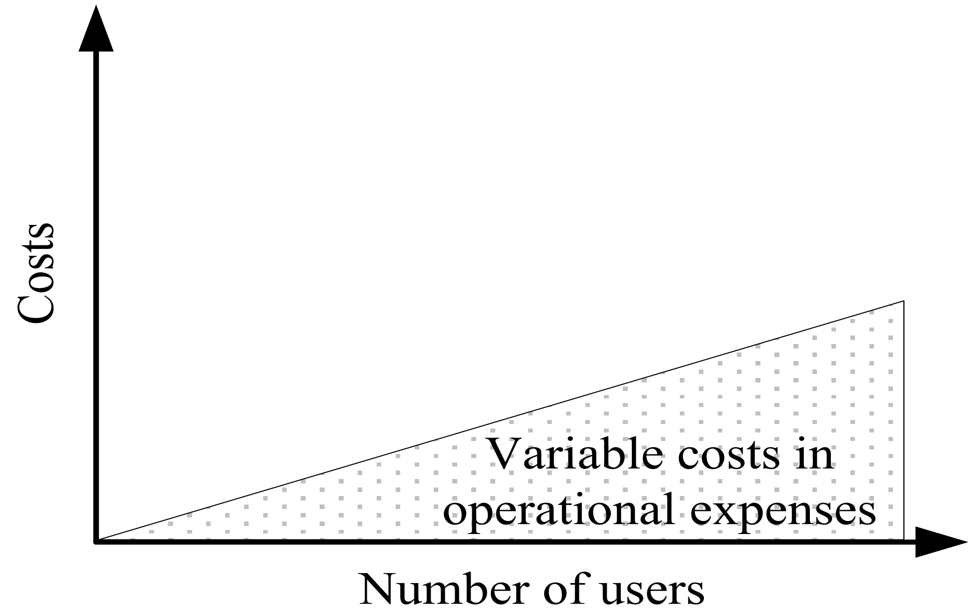
Cloud Ecosystem and Enabling Technologies

- **Cloud Design Objectives:**
  - **Shifting computing from desktops to data centers**
    - Computer processing, storage, and software delivery is shifted away from desktops and local servers and toward data centers over the Internet.
  - **Service provisioning and cloud economics**
    - Providers supply cloud services by signing SLAs with consumers and end users. The services must be efficient in terms of computing, storage, and power consumption. Pricing is based on a pay-as-you-go policy.
  - **Scalability in performance**
    - The cloud platforms and software and infrastructure services must be able to scale in performance as the number of users increases.
  - **Data privacy protection**
    - Can you trust data centers to handle your private data and records? This concern must be addressed to make clouds successful as trusted services.
  - **High quality of cloud services**
    - The QoS of cloud computing must be standardized to make clouds interoperable among multiple providers.
  - **New standards and interfaces**
    - This refers to solving the data lock-in problem associated with data centers or cloud providers. Universally accepted APIs and access protocols are needed to provide high portability and flexibility of virtualized applications.

Cloud Ecosystem and Enabling Technologies

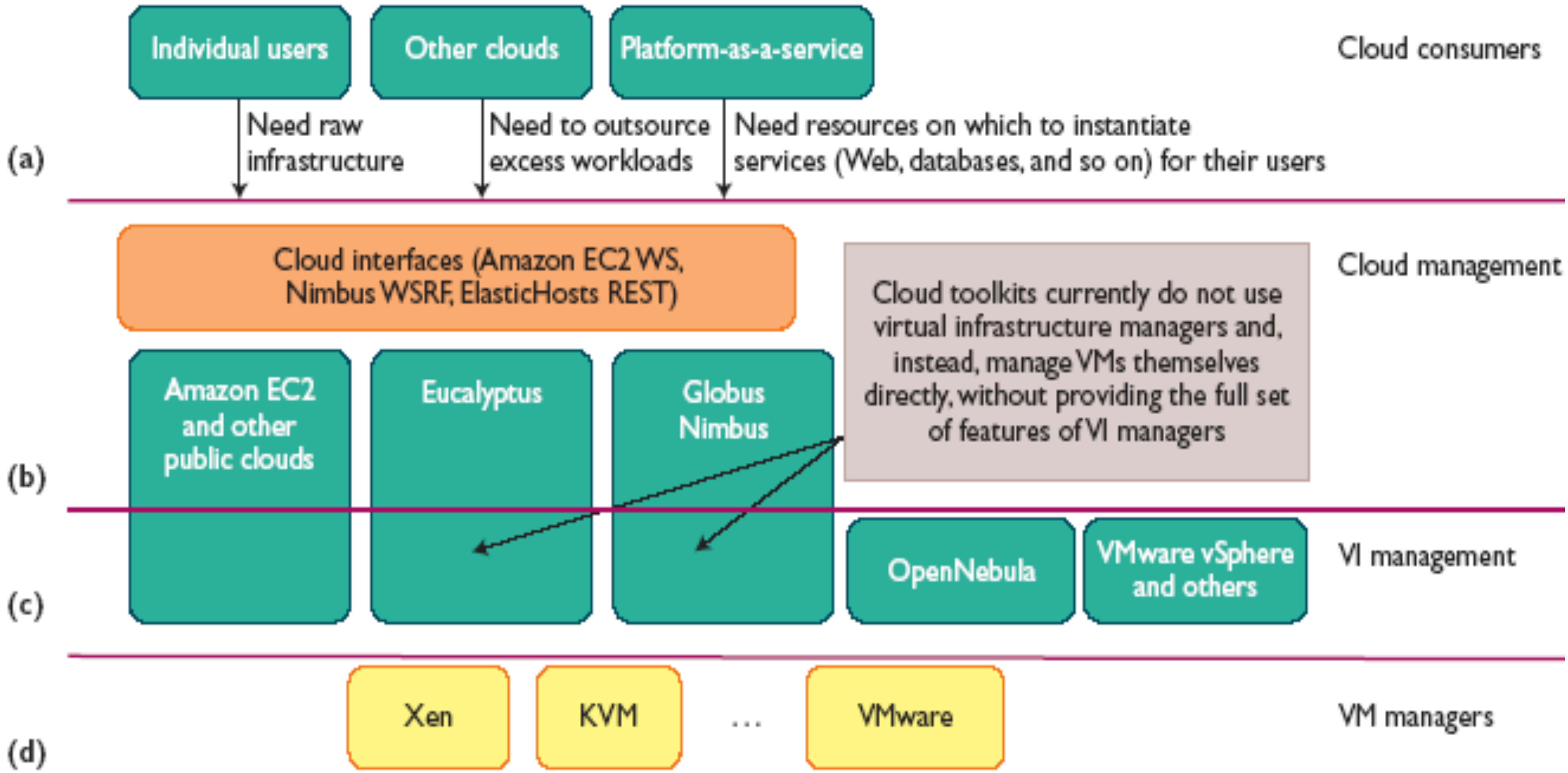


Traditional IT Cost Model



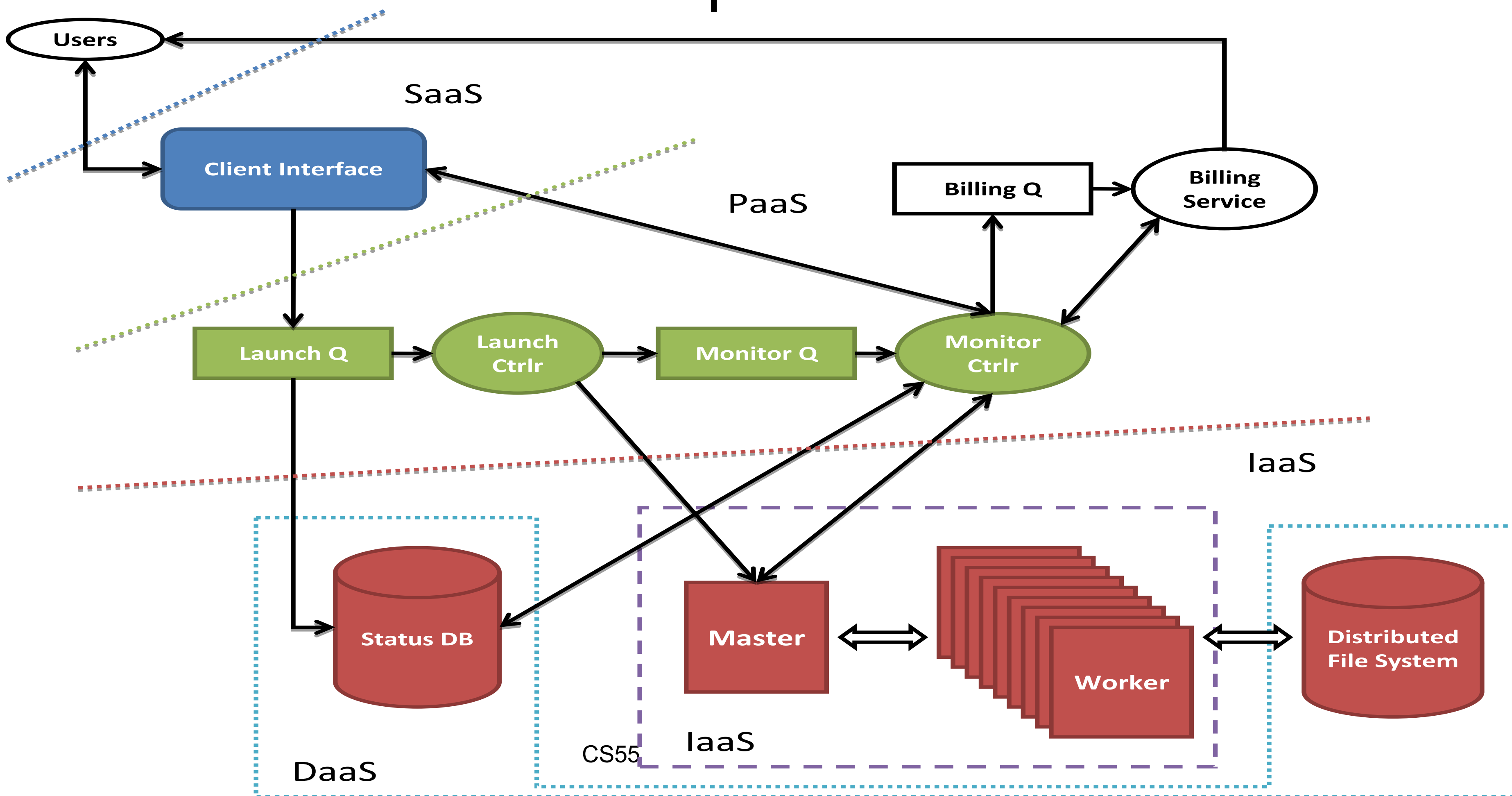
Cloud Computing Cost Model

Cloud Ecosystem and Enabling Technologies



Infrastructure-as-a-Service (IaaS)

- SaaS is often built on top of the PaaS
- PaaS is often built on top of IaaS



Infrastructure-as-a-Service (IaaS)

- IaaS allows users to use virtualized IT resources for computing, storage, and networking
- Users deploy and run their applications over their chosen OS environment
- Users do not manage or control the underlying cloud infrastructure, but have control over the OS, storage, deployed applications, and possibly select networking components
- IaaS model encompasses *storage as a service*, compute *instances as a service*, and *communication as a service*
- Many startup cloud providers have appeared in recent years
  - GoGrid, FlexiScale, and Aneka

Infrastructure-as-a-Service (IaaS)

- IaaS public cloud providers

Cloud Name	VM Instance Capacity	API and Access Tools	Hypervisor, Guest OS
Amazon EC2	Each instance has 1–20 EC2 processors, 1.7–15 GB of memory, and 160–1.69 TB of storage.	CLI or Web Service (WS) portal	Xen, Linux, Windows
GoGrid	Each instance has 1–6 CPUs, 0.5–8 GB of memory, and 30–480 GB of storage.	REST, Java, PHP, Python, Ruby	Xen, Linux, Windows
Rackspace Cloud	Each instance has a four-core CPU, 0.25–16 GB of memory, and 10–620 GB of storage.	REST, Python, PHP, Java, C#, .NET	Xen, Linux
FlexiScale in the UK	Each instance has 1–4 CPUs, 0.5–16 GB of memory, and 20–270 GB of storage.	Web console	Xen, Linux, Windows
Joyent Cloud	Each instance has up to eight CPUs, 0.25–32 GB of memory, and 30–480 GB of storage.	No specific API, SSH, Virtual/Min	OS-level virtualization, OpenSolaris

Platform-as-a-Service (PaaS)

- Enables the development, deployment, and manageability of the execution of applications using provisioned resources demands a cloud platform with the proper software environment
- Such a platform includes operating system and runtime library support
- The platform cloud is an integrated computer system consisting of both hardware and software infrastructure
- The user application can be developed on this virtualized cloud platform using some programming languages and software tools supported by the provider (e.g., Java, Python, .NET)
- The user does not manage the underlying cloud infrastructure
- The cloud provider supports user application development and testing on a well-defined service platform
- Enables a collaborated software development platform for users from different parts of the world
- Encourages third parties to provide software management, integration, and service monitoring solutions

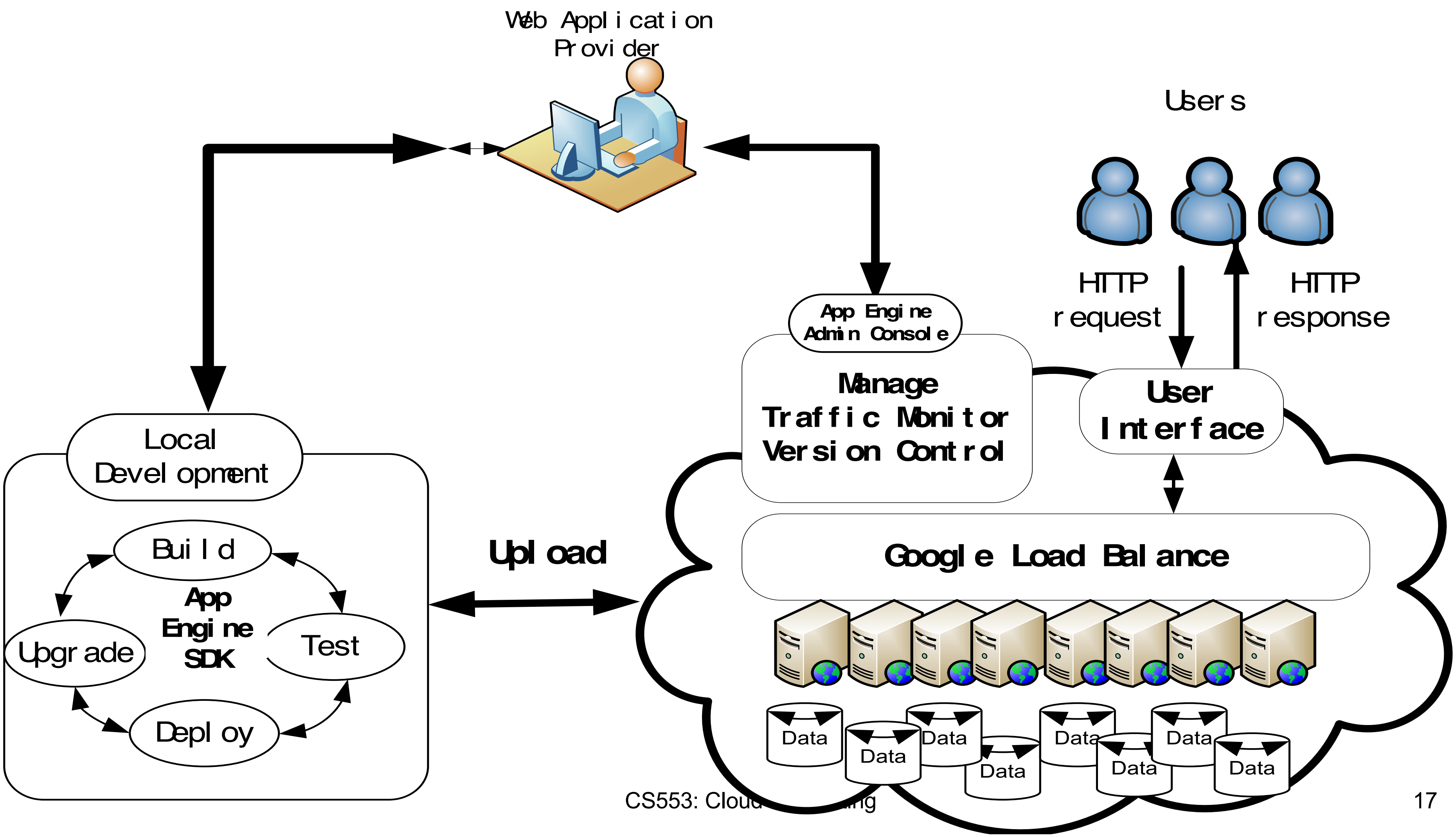
Platform-as-a-Service (PaaS)

- Public PaaS cloud providers

Cloud Name	Languages and Developer Tools	Programming Models Supported by Provider	Target Applications and Storage Option
Google App Engine	Python, Java, and Eclipse-based IDE	MapReduce, Web programming on demand	Web applications and BigTable storage
Salesforce.com's Force.com	Apex, Eclipse-based IDE, Web-based Wizard	Workflow, Excel-like formula, Web programming on demand	Business applications such as CRM
Microsoft Azure	.NET, Azure tools for MS Visual Studio	Unrestricted model	Enterprise and Web applications
Amazon Elastic MapReduce	Hive, Pig, Cascading, Java, Ruby, Perl, Python, PHP, R, C++	MapReduce	Data processing and e-commerce
Aneka	.NET, stand-alone SDK	Threads, task, MapReduce	.NET enterprise applications, HPC

Platform-as-a-Service (PaaS)

- **Google App Engine platform for PaaS operations**



Logistics

- PA2 part A is out → due next Wednesday
- PA2 part B will come next week, and be due 2 weeks later
- Don't forget about the project, due in less than 3 weeks
- A final written assignment will come mid-April, and be due April 27th

Sort

- Many sorting algorithms:
  - https://en.wikipedia.org/wiki/Sorting_algorithm
- Find one that is O(n lg n) time complexity
  - E.g. merge sort, quick sort
  - https://en.wikipedia.org/wiki/Merge_sort
  - https://en.wikipedia.org/wiki/Quicksort
- Must implement from scratch; can use online resources to learn about these algorithms (e.g. wikipedia); do not copy and paste code; cite your work

External Sort

- External sort is required when data to be sorted does not fit in memory
  - https://en.wikipedia.org/wiki/External_sorting
- Merge sort can be adapted:
  - https://en.wikipedia.org/wiki/External_sorting#External_merge_sort
- It involves chunking up data into smaller pieces that fit in memory
- It will require more than 1 read and write to disk

Software-as-a-Service (SaaS)

- Provides software applications as a service
- Browser-initiated application software over thousands of cloud customers
- On the customer side, there is no upfront investment in servers or software licensing
- On the provider side, costs are kept rather low, compared with conventional hosting of user applications
- Customer data is stored in the cloud that is either vendor proprietary or publicly hosted to support PaaS and IaaS
- Examples of SaaS services:
  - Google Gmail and docs
  - Microsoft SharePoint
  - CRM software from Salesforce.com

Data-Center Design and Interconnection Networks

- A data center is often built with a large number of servers through a huge interconnection network
- Study the design of large-scale data centers and small modular data centers that can be housed in a 40-ft truck container
- Study the interconnection of modular data centers and their management issues and solutions

Warehouse-Scale Data-Center Design

- A huge data center that Is 11 times the size of a football field, housing 400,000 to 1 million servers

Warehouse-Scale Data-Center Design

- Data centers are built economics of scale - meaning lower unit cost for larger data centers
- Small data centers could have 1,000 servers
- The larger the data center, the lower the operational cost
- The approximate monthly cost to operate a 400-server data center is estimated by:
  - Network cost $13/Mbps
  - Storage cost $0.4/GB
  - Administration costs
- The network cost to operate a small data center is about seven times greater and the storage cost is 5.7 times greater

Warehouse-Scale Data-Center Design

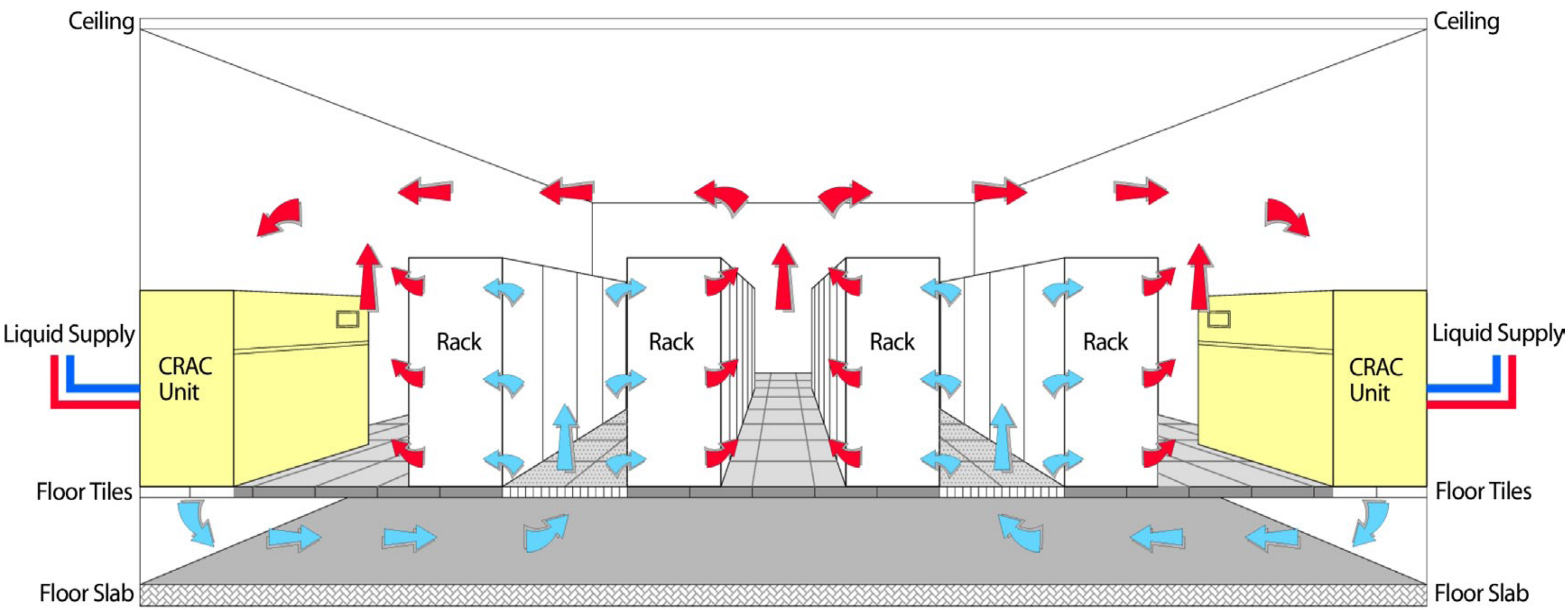
- **Data-Center Construction Requirements:**
  - Most data centers are built with commercially available components
  - An off-the-shelf server consists of a number of processor sockets, each with a multicore CPU and its internal cache hierarchy, local shared and coherent DRAM, and a number of directly attached disk drives
  - The DRAM and disk resources within the rack are accessible through first-level rack switches and all resources in all racks are accessible via a cluster-level switch
  - Consider a data center built with 2,000 servers, each with 8 GB of DRAM and four 1 TB disk drives. Each group of 40 servers is connected through a 1 Gbps link to a rack-level switch that has an additional eight 1 Gbps ports used for connecting the rack to the cluster-level switch.
- Disk bandwidth changes drastically between local and off-rack access
  - Local disks is 200 MB/s, whereas the bandwidth from off-rack disks is 25 MB/s via shared rack uplinks
  - The total disk storage in the cluster is almost 10 million times larger than local DRAM
  - A large application must deal with large discrepancies in latency, bandwidth, and capacity
- In a very large-scale data center, components are relatively cheaper, and are very different from those in building supercomputer systems

Warehouse-Scale Data-Center Design

- With a scale of thousands of servers, concurrent failure, either hardware failure or software failure, of 1 percent of nodes is common
- Many failures can happen in hardware
  - CPU failure, disk I/O failure, and network failure
- It is possible that the whole data center does not work in the case of a power crash
- Some failures are software related
- The service and data should not be lost in a failure situation
- Reliability can be achieved by redundant hardware
- The software must keep multiple copies of data in different locations and keep the data accessible while facing hardware or software errors

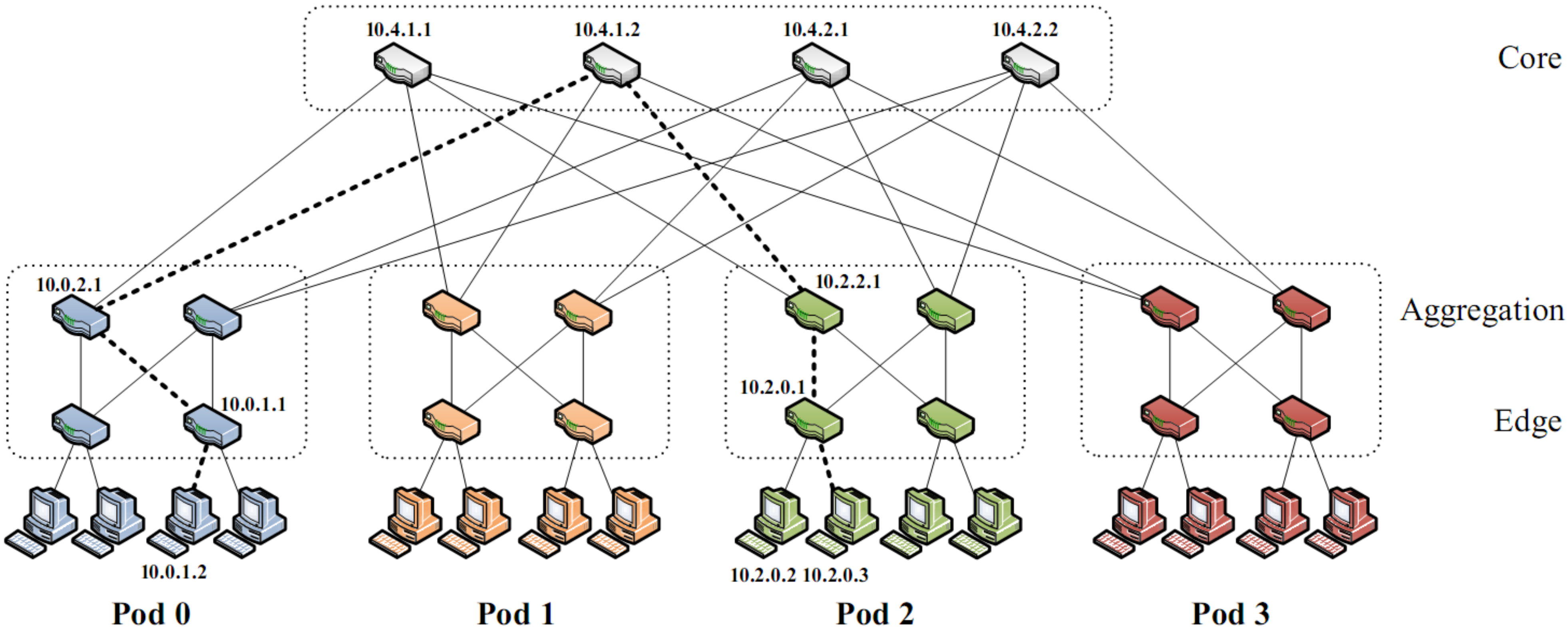
Warehouse-Scale Data-Center Design

- **The cooling system in a raised-floor data center with hot-cold air circulation supporting water heat exchange facilities**

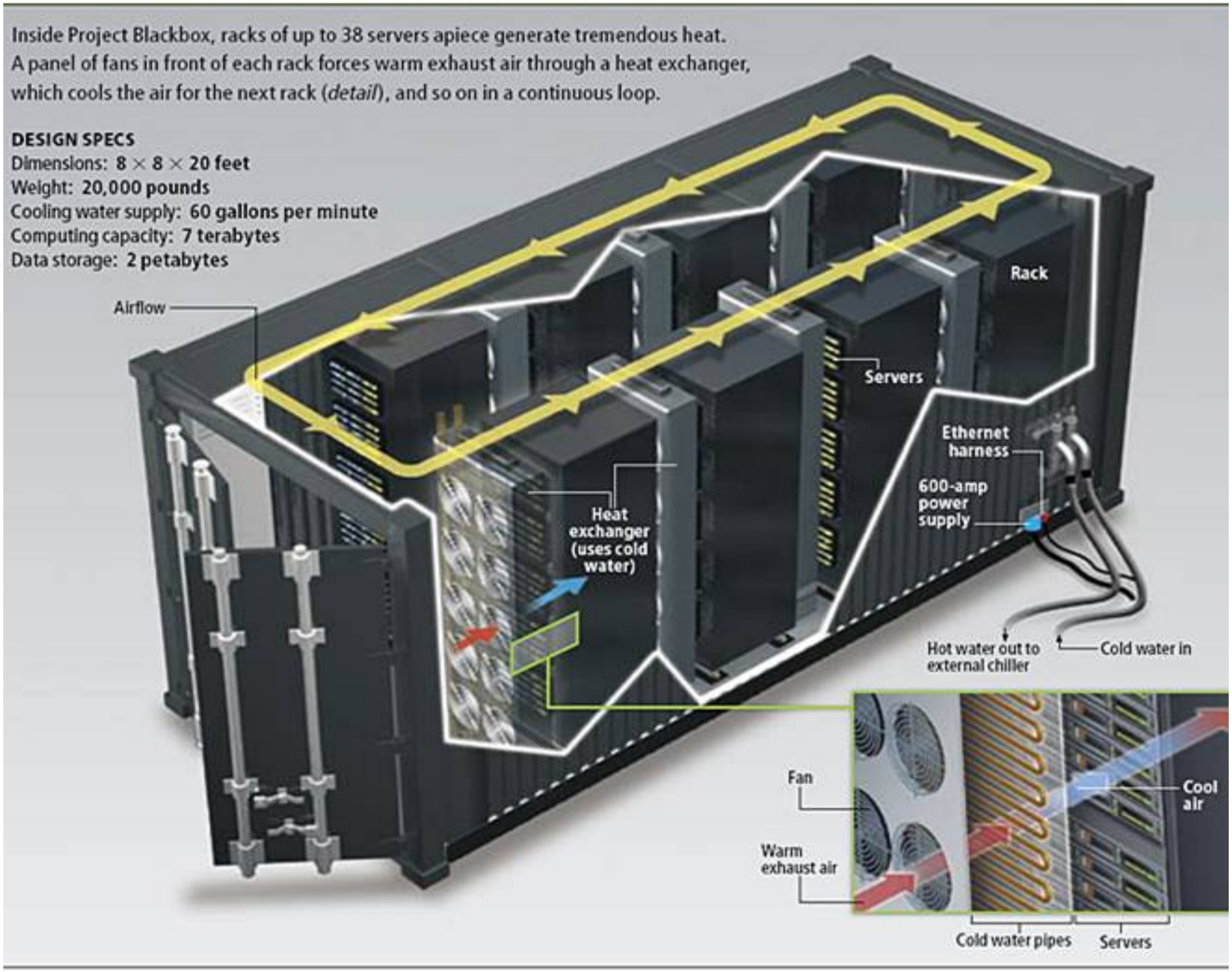


Data-Center Interconnection Networks

- **A fat-tree interconnection topology for scalable data-center construction**



Modular Data Center in Shipping Containers



Data-Center Management Issues

- Basic requirements for managing the resources of a data center:
  - **Making common users happy**
    - The data center should be designed to provide quality service to the majority of users for at least 30 years.
  - **Controlled information flow**
    - Information flow should be streamlined. Sustained services and high availability (HA) are the primary goals.
  - **Multiuser manageability**
    - The system must be managed to support all functions of a data center, including traffic flow, database updating, and server maintenance.
  - **Scalability to prepare for database growth**
    - The system should allow growth as workload increases. The storage, processing, I/O, power, and cooling subsystems should be scalable.
  - **Reliability in virtualized infrastructure**
    - Failover, fault tolerance, and VM live migration should be integrated to enable recovery of critical applications from failures or disasters.
  - **Low cost to both users and providers**
    - The cost to users and providers of the cloud system built over the data centers should be reduced, including all operational costs.
  - **Security enforcement and data protection**
    - Data privacy and security defense mechanisms must be deployed to protect the data center against network attacks and system interrupts and to maintain data integrity from user abuses or network attacks.
  - **Green information technology**
    - Saving power consumption and upgrading energy efficiency are in high demand when designing and operating current and future data centers.

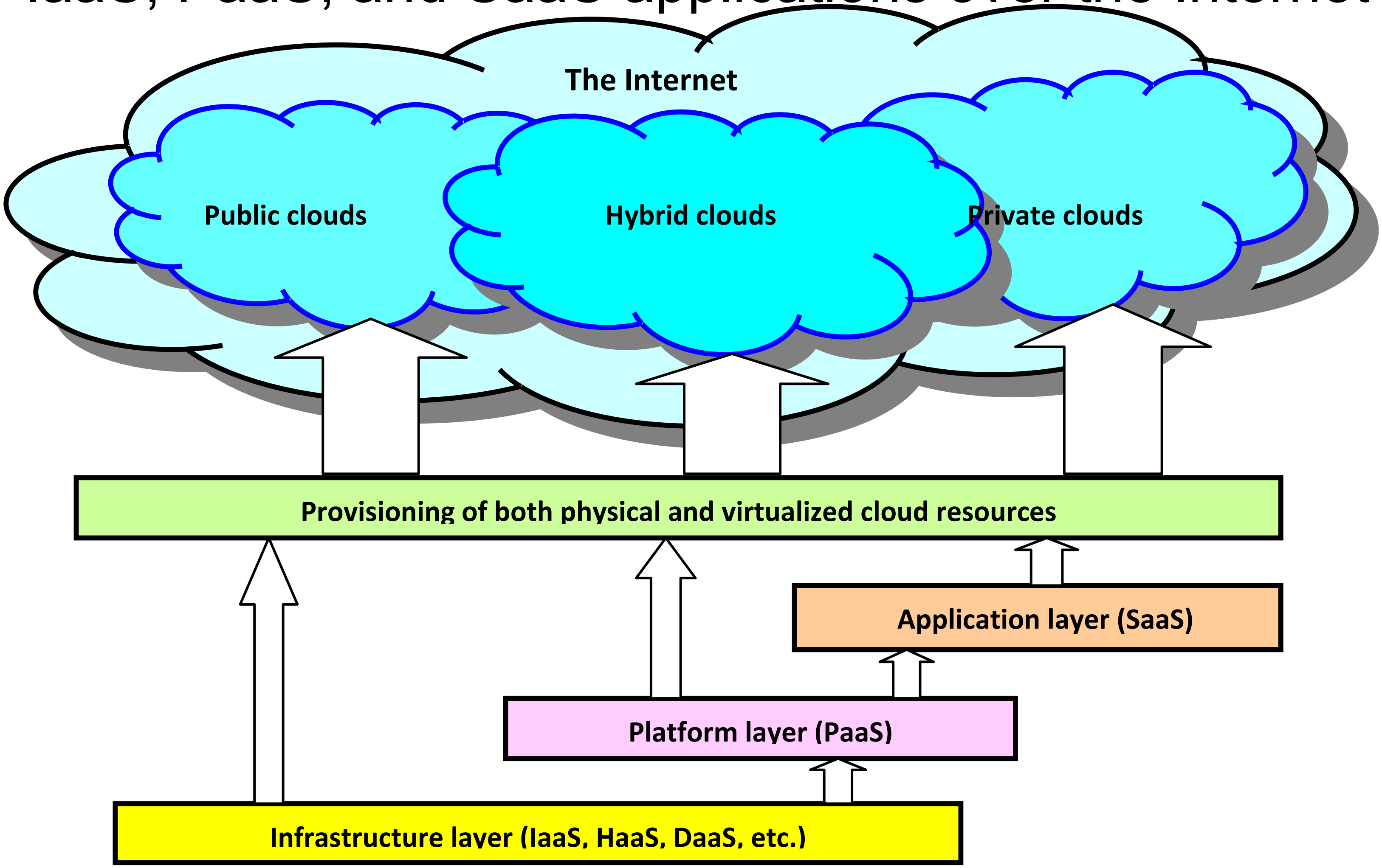
A Generic Cloud Architecture Design

- **Cloud-Enabling Technologies in Hardware, Software, and Networking**

Technology	Requirements and Benefits
Fast platform deployment	Fast, efficient, and flexible deployment of cloud resources to provide dynamic computing environment to users
Virtual clusters on demand	Virtualized cluster of VMs provisioned to satisfy user demand and virtual cluster reconfigured as workload changes
Multitenant techniques	SaaS for distributing software to a large number of users for their simultaneous use and resource sharing if so desired
Massive data processing	Internet search and Web services which often require massive data processing, especially to support personalized services
Web-scale communication	Support for e-commerce, distance education, telemedicine, social networking, digital government, and digital entertainment applications
Distributed storage	Large-scale storage of personal records and public archive information which demands distributed storage over the clouds
Licensing and billing services	License management and billing services which greatly benefit all types of cloud services in utility computing

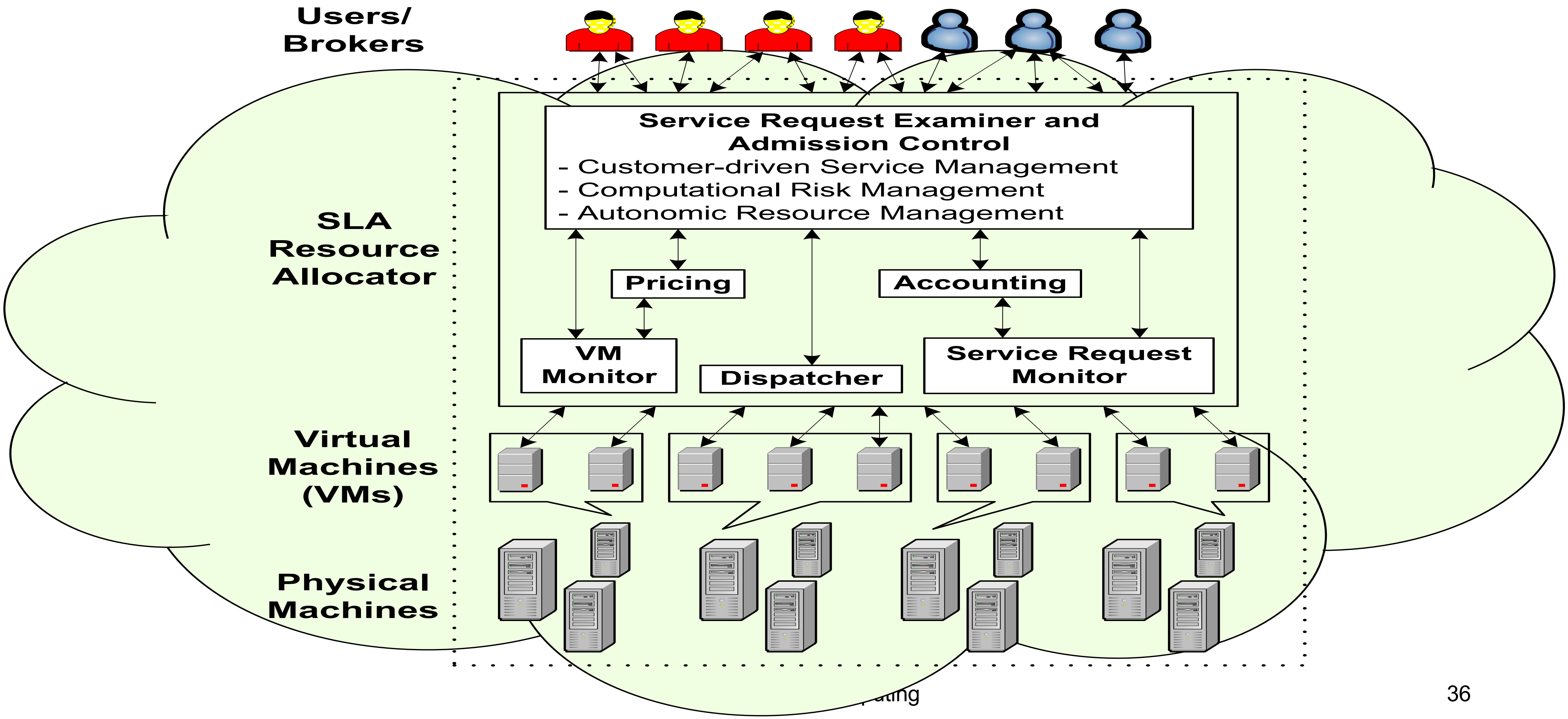
Layered Cloud Architectural Development

- Layered architectural development of the cloud platform for IaaS, PaaS, and SaaS applications over the Internet



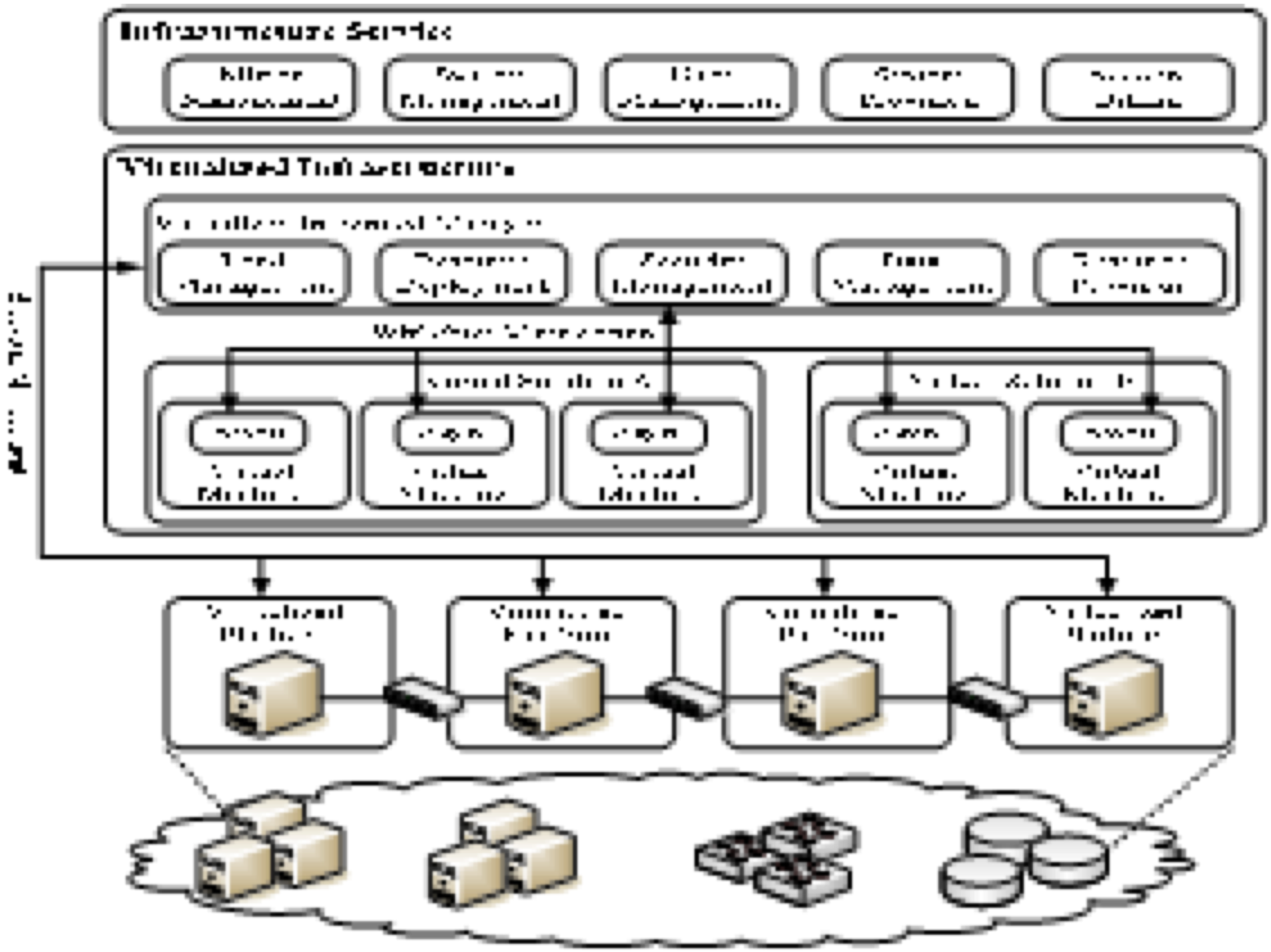
Layered Cloud Architectural Development

- Market-oriented cloud architecture to expand/shrink leasing of resources with variation in QoS/demand from users



Virtualization Support and Disaster Recovery

- Virtualized servers, storage, and network for cloud platform construction

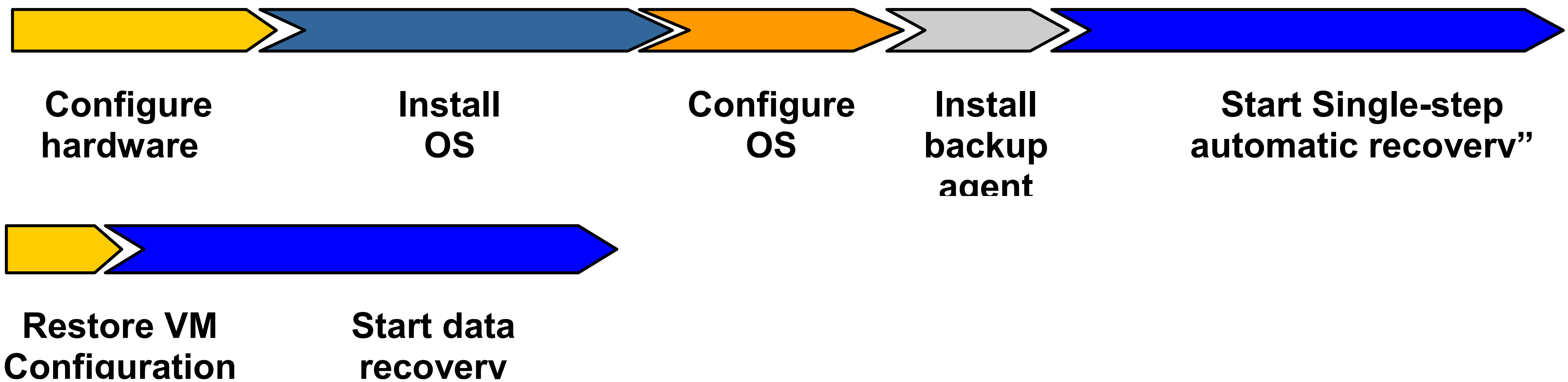


Virtualized Resources in Compute, Storage, and Network Clouds

Provider	AWS	Microsoft Azure	GAE
Compute cloud with virtual cluster of servers	x86 instruction set, Xen VMs, resource elasticity allows scalability through virtual cluster, or a third party such as RightScale must provide the cluster	Common language runtime VMs provisioned by declarative descriptions	Predefined application framework handlers written in Python, automatic scaling up and down, server failover inconsistent with the Web applications
Storage cloud with virtual storage	Models for block store (EBS) and augmented key/blob store (SimpleDB), automatic scaling varies from EBS to fully automatic (SimpleDB, S3)	SQL Data Services (restricted view of SQL Server), Azure storage service	MegaStore/BigTable
Network cloud services	Declarative IP-level topology; placement details hidden, security groups restricting communication, availability zones isolate network failure, elastic IP applied	Automatic with user's declarative descriptions or roles of app. components	Fixed topology to accommodate three-tier Web app. structure, scaling up and down is automatic and programmer-invisible

Virtualization Support and Disaster Recovery

- **Recovery overhead of a conventional disaster recovery scheme, compared with that required to recover from live migration of VMs**



Architectural Design Challenges

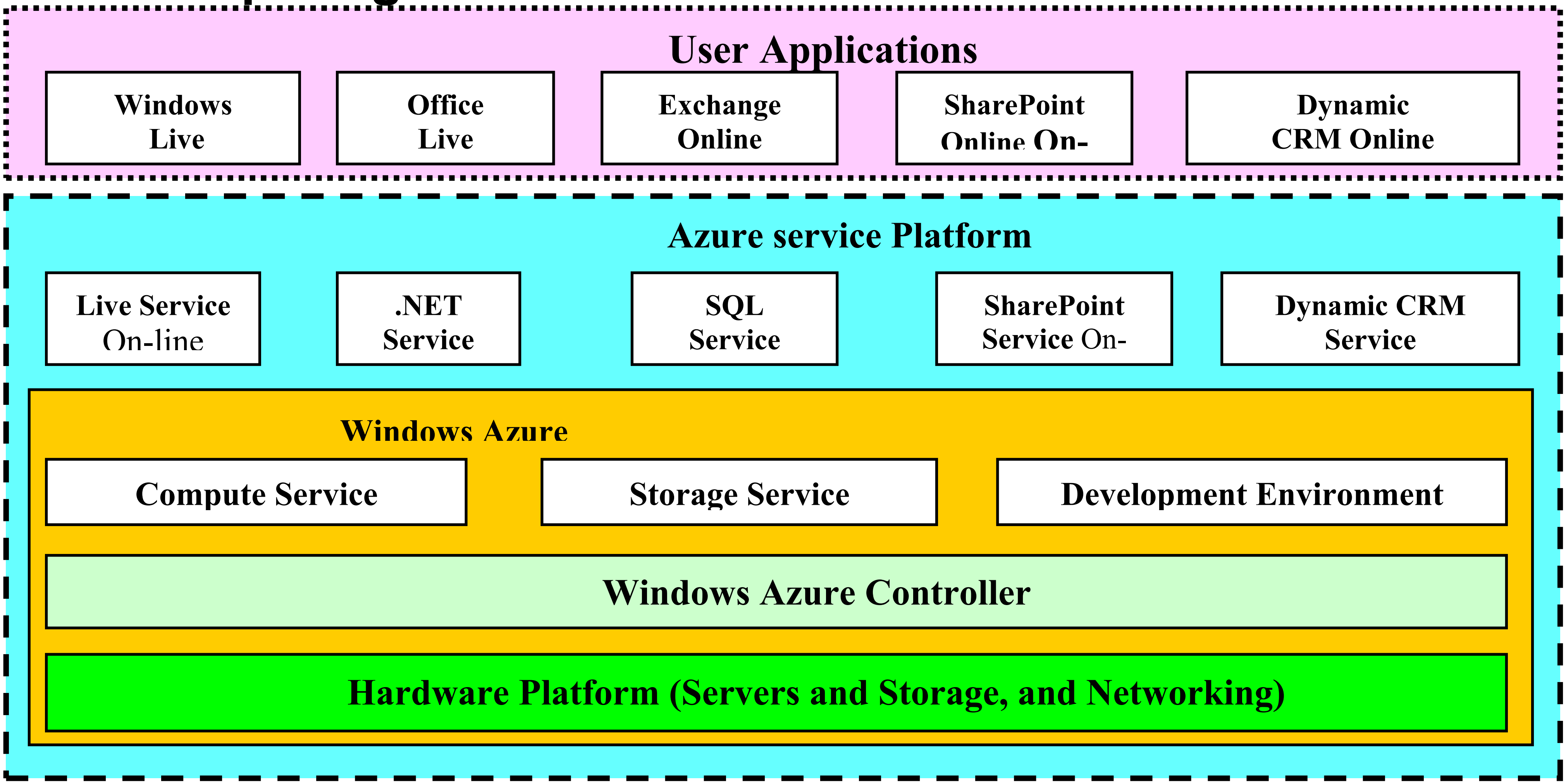
- Six open challenges in cloud architecture development
  - #1 – Service Availability and Data Lock-in Problem
  - #2 – Data Privacy and Security Concerns
  - #3 – Unpredictable Performance and Bottlenecks
  - #4 – Distributed Storage and Software Bugs
  - #5 – Cloud Scalability, Interoperability, and Standardization
  - #6 – Software Licensing and Reputation Sharing

Microsoft Windows Azure

- Launched in 2008 to meet challenges of Cloud Computing
- 3 Major Components:
  - User Applications
    - Installed in VMs
  - Azure Service Platform
    - Live service: Users can visit Microsoft Live applications and apply the data involved across multiple machines concurrently.
    - .NET service: This package supports application development on local hosts and execution on cloud machines.
    - SQL Azure: This function makes it easier for users to visit and use the relational database associated with the SQL server in the cloud.
    - SharePoint service: This provides a scalable and manageable platform for users to develop their special business applications in upgraded Web services.
    - Dynamic CRM service: This provides software developers a business platform in managing CRM applications in financing, marketing, and sales and promotions.
  - Windows Azure
    - manages all servers, storage, and network resources of the data center

Microsoft Windows Azure

- **Microsoft Windows Azure platform for cloud computing**

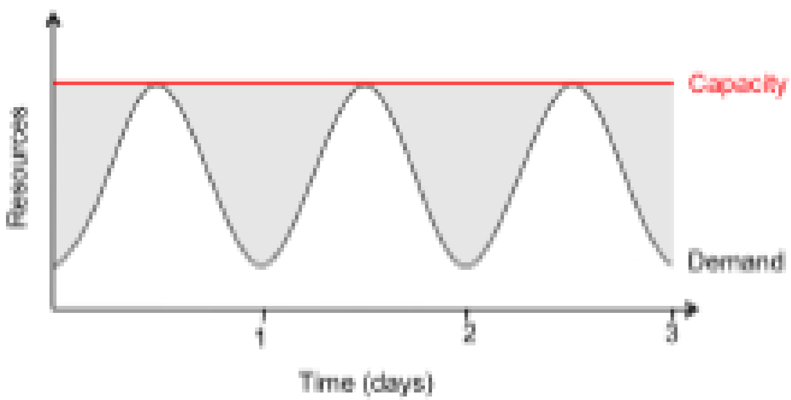


Resource Provisioning and Platform Deployment

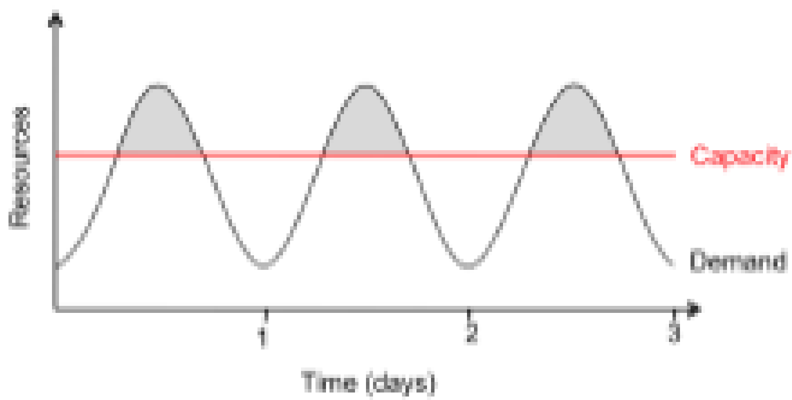
- Providers supply cloud services by signing SLAs with end users
- The SLAs must commit sufficient resources such as CPU, memory, and bandwidth that the user can use for a preset period
- Under-provisioning of resources will lead to broken SLAs and penalties
- Overprovisioning of resources will lead to resource underutilization, and consequently, a decrease in revenue for the provider
- Deploying an autonomous system to efficiently provision resources to users is a hard problem
  - The difficulty comes from the unpredictability of consumer demand, software and hardware failures, heterogeneity of services, power management, and conflicts in signed SLAs between consumers and service providers

Resource Provisioning and Platform Deployment

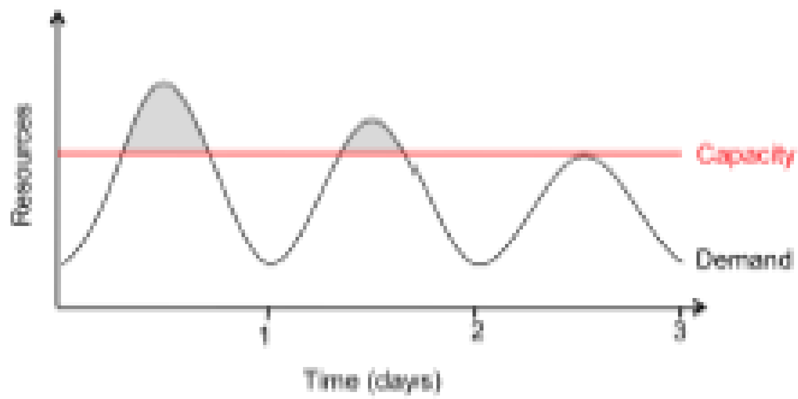
- Three cases of cloud resource provisioning without elasticity: (a) heavy waste due to overprovisioning, (b) underprovisioning and (c) under- and then overprovisioning



(a) Provisioning for peak load



(b) Underprovisioning 1



(c) Underprovisioning 2

Resource Provisioning and Platform Deployment

- **Demand-Driven Resource Provisioning:**
  - This method adds or removes computing instances based on the current utilization level of the allocated resources
  - For example: the demand-driven method automatically allocates two Xeon processors for the user application, when the user was using one Xeon processor more than 60 percent of the time for an extended period
  - In general, when a resource has surpassed a threshold for a certain amount of time, the scheme increases that resource based on demand
  - When a resource is below a threshold for a certain amount of time, that resource could be decreased accordingly
  - Amazon implements such an auto-scale feature in its EC2 platform
  - Pros: This method is easy to implement
  - Cons: The scheme does not work out right if the workload changes abruptly

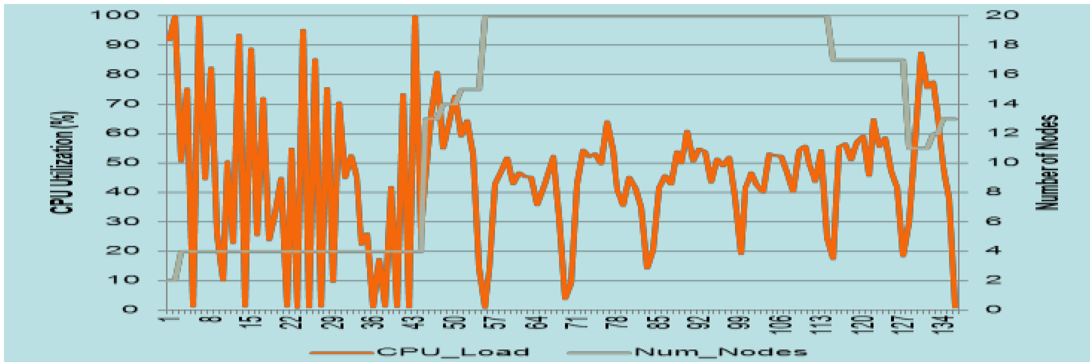
Resource Provisioning and Platform Deployment

- **Event-Driven Resource Provisioning:**
  - This scheme adds or removes machine instances based on a specific time event
  - The scheme works better for seasonal or predicted events such as Christmastime in the West and the Lunar New Year in the East
  - During these events, the number of users grows before the event period and then decreases during the event period
  - This scheme anticipates peak traffic before it happens
  - The method results in a minimal loss of QoS, if the event is predicted correctly
  - Otherwise, wasted resources are even greater due to events that do not follow a fixed pattern

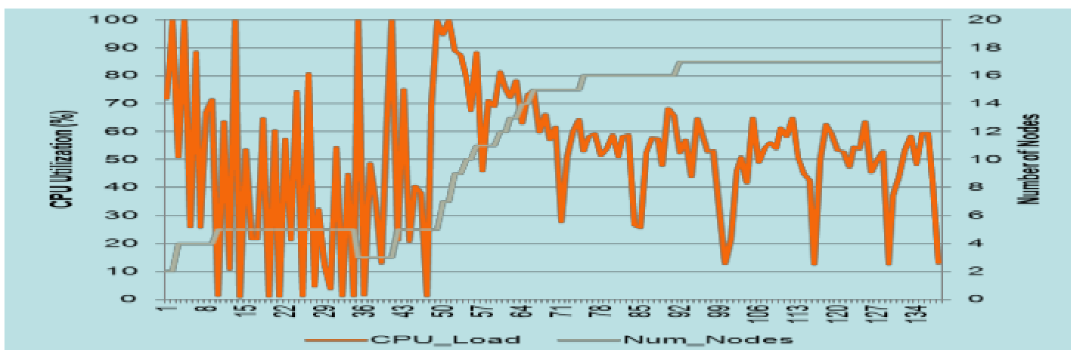
Resource Provisioning and Platform Deployment

- **Popularity-Driven Resource Provisioning:**
  - Popularity is determined from the Internet, and instances are created by popularity demand
  - The scheme anticipates increased traffic with popularity
  - The scheme has a minimal loss of QoS, if the predicted popularity is correct
  - Resources may be wasted if traffic does not occur as expected

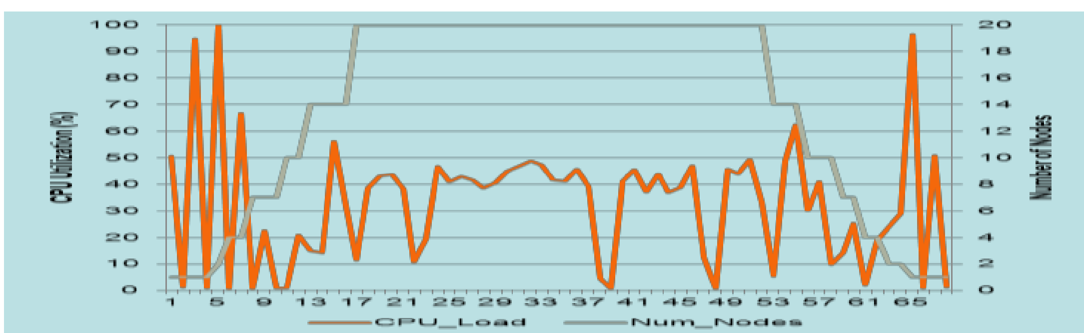
Resource Provisioning and Platform Deployment



(a) Demand-driven



(b) Event-driven



(c) Popularity-driven

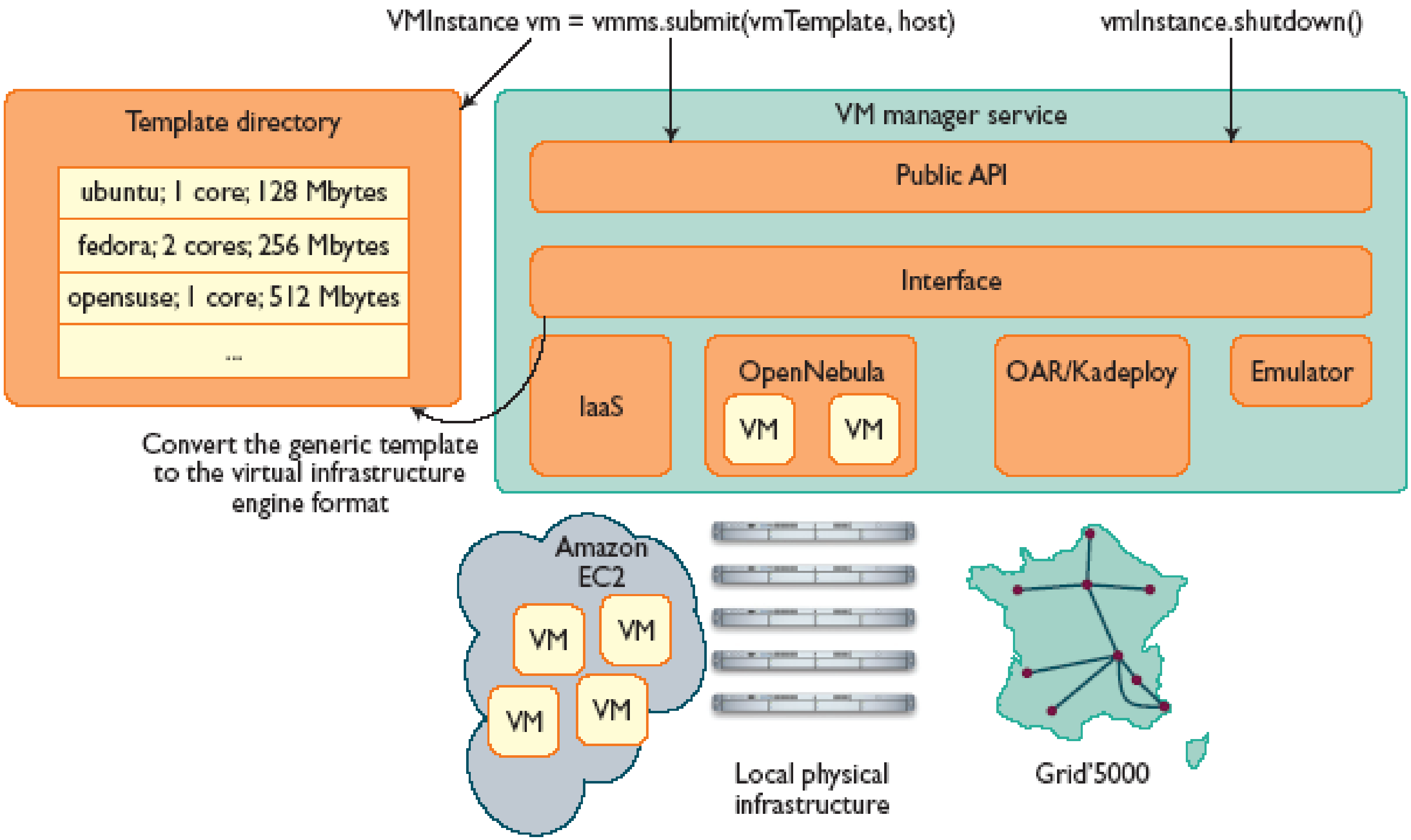
Resource Provisioning and Platform Deployment

- Storage Services in Three Cloud Computing Systems
  - The data storage layer is built on top of the physical or virtual servers
  - As the cloud computing applications often provide service to users, it is unavoidable that the data is stored in the clusters of the cloud provider

Storage System	Features
GFS: Google File System	Very large sustainable reading and writing bandwidth, mostly continuous accessing instead of random accessing. The programming interface is similar to that of the POSIX file system accessing interface.
HDFS: Hadoop Distributed File System	The open source clone of GFS. Written in Java. The programming interfaces are similar to POSIX but not identical.
Amazon S3 and EBS	S3 is used for retrieving and storing data from/to remote servers. EBS is built on top of S3 for using virtual disks in running EC2 instances.

Virtual Machine Creation and Management

- Interactions among VM managers for cloud creation and management
- The manager provides a public API for users to submit and control the VMs



Cloud Security and Trust Management

- Here are some cloud components that demand special security protection:
  - Protection of servers from malicious software attacks such as worms, viruses, and malware
  - Protection of hypervisors or VM monitors from software-based attacks and vulnerabilities
  - Protection of VMs and monitors from service disruption and DoS attacks
  - Protection of data and information from theft, corruption, and natural disasters
  - Providing authenticated and authorized access to critical data and services

Cloud Security and Trust Management

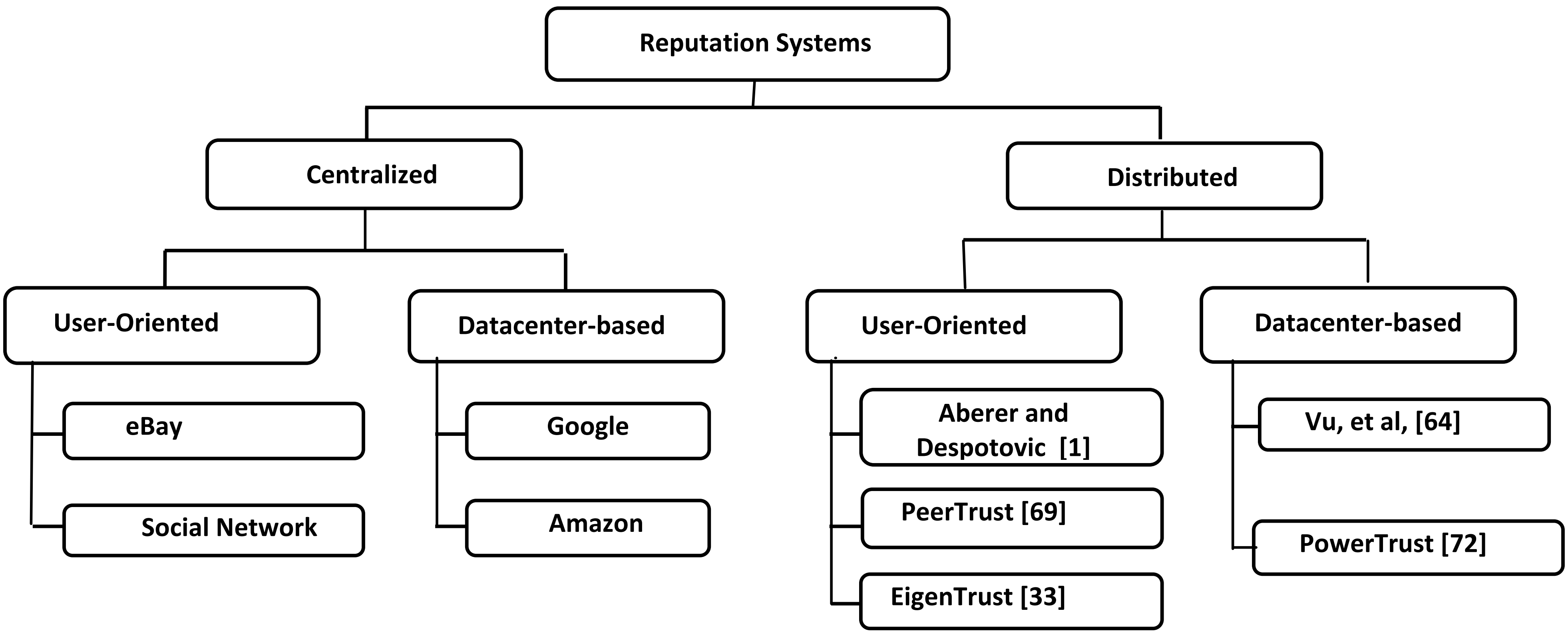
- **Security Challenges in VMs:**
  - Hypervisor malware, guest hopping and hijacking, or VM rootkits
  - Man-in-the-middle attack for VM migrations
  - Passive attacks steal sensitive data or passwords
  - Active attacks may manipulate kernel data structures which will cause major damage to cloud servers
  - Program shepherding can be applied to control and verify code execution
  - Other defense technologies include using the RIO dynamic optimization infrastructure, or VMware's vSafe and vShield tools, security compliance for hypervisors, and Intel vPro technology
  - Others apply a hardened OS environment or use isolated execution and sandboxing

Cloud Security Defense Strategies

- **Cloud Defense Methods:**
  - Virtualization enhances cloud security
  - However VMs add an additional layer of software that could become a single point of failure
  - With virtualization, a single physical machine can be divided or partitioned into multiple VMs
    - This provides each VM with better security isolation and each partition is protected from DoS attacks by other partitions
    - Security attacks in one VM are isolated and contained from affecting the other VMs

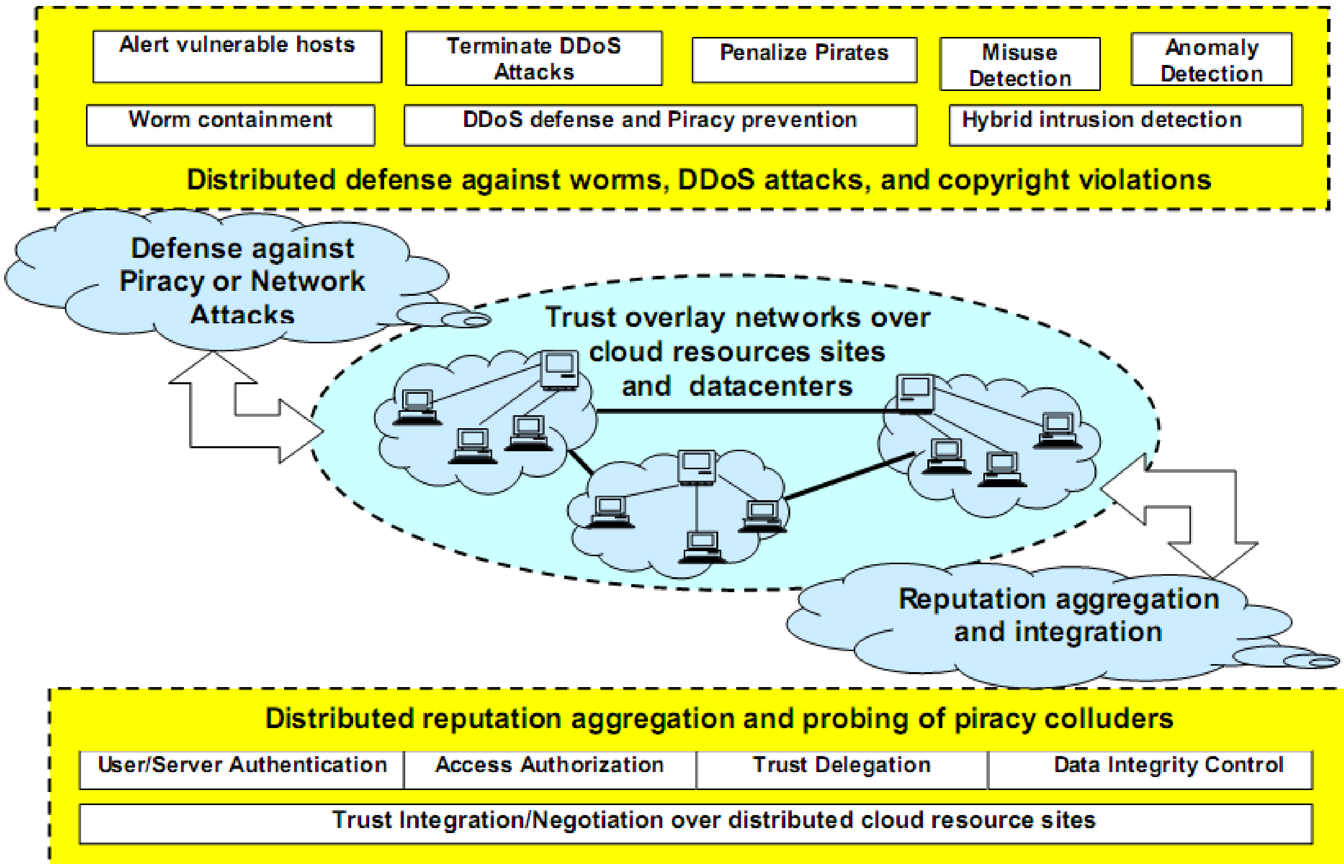
Reputation-Guided Protection of Data Centers

- Design options of reputation systems for social networks and cloud platforms



Reputation-Guided Protection of Data Centers

- DHT-based trust overlay networks built over cloud resources provisioned from multiple data centers for trust management and distributed security enforcement



Questions

?