Kai Hwang : *Cloud Computing for Machine Learning and Cognitive Applications*

The MIT Press, Cambridge, MA. 2017

Chapter 4: Cloud Architecture, Services, and AWS Project

(100 slides for 6-hour lectures)

Some Representative Public Clouds

Table 4.1
Five public cloud platforms and their service offerings (2016)

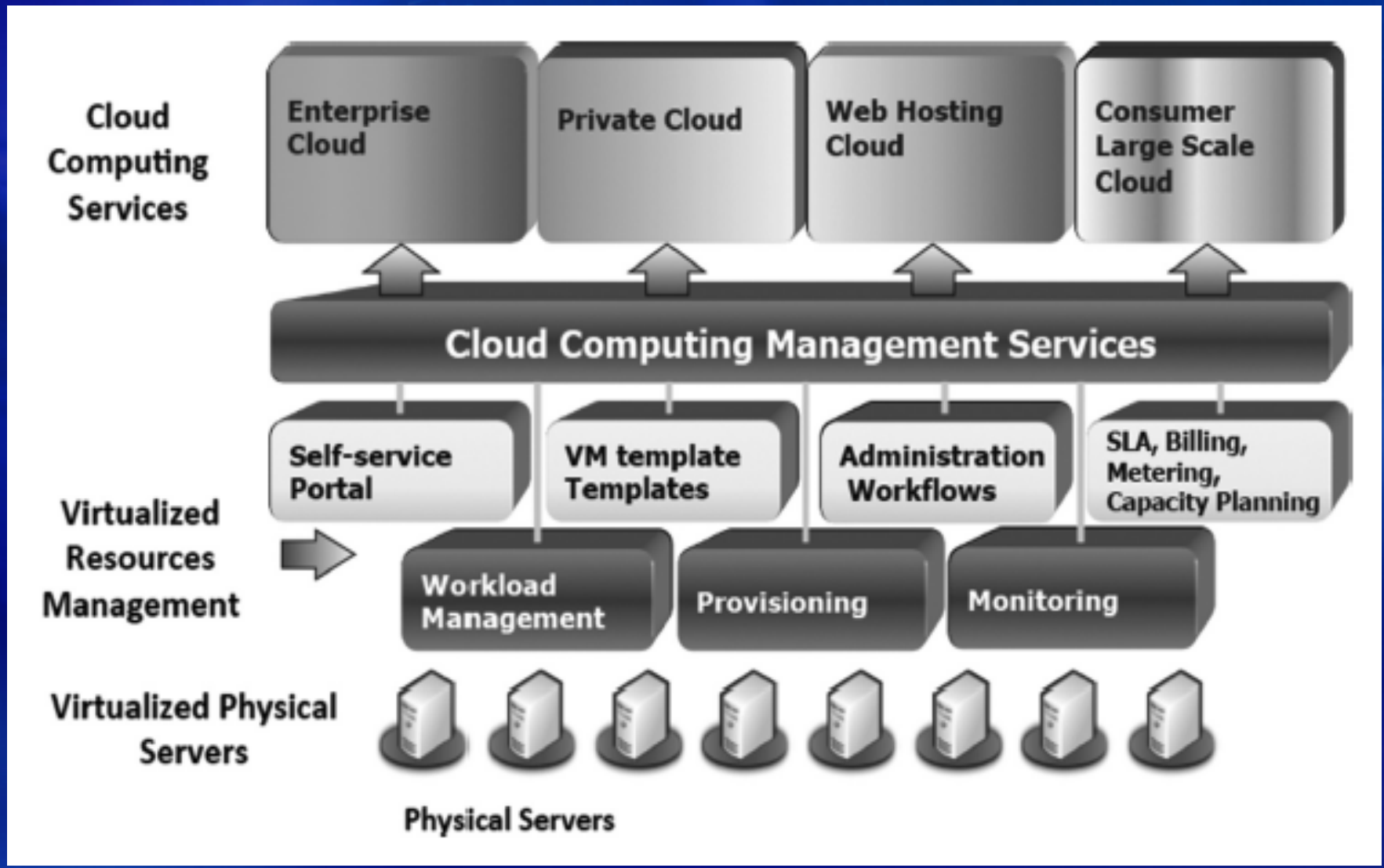
Public Clouds	Platform Model(s)	Typical Service Offerings	Website and Coverage in Book
Amazon Web Service (AWS)	IaaS, PaaS	EC2, S3, SQS, EMR, VPC, EBS, SNS, CloudFront, etc.	http://aws.amazon.com/, Section 4.3
Google App Engine	PaaS, SaaS	Gmail, Docs, GFS, BigTable, Chubby	https://developer.google.com, Section 4.4.1~2
Microsoft Azure	PaaS, SaaS	Live, SQL, Office 365, Dynamic CRM	http://www.windowsazure.com, Section 4.4.3
IBM SmartCloud	PaaS, SaaS, IaaS	Compute, Storage, Backup, Networking, Virtualization	http:/www.ibm.com/cloud-computing, Section 4.5.2
SalesForce Clouds	SaaS, PaaS	CRM, Sales, Marketing, Apex, Visual force	https://salesforce.com, Section 4.5.1

Enabling Technologies for The Clouds

Table 4.3 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

A Generic Architecture of Cloud Systems



What is a computing cluster?

- A computing cluster consists of a collection of interconnected stand-alone/complete computers, which can cooperatively work together as a single, integrated computing resource. Cluster explores parallelism at job level and distributed computing with higher availability.
- An ideal cluster:
  - Merging multiple system images to a SSI (single-system image) at certain functional levels
  - Low latency communication protocols applied
  - Highly available than an SMP under the control of a single operating system

Multi-Computer Cluster Components

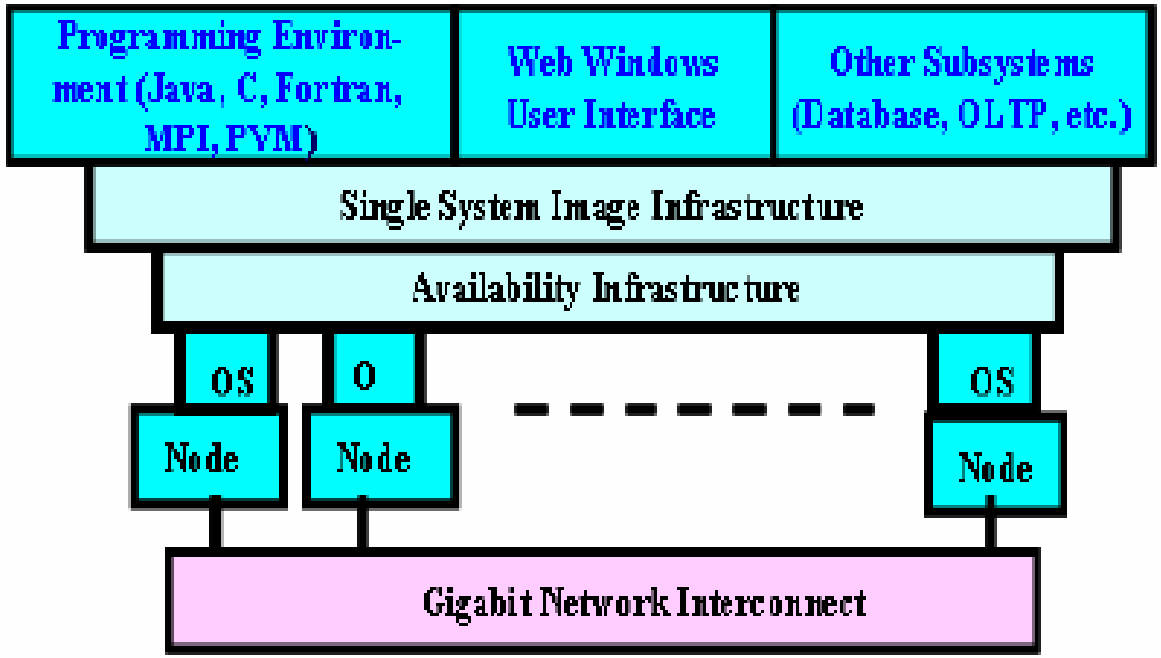


Figure 1.16 The architecture of a working cluster with full hardware, software, and middleware support for availability and single system image.

Operational Benefits of Clustering

- **High Availability (HA):** Cluster offers inherent high system availability due to the redundancy of hardware, operating systems, and applications
- **Fault Tolerance:** Cluster has some degree of redundancy in most system components including both hardware and software modules
- **OS and application reliability:** Run multiple copies of the OS and applications, and through this redundancy
- **Scalability:** Adding servers to a cluster or adding more clusters to a network as the application need arises
- **High Performance:** Running cluster enabled programs to yield higher throughput

Issues in Cluster Design

- Size scalability (physical & application)
- Enhanced availability (failure management)
- Single System Image (Middleware, OS extensions)
- Fast communication (networks & protocols)
- Load balancing (CPU, Net, Memory, Disk)
- Security and encryption (clusters and Grids)
- Distributed environment (User friendly)
- Manageability (Jobs and resources)
- Programmability (simple API required)
- Applicability (cluster- and grid-awareness)

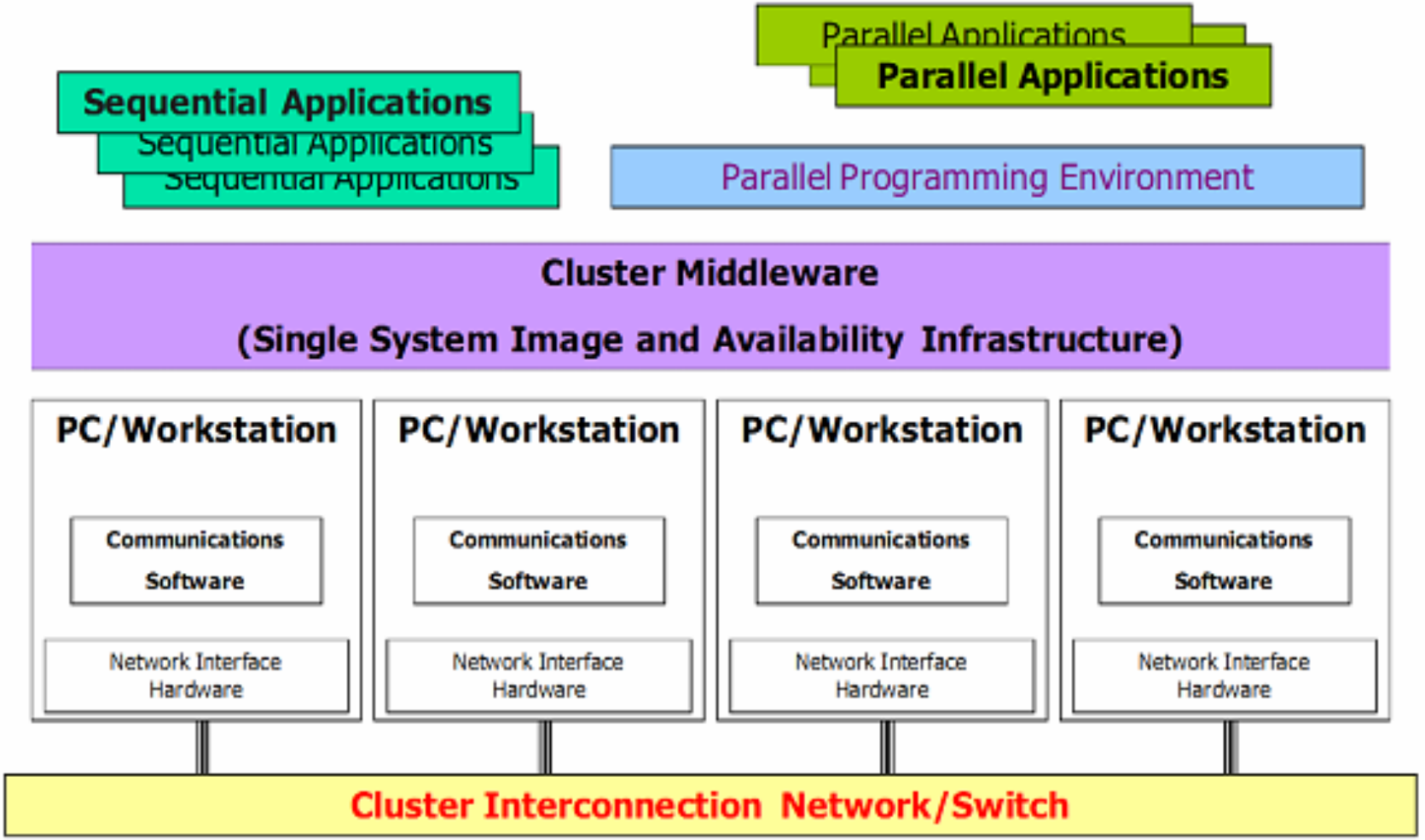


Figure 2.7 The architecture of a computer cluster with hardware, software, and middleware support for availability and single system image.
(Courtesy of M. Baker and R. Buyya, reprint with permission [2])

Three Classes of Commercial Interconnects for Building Clusters

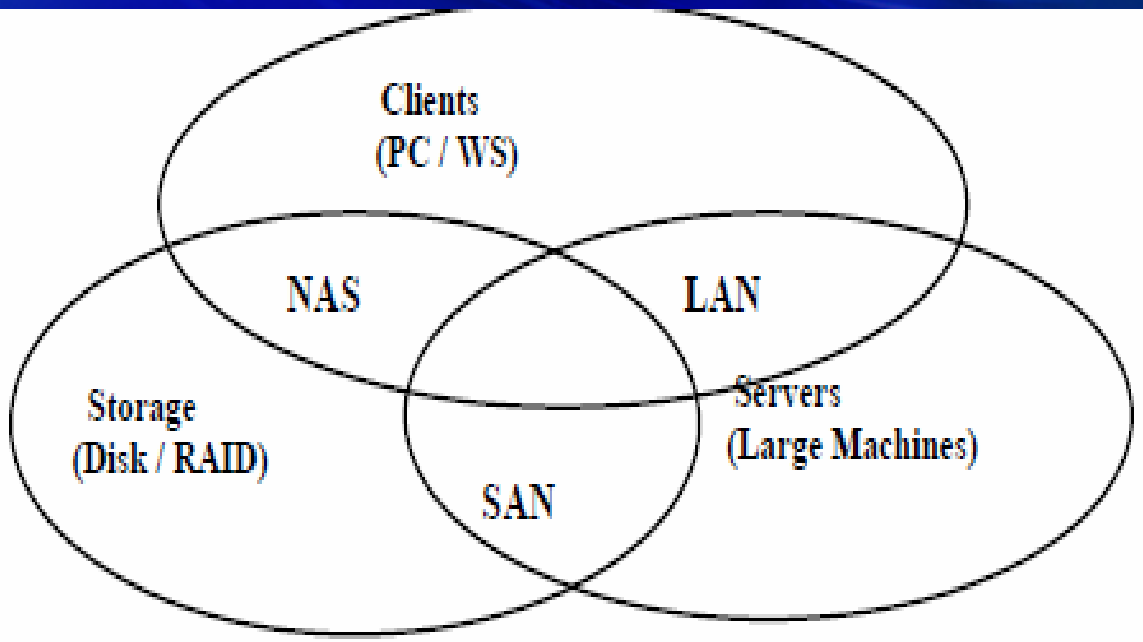


Figure 2.9 Three interconnection networks connecting servers, client hosts and storage devices: the LAN between client hosts and servers. The SAN between servers and disk arrays, and the NAS between clients and storage system.

What is Single System Image (SSI) ?

- A single system image is the illusion, created by software or hardware, that presents a collection of resources as an integrated powerful resource
- SSI makes the cluster appear like a single machine to the user, applications, and network
- A cluster with multiple system images is nothing but a collection of independent computers (Distributed systems in general)

Desired SSI Services

- **Single entry point**
  - telnet cluster.usc.edu
  - telnet node1.cluster.usc.edu
- **Single file hierarchy:** xFS, AFS, Solaris MC Proxy
- **Single control point:** Management from single GUI
- **Single virtual networking:** over multiple physical networks
- **Single memory space:** Network RAM / DSM
- **Single job management:** GlUnix, Codine, LSF, etc.
- **Single user interface:** API Tools as universal as possible

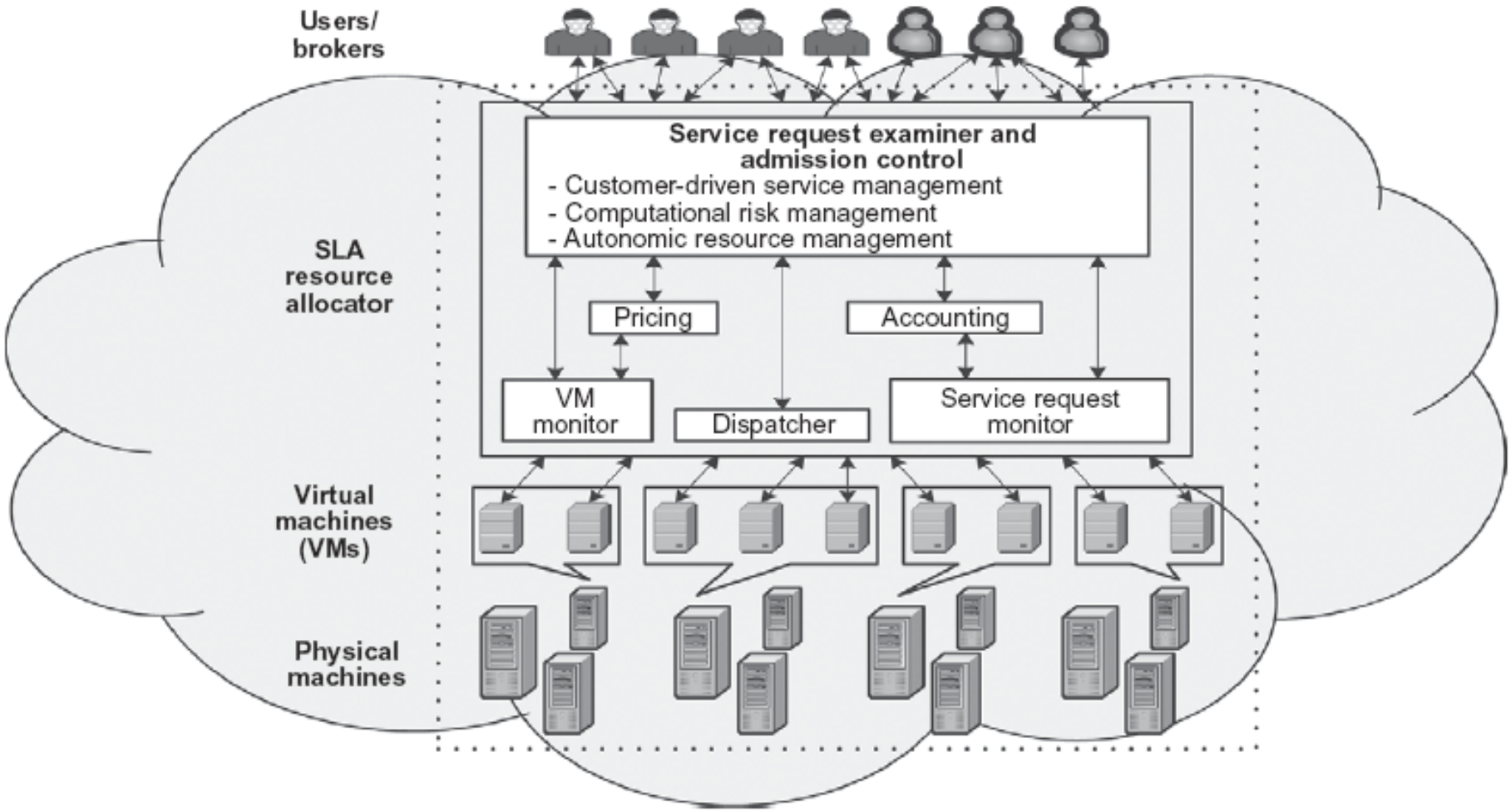


FIGURE 4.16

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

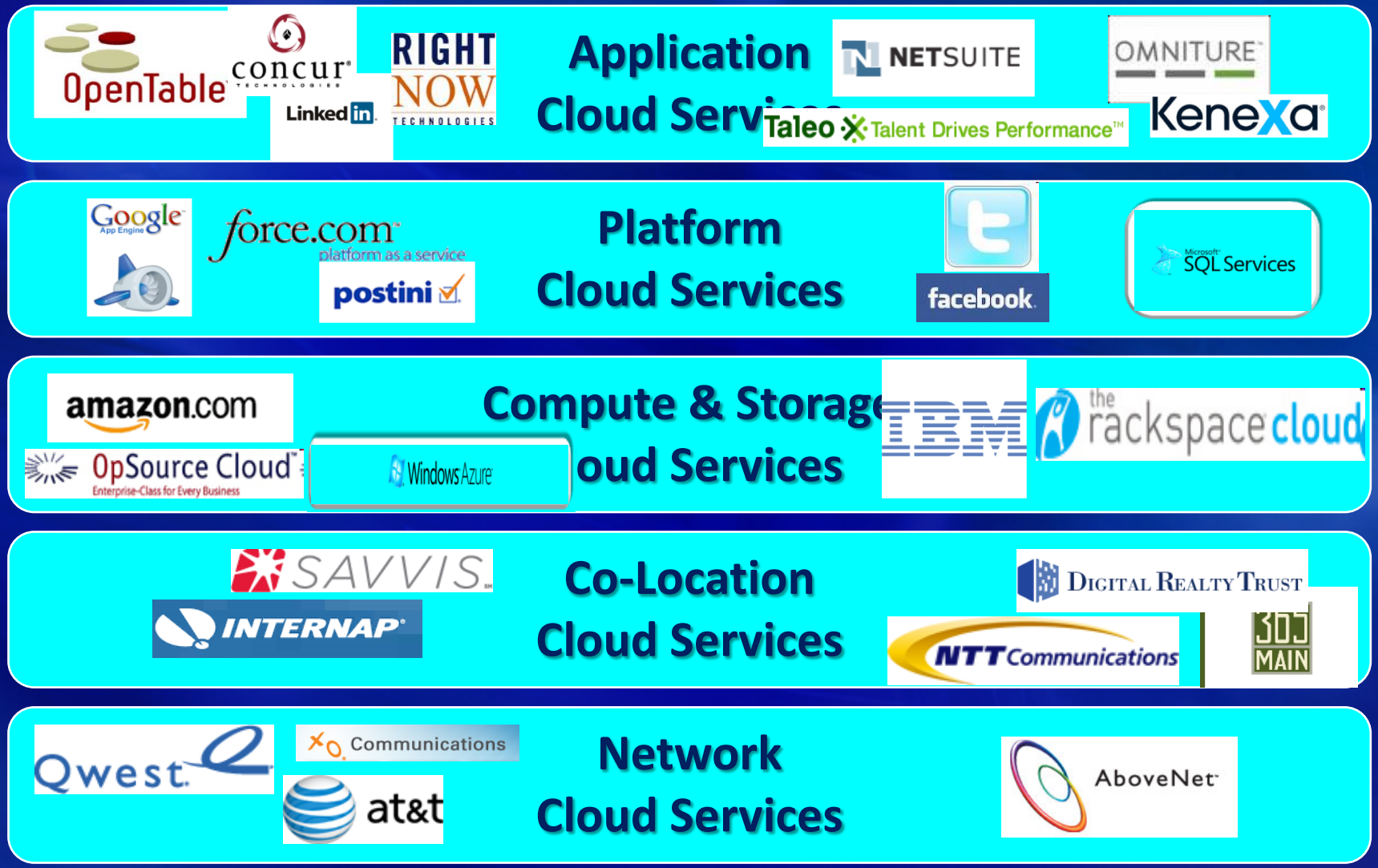
(*Courtesy of Raj Buyya, et al. [11]*)

Cloud Categories Based on Applications

Table 4.2
Five cloud service categories and their representative providers

Cloud Categories	Cloud Service Providers
Application Clouds	OpenTable, Kenexa, NetSuite, RightNow, WebEx, Blackbaud, Concur Cloud, Telco, Omniture, Vocus, Microsoft OWA (Office 365), Google Gmail, Yahoo!, Hotmail
Platform Clouds	Force.com, Google App Engine, Facebook, IBM Blue Cloud, Postini, SQL Server, Twitter, Microsoft Azure, SGI Cyclone, Amazon EMR
Compute and Storage Clouds	Amazon AWS, Rackspace, OpSource, GoGrid, MeePo, FlexiScale, HP Cloud, Banknorth, VMware, XenEnterprise, iCloud
Colocation Clouds	Savvis, Internap, Digital Realty, Trusted Advisor, 365 Main
Network Clouds	AboveNet, AT&T, Qwest, NTT Communications

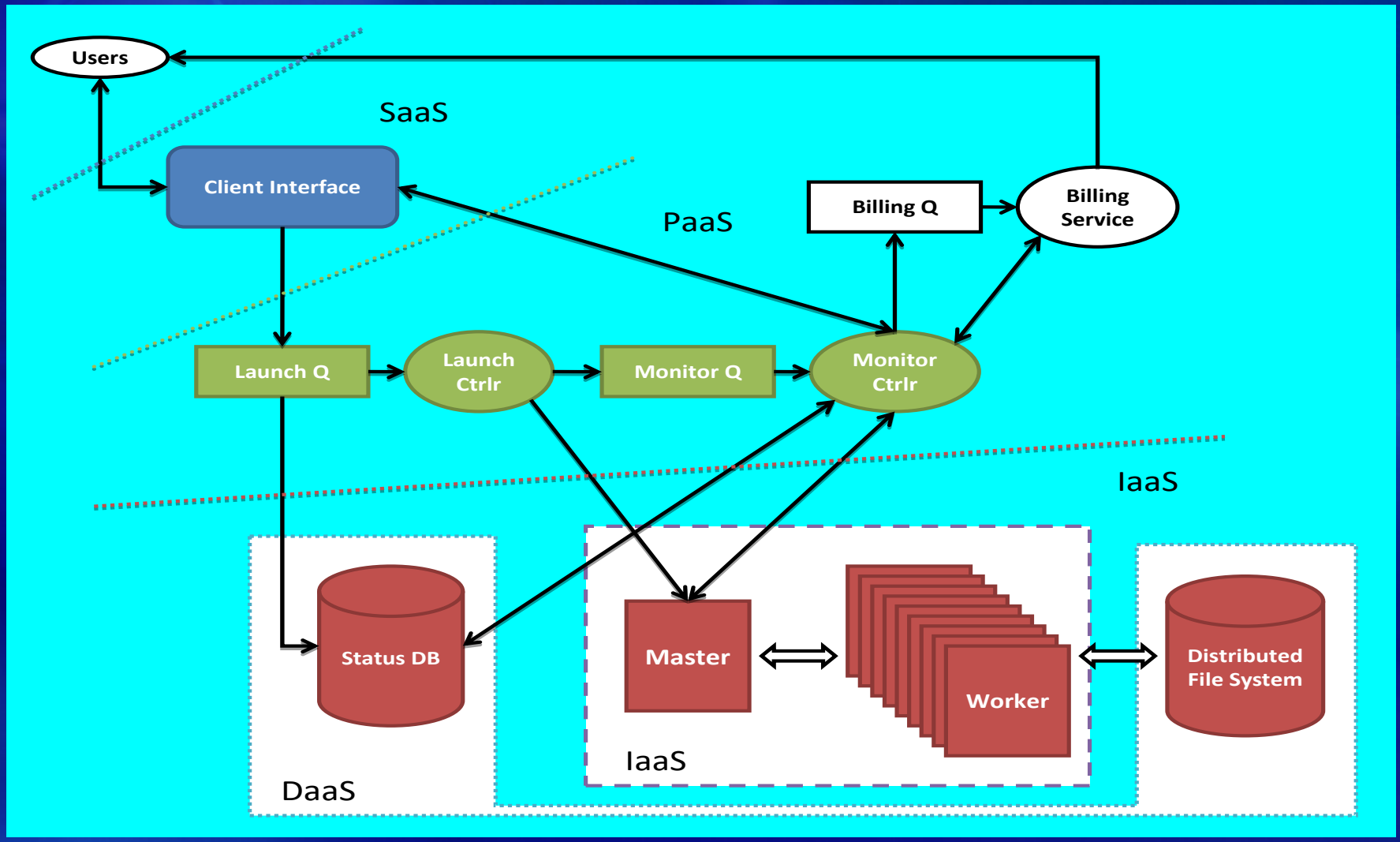
Today's Cloud Services Stack



(Courtesy of T. Chou, 2010)

Extended Cloud Service Models

DaaS, HaaS, SecaaS, NaaS, CoLaaS, etc.



The Cloud is Built on Virtualized Datacenters

Range in size from "edge" facilities to megascale (100K to 1M servers)

Economics of scale

- Approximate costs for a small size center (1K servers) and a larger, 400K server center.

Technology	Cost in small-sized Data Center	Cost in Large Data Center	Ratio
Network	$95 per Mbps/ Month	$13 per Mbps/ month	7.1
Storage	$2.20 per GB/ Month	$0.40 per GB/ month	5.7
Administra-tion	~140 servers/ Administrator	>1000 Servers/ Administrator	7.1

This data center is
11.5 times
the size of a football field

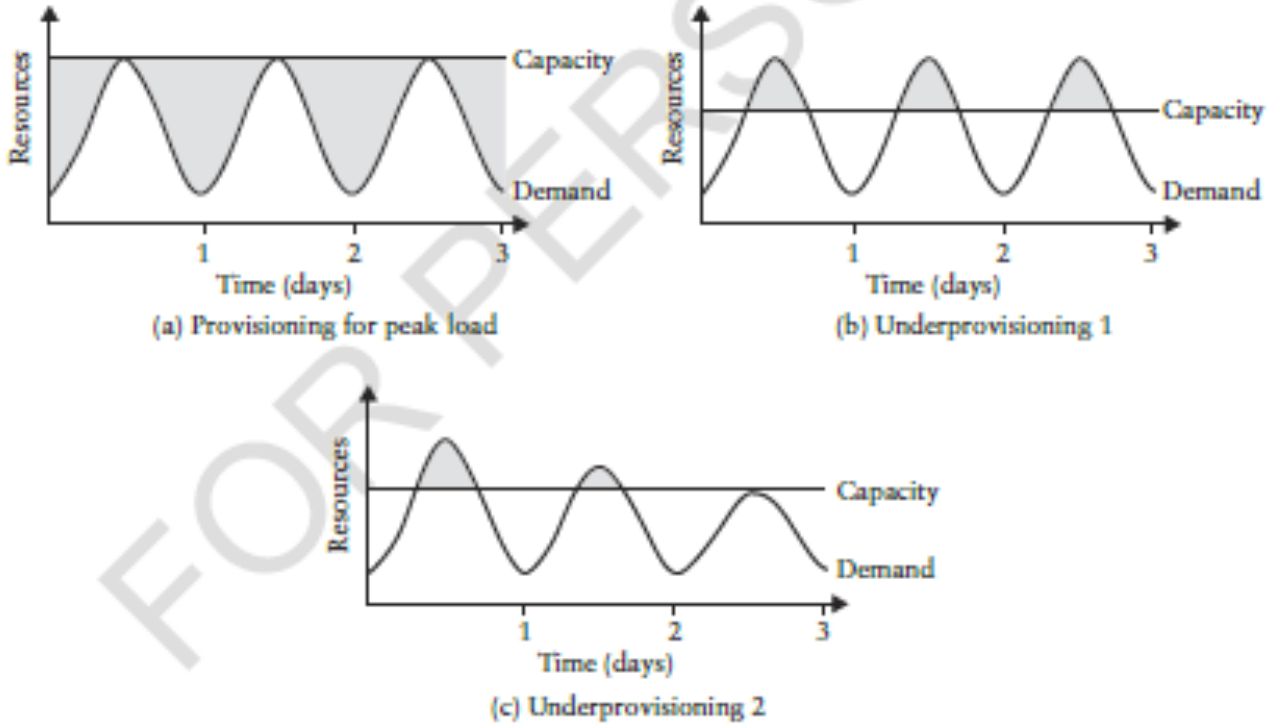


Figure 4.7
Three cloud resource provisioning cases: (a) heavy waste due to over-provisioning of resources, (b) under-resources provisioning, and (c) under-then over-provisioning.

Table 4.5
Open source virtual infrastructure managers and cloud operating systems

Manager/OS, Platforms, License	Resources Being Virtualized, Web Link	Client API, Language	Hypervisors Used	Public Cloud Interface	Special Features
Nimbus, Linux, Apache v2	VM creation, virtual cluster, http://www.nimbusproject.org/	EC2 WS, WSRF, CLI	Xen, KVM	EC2	Virtual networks
Eucalyptus, Linux, BSD	Virtual networking (Example 3.23) http://www.eucalyptus.com/	EC2 WS, CLI	Xen, KVM	EC2	Virtual networks
OpenNebula, Linux, Apache v2	Managing VM, host, virtual network, scheduling tools http://www.opennebula.org/	XML-RPC, CLI, Java	Xen, KVM	EC2, ElasticHosts	Virtual networks, Dynamic provisioning
vSphere/6, Linux, Windows, Proprietary	Virtualizing OS for data centers http://www.vmware.com/products/vsphera/ (Example 4.7)	CLI, GUI, Portal, WS	VMware ESX, ESXi	VMware vCloud partners	Data protection, vStorage, VMFS, high availability

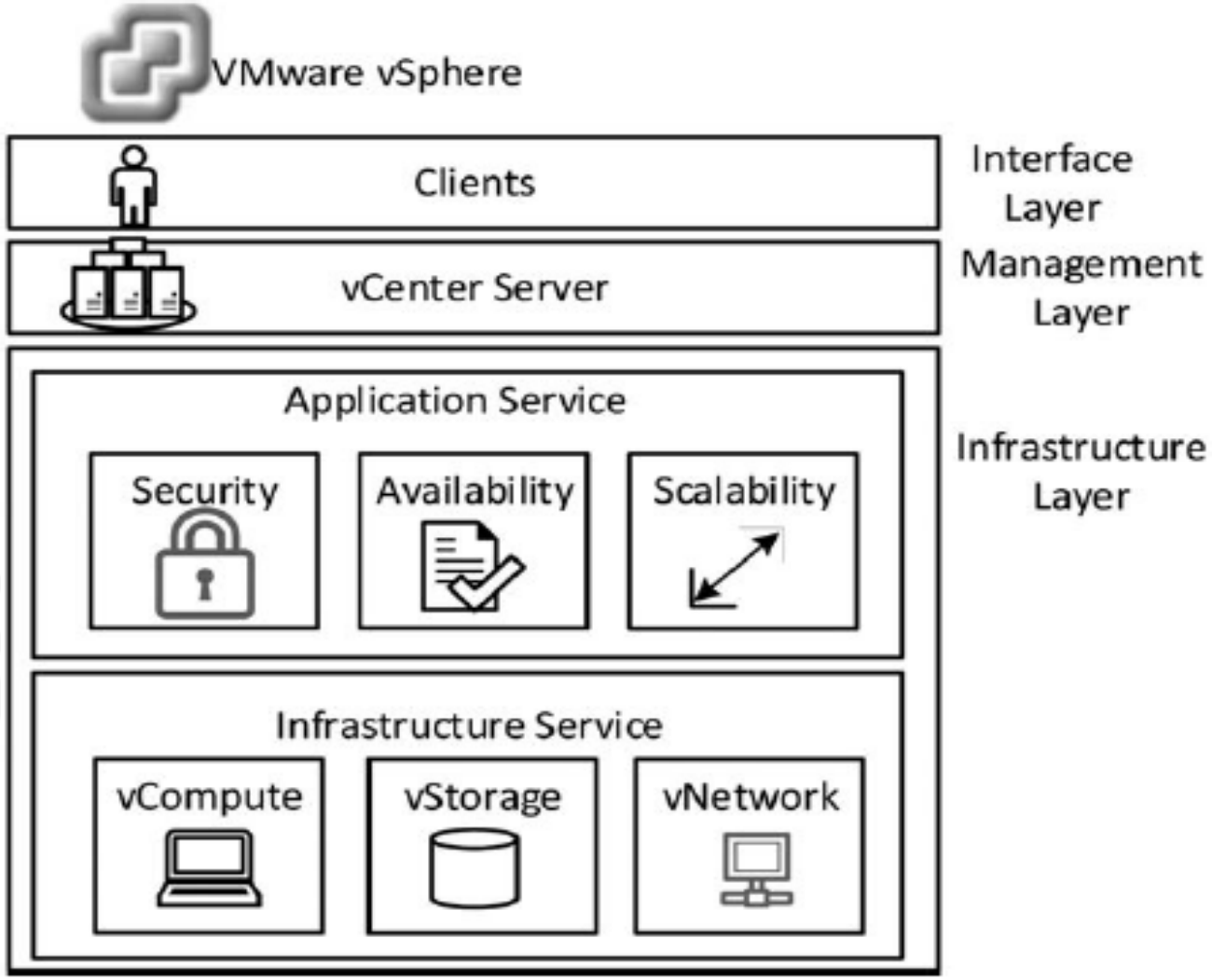


Figure 4.12
Functional modules in VMware vSphere 6 operating system for private/hybrid clouds.

Table 4.6
Comparison of three cloud platform architectures

Cloud System Features	Amazon Web Service (AWS): Public Cloud	OpenStack Systems: Private Cloud	VMWare Systems: Hybrid Cloud
Service Model(s)	IaaS, PaaS	IaaS	IaaS, PaaS
Developer/Provider and Design	Amazon (Sec.4.3)	Rackspace/NASA and Apache (Section 2.3.4)	VMWare (Section 2.3.5) Proprietary
Architecture Packages and Scale	Data centers distributed as availability zones in many global regions (Figure 2.14)	Small cloud at owner sites, licensed thru Apache (Figure 2.15)	Private clouds interacting with public clouds (Figure 2.18)
Cloud OS/ Software Support	Supporting both Linux and Windows machine instances with autoscaling and billing	Open source, extending from Eucalyptus and OpenNebula	vSphere and vCenter, supporting x-86 servers with NSX and vSAN
User Spectrum	General public: enterprises and individual users	Research centers or small businesses	Enterprises and large organizations

Some IaaS Offerings from Public Clouds

Table 4.7
Public IaaS clouds and their VM instance configurations (August 2015)

Cloud Name	Virtual Machine Instance Configurations	API and Access Tools
Amazon EC2	Each instance has 1–20 EC2 processors, 1.7–15 GB memory, and 160 TB storage	CLI or Web Service (WS) porta
GoGrid	Each instance has 1–6 CPUs, 0.5–8 GB memory, and 30–480 GB storage	REST, Java, PHP, Python, Ruby
Rackspace Cloud	Each instance has a 4-core CPU, 0.25–16 GB memory, and 10–620 GB storage	REST, Python, PHP, Java, C#, .I
FlexiScale in UK	Each instance has 1–4 CPUs, 0.5–16 GB memory, and 20–270 GB storage	Web console

The Amazon Web Service (AWS):

The most popular public cloud in use today

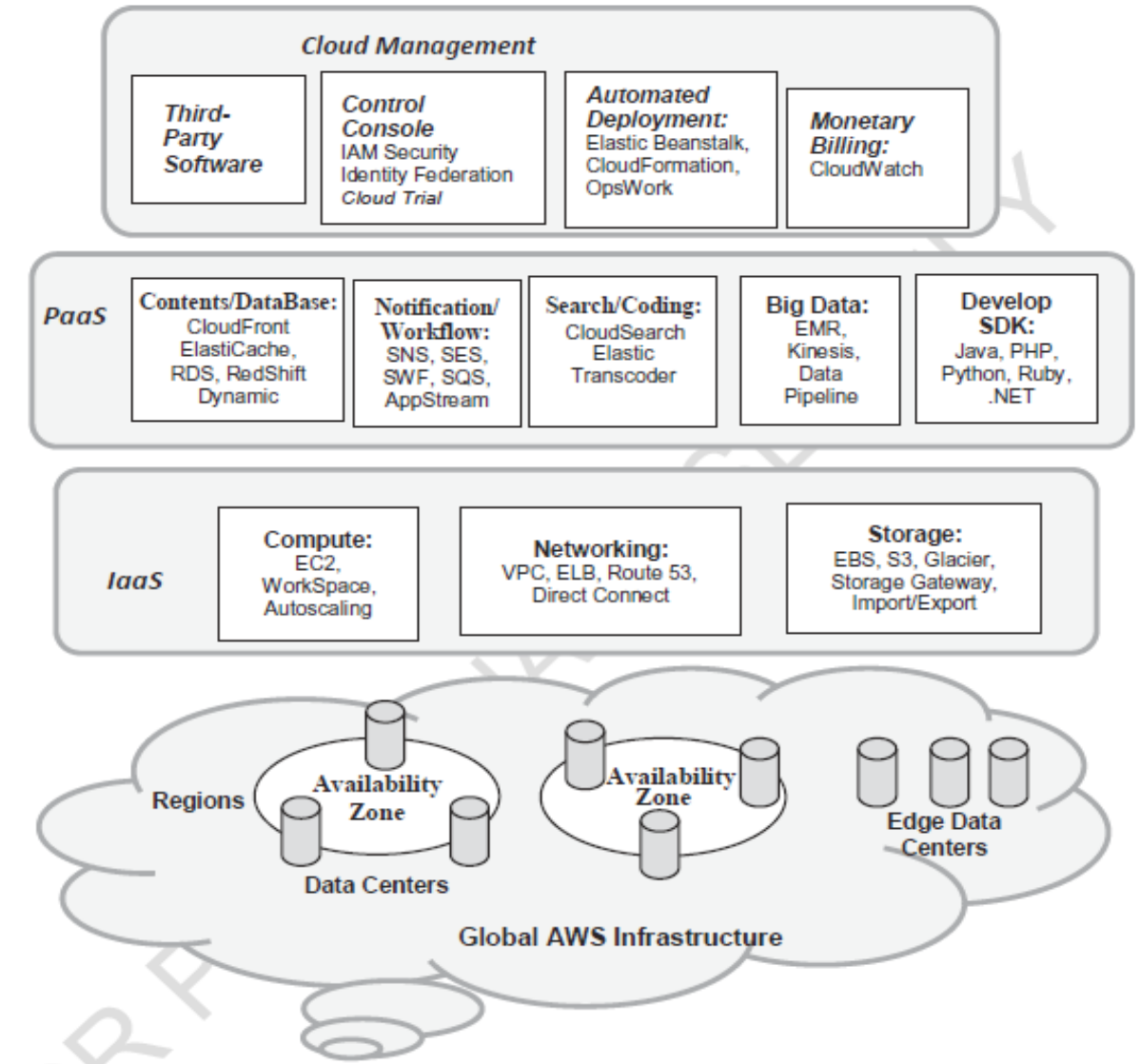


Figure 4.14
The AWS public cloud consisting of the top management layer, PaaS, and IaaS platforms, and the global infrastructure built over data centers in availability zones located in various regions.

Table 4.9
Compute, storage, database, and networking services in AWS cloud

Category	Offering	Service Modules or Short Description
Compute	EC2	Virtual servers in the AWS cloud
	Lambda	Run code in response to events
	EC2 Container Service	Run and manage Docker containers
Storage & Content Delivery	S3	Scalable storage in the AWS cloud
	Elastic File System	Fully managed file system for EC2
	Storage Gateway	Integrate on-premises IT facilities with cloud storage
	Glacier	Archive storage in the AWS cloud
	CloudFront	Global content delivery network
Database	RDS	MySQL, PostgreSQL, Oracle, SQL Server
	DynamicDB	Predictable and scalable NoSQL data store
	ElastiCache	In-memory cache
	RedShift	Managed petabyte-scale warehouse service
Networking	VPC	Virtual private cloud as isolated cloud resources
	Direct Connect	Dedicated network connection to AWS
		Scalable DNS and domain name registration

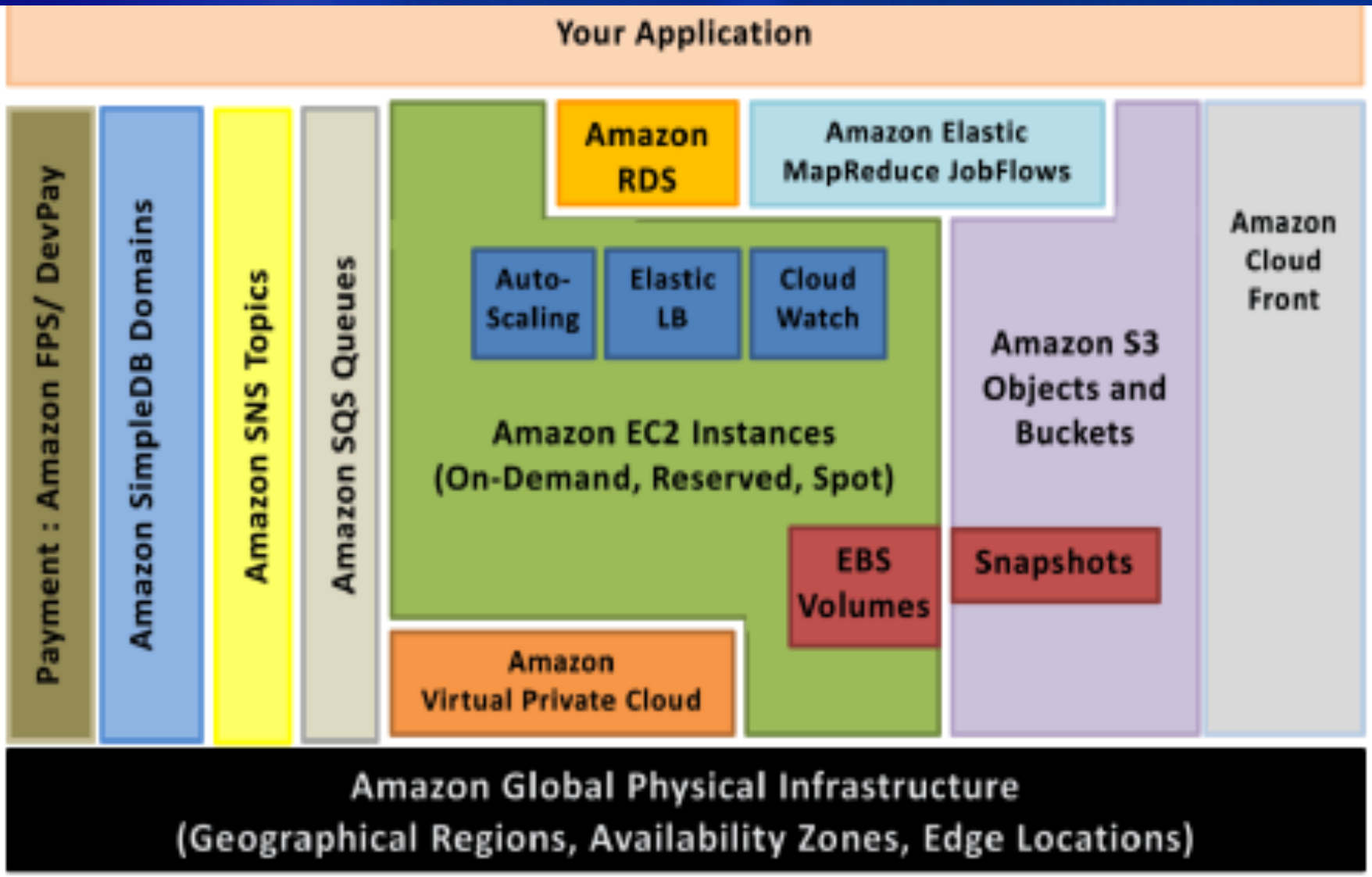
Compute, Storage, Database and Networking Services in The AWS Cloud in 2015

Category	Offering	Service Modules or Short Description
Compute	EC2	Virtual Servers in the AWS Cloud
	Lambda	Run Code in Respoinse to Events
	EC2 Container Service	Run and Manage Docker Containers
Storage & Content Delivery	S3	Scalable Storage in the AWS Cloud
	Elastic File System	Fully Management File System for EC2 (Preview)
	Storage Gateway	Integrate On-Premises IT Facilities with Cloud Storage
	Glacier	Archive Storage in the Cloud
	CloudFront	Global Content Delivery Network
Database	RDS	MySQL, Postgres, Oracle, SQL Server, and Amazon
	DynamicDB	Predictable and Scalable NoSQL Data Store
	ElastiCache	In-Memory Cache
	Redshift	Managed Petabyte-Scale Warehouse Service
Networking	VPC	Virtual Private Cloud as Isolated Cloud Resources
	Direct Connect	Dedicated Network Connection to AWS
	Route S3	Scalable DNS and Domain Name Registration

IaaS Clouds and Their Infrastructures and Service Offerings (Aug. 2015)

Cloud Name	Virtual Machine Instance Configurations	API and Access Tools
Amazon EC2	Each instance has 1-20 EC2 processors, 1.7-15 GB memory, and 160 TB storage	CLI or Web Service (WS) portal
GoGrid	Each instance has 1-6 CPUs, 0.5-8 GB memory and 30-480 GB storage	REST, Java, PHP, Python, Ruby
Rackspace Cloud	Each instance has a 4-core CPU, 0.25-16 GB memory and 10-620 GB storage	REST, Python, PHP, Java, C#, .NET
Flexiscale in UK	Each instance has 1-4 CPUs, 0.5-16 GB memory, and 20-270 GB storage	Web console

AWS Architecture



Important AWS Links To Visit and Documented Reports/Articles To Read

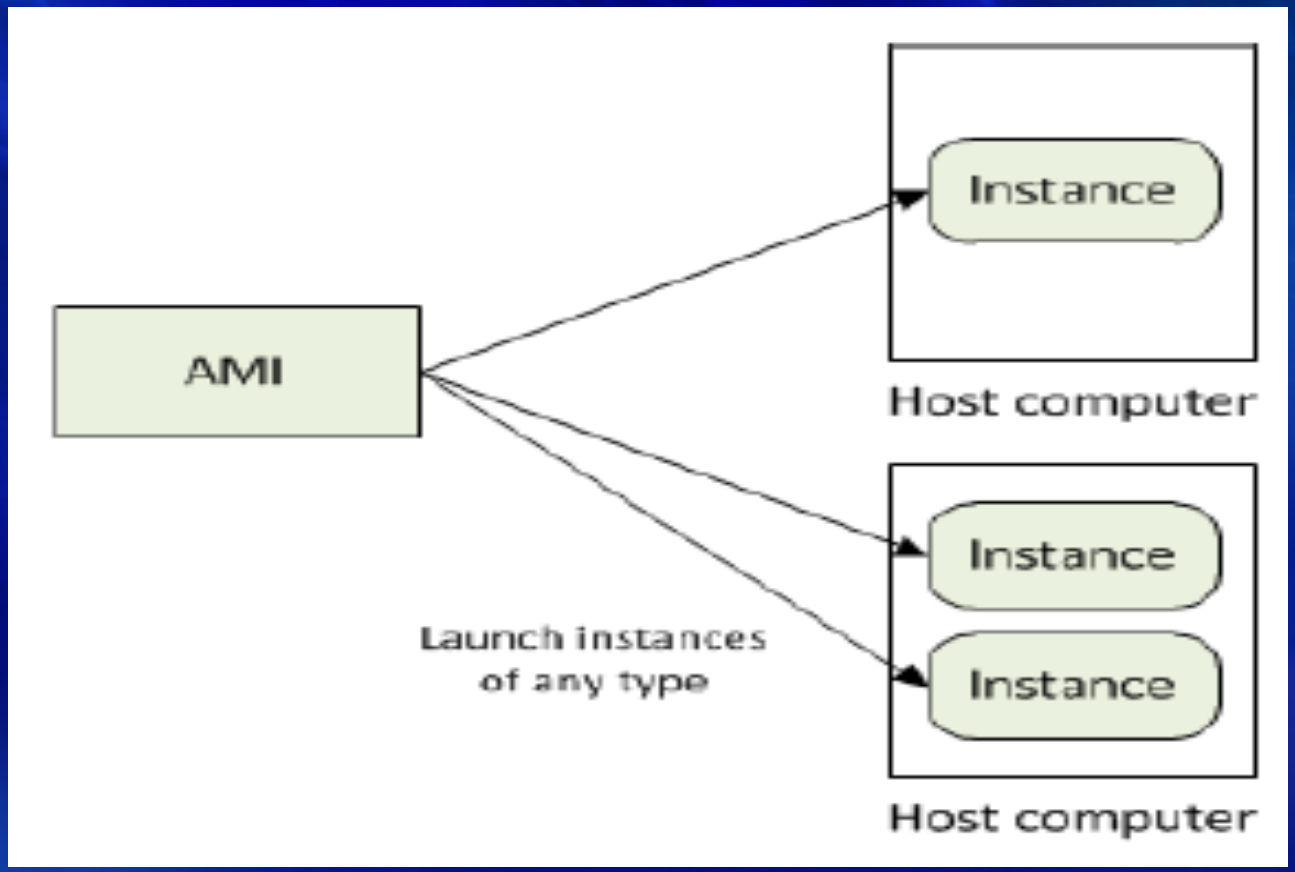
1. The main AWS site: http://aws.amazon.com
2. Just use http://aws.amazon.com/ec2/ to access ec2, or http://aws.amazon.com/s3/ for S3, or …./sqs/, ../sns/, or ../simpledb/, ……/sdk/, ……/fps/, etc.
3. Many PDF reports and articles, application examples, SDK tools, etc. can be found in the above websites. You do not need an account to access these sites.
4. You need to establish your team accounts on AWS ASAP. You need to use the AWS in doing some problems in HW#2 and to do the team project as well.

EC2 Features on The AWS Platform

- Virtual computing environments, known as *instances*
- Pre-configured templates for your instances, known as *Amazon Machine Images (AMIs)*, that package the bits you need for your server (including the operating system and additional software)
- Various configurations of CPU, memory, storage, and networking capacity for your instances, known as *instance types*
- Secure login information for your instances using *key pairs* (AWS stores the public key, and you store the private key in a secure place)
- Storage volumes for temporary data that's deleted when you terminate your instance, known as *instance store volumes*
- Persistent storage volumes for your data using Amazon Elastic Block Store (Amazon EBS), known as *Amazon EBS volumes*
- Multiple physical locations for your resources, such as instances and Amazon EBS volumes, known as *regions* and *Availability Zones*
- A firewall that enables you to specify the protocols, ports, and source IP ranges that can reach your instances using *security groups*
- Static IP addresses for dynamic cloud computing, known as *Elastic IP addresses*
- Metadata, known as *tags*, that you can create and assign to your Amazon EC2 resources
- Virtual networks you can create that are logically isolated from the rest of the AWS cloud, and that you can optionally connect to your own network, known as *virtual private clouds* (VPCs)

AMIs and Instance Types

- From an AMI, you can launch an instance, which is a copy of the AMI running as a virtual server in the cloud.
- You can launch multiple instances of an AMI

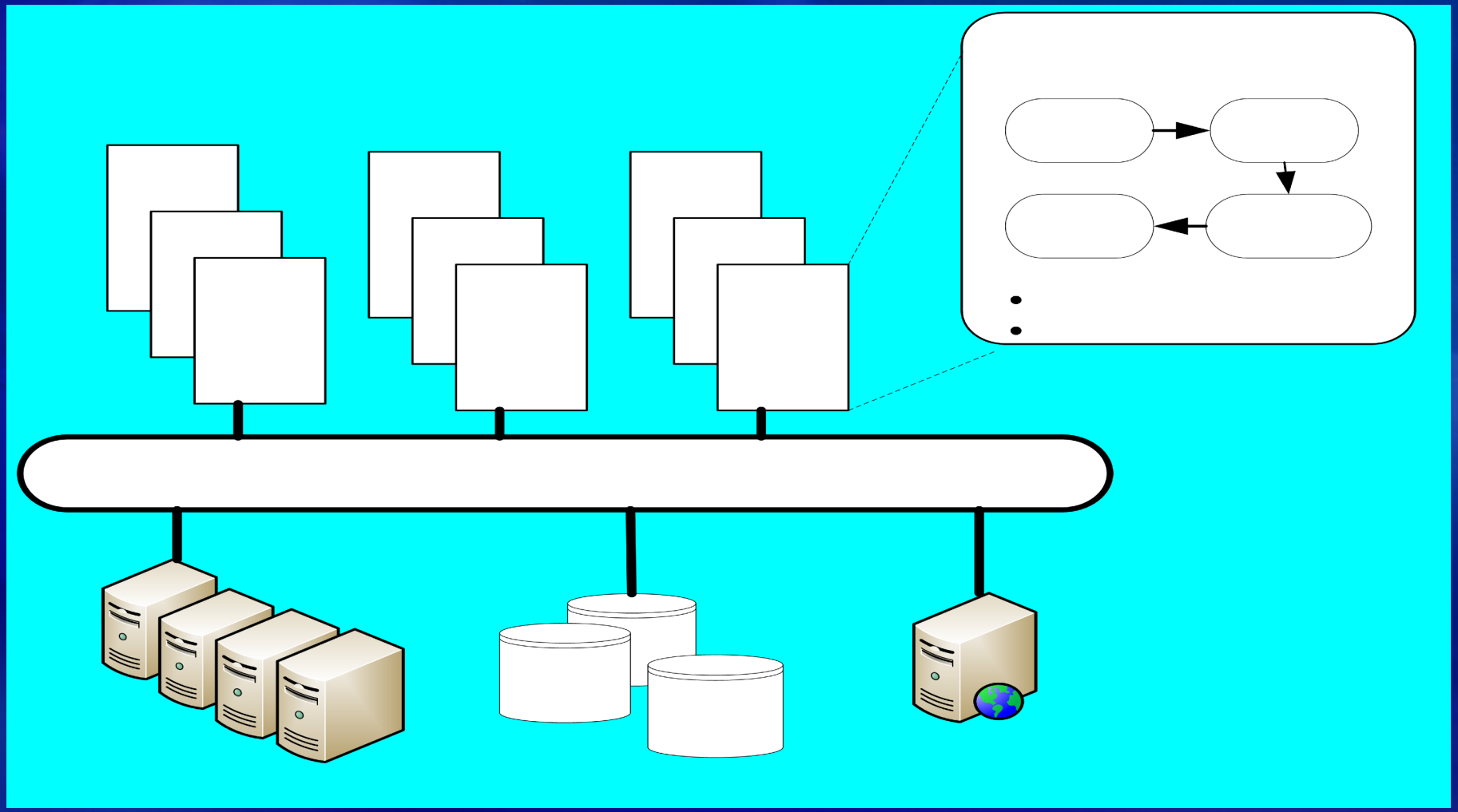


Amazon Machine Images (AMI)

Image Type	Definition
Private	Images created by you, which are private by default. You can grant access to other users to launch your private images.
Public	Images created by users and released to the Amazon Web Services community, so anyone can launch instances based on them and use them any way they like. The Amazon Web Services Developer Connection Web site lists all public images.
Paid	You can create images providing specific functions that can be launched by anyone willing to pay you per each hour of usage on top of Amazon charges.

- AMI is a packaged server environment in EC2, based on Linux running any user software or application. AMIs are the templates for VM instances.
- Elastic IP address is specially reserved for EC2. Elastic Block Store offers persistent storage for EC2 instances.

Amazon EC2 Execution Environment



Amazon EC2 Instances

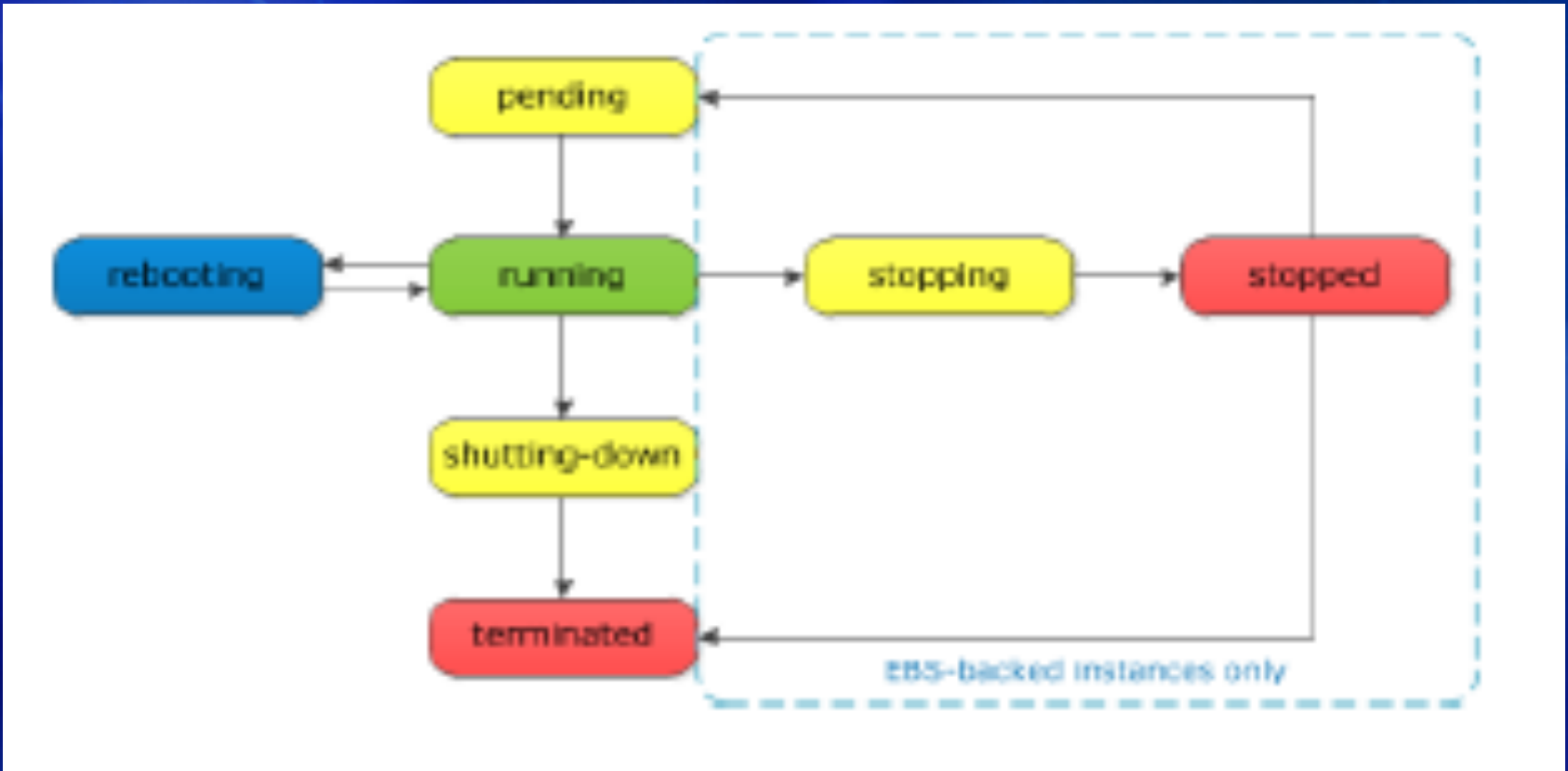
Family	Description
Cluster Compute	Have a very large amount of CPU coupled with increased networking performance. They're well-suited for High Performance Compute (HPC) applications and other demanding network-bound applications. For more information, see Overview (p. 121).
Cluster GPU	Provide general-purpose graphics processing units (GPUs), with proportionally high CPU and increased network performance for applications that benefit from highly parallelized processing. They're well-suited for HPC applications as well as rendering and media processing applications. For more information, see Overview (p. 121).
High CPU	Have proportionally more CPU resources than memory (RAM). They're well-suited for compute-intensive applications.
High I/O	Provide tens of thousands of low-latency, random I/O operations per second (IOPS) to an application. They're well-suited for NoSQL databases, clustered databases, and OLTP (online transaction processing) systems. For more information, see High I/O Instances (p. 117).
High Memory	Have proportionally more memory resources. They're well suited for high-throughput applications, such as database and memory caching applications.
High-Memory Cluster	Have large amounts of memory coupled with high CPU and network performance. These instances are well suited for in-memory analytics, graph analysis, and scientific computing applications.
High Storage	Provide very high storage density and high sequential read and write performance per instance. They are well-suited for data warehousing, Hadoop/MapReduce, and parallel file systems. For more information, see High Storage Instances (p. 119).
Micro	Provide a small amount of consistent CPU resources and enable you to burst CPU capacity when additional cycles are available. They're well-suited for lower throughput applications and websites that consume significant compute cycles periodically. For more information, see Micro Instances (p. 110).
Standard	Have memory-to-CPU ratios suitable for most general-purpose applications.

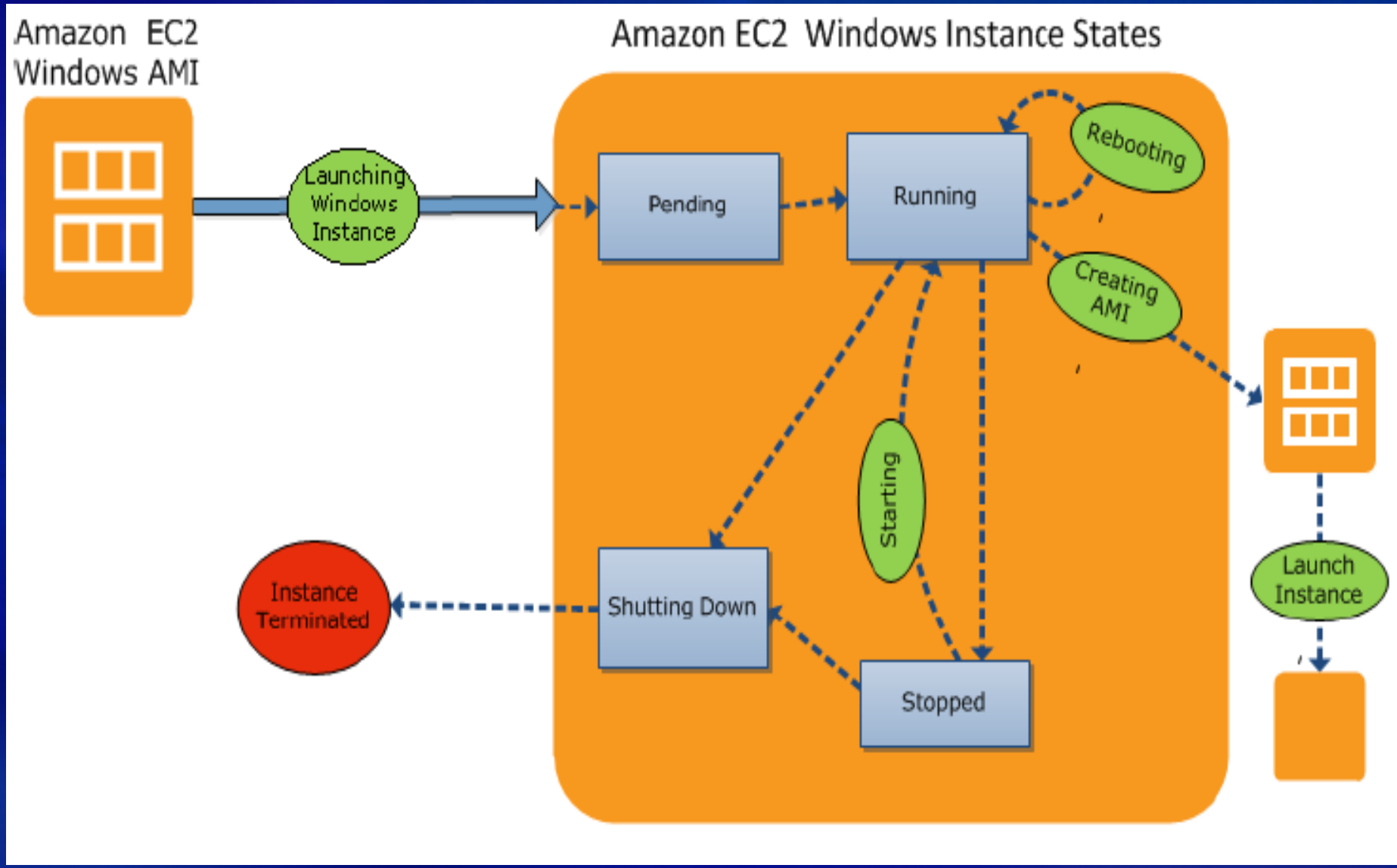
Instance Types available on Amazon EC2 (Oct. 6, 2010)

Compute Instance	Memory GB	ECU or EC2 Units	Virtual Cores	Storage GB	32/64 Bit
Standard: Small	1.7	1	1	160	32
Standard: Large	7.5	4	2	850	64
Standard: Extra Large	15	8	4	1690	64
Micro	0.613	Up to 2		Only EBS	32 or 64
High-Memory	17.1	6.5	2	420	64
High-Memory: Double	34.2	13	4	850	64
High-Memory: Quadruple	68.4	26	8	1690	64
High-CPU : Medium	1.7	5	2	350	32
High-CPU: Extra Large	7	20	8	1690	64
Cluster Compute	23	33.5	8	1690	64

Note: The information in this table is outdated. You need to visit the AWS website to obtain the current services and price quote.

EC2 Instance Lifecycle





Name	Memory	Compute Units	Virtual Cores	Instance Store Volumes*	Architecture	Network Performance	Available for Spot Instance	API Name
Cluster Compute								
Cluster Compute Eight Extra Large	60.5 GiB	88	16 (2 x Intel Xeon E5-2670, eight-core with hyperthread)	3360 GB (4 x 840 GB)	64-bit	Very high (10 Gbps Ethernet)	Yes	cc2.8xlarge
Cluster GPU								
Cluster GPU Quadruple Extra Large**	22.5 GiB (see note after this table)	33.5	8 (2 x Intel Xeon X5570, quad-core with hyperthread), plus 2 NVIDIA Tesla M2050 GPUs	1680 GB (2 x 840 GB)	64-bit	Very high (10 Gbps Ethernet)	Yes	cg1.4xlarge
High CPU								
High-CPU Extra Large	7 GiB	20	8 (with 2.5 ECUs each)	1690 GB (4 x 420 GB)	64-bit	High	Yes	c1.xlarge
High-CPU Medium	1.7 GiB†	5	2 (with 2.5 ECUs each)	350 GB (1 x 350 GB)	32-bit and 64-bit	Moderate	Yes	c1.medium
High I/O								
High I/O Quadruple Extra Large***	60.5 GiB	35	8 (with 4.37 ECUs each)	2,048 GB (2 x 1,024 GB) SSD	64-bit	Very high (10 Gbps Ethernet)	No	hi1.4xlarge
High Memory								
High Memory Double Extra Large	34.2 GiB	13	4 (with 3.25 ECUs each)	850 GB (1 x 850 GB)	64-bit	Moderate	Yes	m2.2xlarge

Available EC2 Instances

Name	Memory	Compute Units	Virtual Cores	Instance Store Volumes*	Architecture	Network Performance	Available for Spot Instance	API Name
High-Memory Extra Large	17.1 GiB	6.5	2 (with 3.25 ECUs each)	420 GB (1 x 420 GB)	64-bit	Moderate	Yes	m2.xlarge
High-Memory Quadruple Extra Large	68.4 GiB	26	8 (with 3.25 ECUs each)	1680 GB (2 x 840 GB)	64-bit	High	Yes	m2.4xlarge
High-Memory Cluster								
High-Memory Cluster Eight Extra Large	244 GiB	88	16 (2 x Intel Xeon E5-2670, eight-core)	240 GB (2 x 120 GB) SSD	64-bit	Very high (10 Gbps Ethernet)	Yes	cr1.8xlarge
High Storage								
High Storage Eight Extra Large	117 GiB	35	16 (8 cores + 8 hyperthreads)	49,152 GB (24 x 2,048 GB) hard disk drives	64-bit	Very high (10 Gbps Ethernet)	No	hs1.8xlarge
Micro								
Micro	615 MiB	Up to 2 (for short periodic bursts)	1	EBS storage only	32-bit and 64-bit	Very Low	Yes	t1.micro
Standard								
M1 Extra Large	15 GiB	8	4 (with 2 ECUs each)	1680 GB (4 x 420 GB)	64-bit	High	Yes	m1.xlarge
M1 Large	7.5 GiB	4	2 (with 2 ECUs each)	840 GB (2 x 420 GB)	64-bit	Moderate	Yes	m1.large
M1 Medium	3.75 GiB	2	1	410 GB (1 x 410 GB)	32-bit and 64-bit	Moderate	Yes	m1.medium
M1 Small	1.7 GiB†	1	1	160 GB (1 x 160 GB)	32-bit and 64-bit	Low	Yes	m1.small
M3 Double Extra Large	30 GiB	26	8 (with 3.25 ECUs each)	EBS storage only	64-bit	High	Yes	m3.2xlarge

Available EC2 Instances

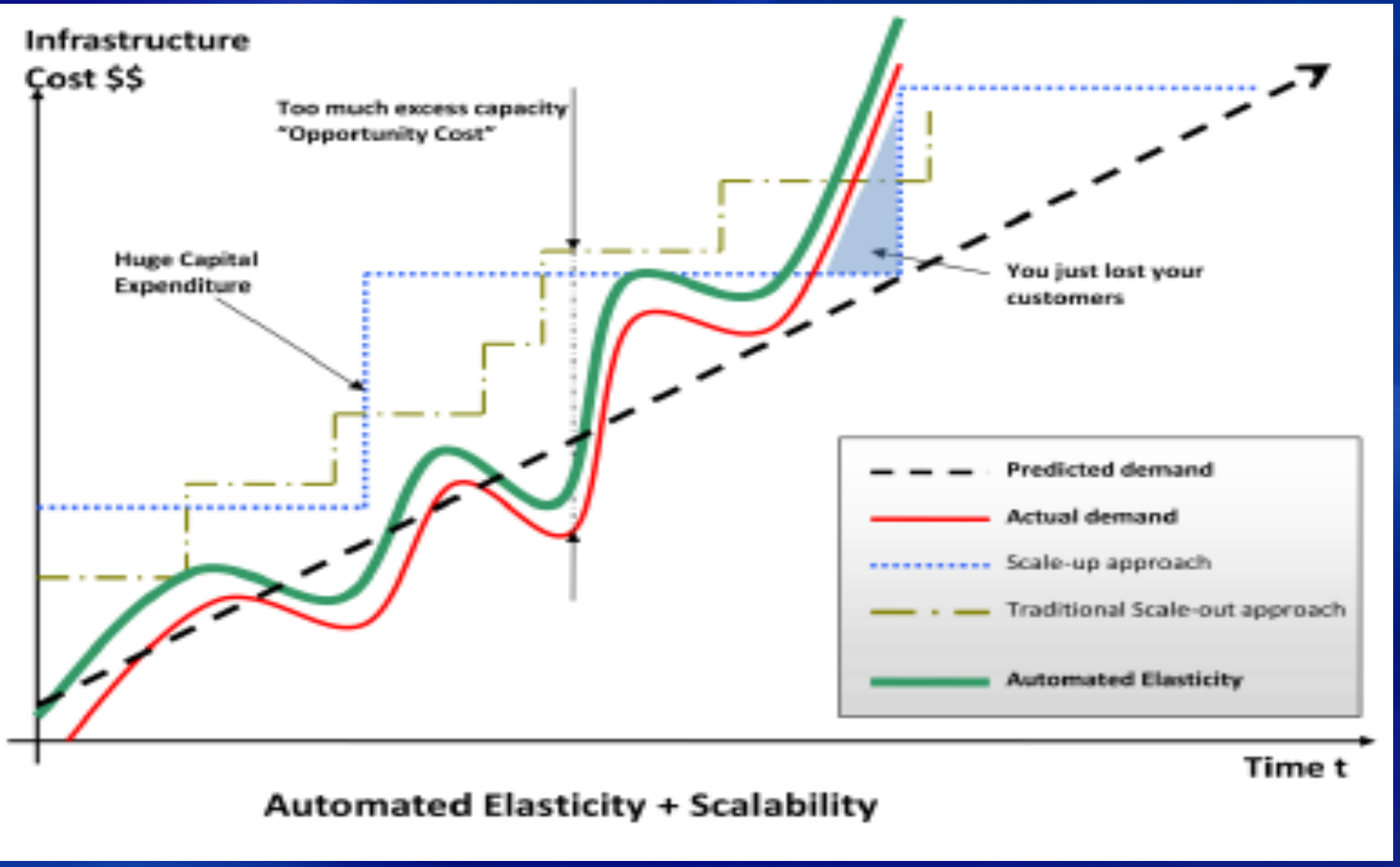
Use of EC2 Instances in AWS platform

1. Standard Instances are well suited for most applications
2. Micro Instances provide a small amount of consistent CPU resources and allow you to burst CPU capacity when additional cycles are available. They are well suited for lower throughput applications and websites that consume significant compute cycles periodically
3. High-Memory Instances offer large memory sizes for high throughput applications, including database and memory caching applications
4. High-CPU Instances have proportionally more CPU resources than memory (RAM) and are well suited for compute-intensive applications
5. Cluster Compute Instances provide proportionally high CPU with increased network performance and are well suited for High Performance Compute (HPC) applications and other demanding network-bound applications. AWS uses 10 Gigabit Ethernet

Auto Scaling Features on Amazon EC2

- Auto Scaling allows you to scale your Amazon EC2 capacity up or down automatically according to conditions you define
- With Auto Scaling, you can ensure that the number of Amazon EC2 instances you're using increases seamlessly during demand spikes to maintain performance, and decreases automatically during demand lulls to minimize costs
- Auto Scaling is particularly well suited for applications that experience hourly, daily, or weekly variability in usage
- Auto Scaling is enabled by Amazon CloudWatch and available at no additional charge beyond Amazon CloudWatch fees

Understanding Elasticity in Cloud Resources



S3: Simple Storage Service
Storage Cloud Service: Amazon

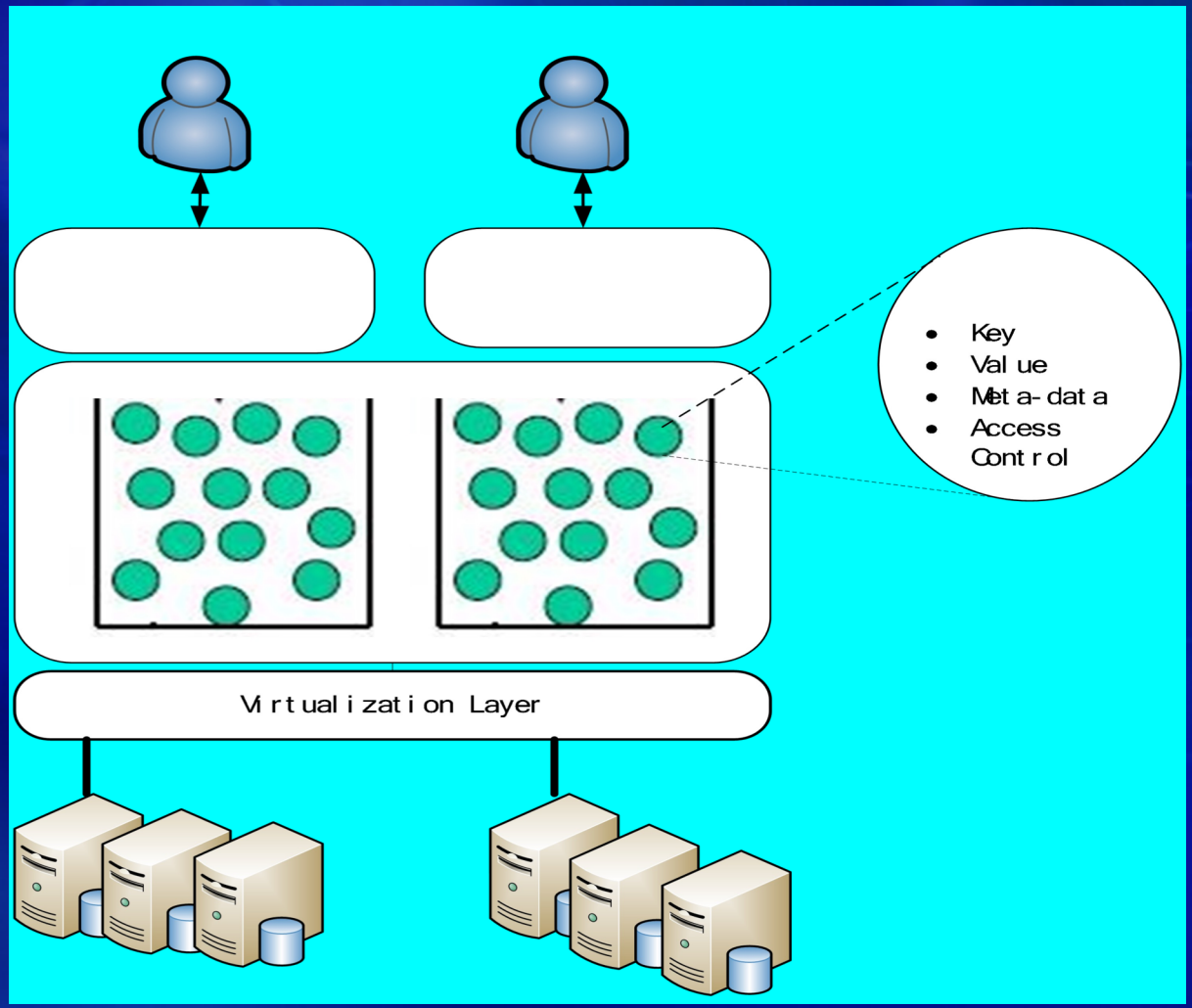
- Object-based storage
- 1 B – 5 GB / object
- Redundant thru geographic dispersion
- 99.99% availability goal
- Private or public
- Per-object URLs & ACLs
- BitTorrent Support

$.15 per GB per month storage

$.01 for 1000 to 10,000 requests

$.10 - $.18 per GB data transfer

Amazon S3 for Storage Provisioning



- Object is the basic unit of data
- Bucket for storing objects
- Key for data object retrieval
- Object is attributed to value, metadata, and access control

Key Features in Amazon S3

- Write, read, and delete objects containing from 1 byte to 5 TB of data each. The number of objects you can store is unlimited
- Each object is stored in a bucket and retrieved via a unique, developer-assigned key
- A bucket can be stored in one of several Regions. You can choose a Region to optimize for latency, minimize costs, or address regulatory requirements
- The US Standard Region automatically routes requests to facilities in Northern Virginia or the Pacific Northwest using network maps
- Objects stored in a Region never leave the Region unless you transfer them out. For example, objects stored in the EU (Ireland) Region never leave the EU

Steps to Use Amazon S3

- Create a bucket to store your data. You can choose a Region where your bucket and object(s) reside to optimize latency, minimize costs, or address regulatory requirements
- Upload objects to your bucket. Your data is durably stored and backed by the Amazon S3 Service Level Agreement
- Optionally, set access controls. You can grant others access to your data from anywhere in the world

Application, Mobile and Analytics Services in the AWS Cloud

Category	Offering	Service Modules or Short Description
Application Services	SQS	Message Queue Services
	SWF	Workflow for Coordinating App Components
	AppStream	Low Latency Application Streaming
	Elastic Transcoder	Easy-To-Use Scalable Media Transcoding
	SES	Email Sending Service
	CloudSearch	Managed Search Service
	API Gateway	Build, Deploy and Mange APIs
Mobile Services	Cognito	User Identity and App Data Synchronization
	Device Farm	Test Android and iOS apps on Mobile Devices
	Mobile Analytics	Collect, View and Export App Analytics
	SNS	Simple Push Notification Service
Analytics Services	EMR	Managed Elastic Hadoop Framework
	Kinesis	Real-Time Processing of Streaming Big Data
	Data Pipeline	Orchestration for Data-Driven Workflows
	Machine Learning	Build Smart Applications Quickly and Easily

Amazon Simple Queue Service (Amazon SQS)

- Amazon SQS is a fast, reliable, scalable, fully managed queue service. SQS makes it simple and cost-effective to decouple the components of a cloud application. You can use SQS to transmit any volume of data, at any level of throughput, without losing messages.

- Amazon SQS can be used with other AWS Services such as Redshift, DynamoDB, RDS, EC2, and S3, to make distributed applications more scalable and reliable. Some common queue designs follow:

Amazon Simple Queue Service (Amazon SQS)

- Work queues: Decouple components of a distributed application that may not all process the same amount of work simultaneously
- Buffer and batch operations: Add scalability and reliability to your architecture, and smooth out temporary volume spikes without losing messages or increasing latency
- Request offloading: Move slow operations off of interactive request paths by enqueing the request
- Fanout: Combine SQS with Simple Notification Service (SNS) to send identical copies of a message to multiple queues in parallel for simultaneous processing

Amazon Simple Notification Services (SNS) :

- The SNS is a fast, flexible, fully managed push messaging service. SNS makes it simple and cost-effective to push to mobile devices such as iPhone, iPad, Android, Kindle Fire, and internet connected smart devices, as well as pushing to other distributed services
- Besides pushing directly to mobile devices, SNS can also deliver notifications by SMS text message or email, to Simple Queue Service (SQS) queues, or to any HTTP endpoint
- Amazon SNS lets you push messages to mobile devices or distributed services, via API or an easy-to-use management console. You can seamlessly scale from a handful of messages per day to millions of messages or higher
- With SNS you can publish a message once, and deliver it one or more times. So you can choose to direct unique messages to individual Apple, Google or Amazon devices, or broadcast deliveries to many mobile devices with a single publish request

Amazon SimpleDB Service

- Provide a simplified relational database model. Structured data are organized into domains in terms of relational tables.
- Data items correspond to rows in the table. Specific attributes form the columns.
- Cells in the table carry the values of attributes. An attribute may have multiple values.

Amazon Elastic Block Service (EBS)

- EBS provides volume block interface from 1 GB to 1 TB for saving and restoring VM instances used in EC2.
- The user can also use EBS to save persistent data to be used in various EC2 instances.
- This is different from S3, which is a messaging interface. EBS acts more like a distributed file system. in terms of relational tables.

Administraion, Security, Enterprise and Deployment Services in the AWS Cloud

Category	Offering	Service Modules or Short Description
Administration & Security	Directory Service	Managed Directory in the AWS Cloud
	Identity/Access Manager	Access Control and Key Management
	Trusted Advisor	AWS Cloud Optimization Expert
	Cloud Trail	User Activity and Change Tracking
	Configuration	Resource Configurations and Inventory
	CloudWatch	Resource and Application Monitory
	Service Catalog	Personalized Catalog of AWS Resources
Enterprises Applications	Workplaces	Destops in the AWS Cloud
	WorkDocs	Secure Enterprise Storage and Sharing Service
	WorkMail	Secure Email and Calendaring Service
Deployment and Management	Elastic Beanstalk	AWS Application Containers
	OpsWorks	DevOps Application Managemnt Service
	CloudFormation	Templated AWS Resource Creation
	CodeEeploy	Automated Deployments
	CodeCommit	Managed Git Repositories
	Code Pipeline	Continuous Delivery of Codes

Table 2.3 Global information storage capacity in terms of total bytes in 2007.

Technology	Storage Devices	>Distribution
Analog, 19 EB, 6% of the total	Paper, film, audio tape and vinyl	*6%*
	Analog video tapes	*94%*
Digital, 280 EB, 94% of the total capacity	Portable media and flash drives	*2%*
	Portable hard disks	*2.4%*
	CDs and mini disks	*6.8%*
	Digital tapes	*11.8%*
	DVD and blue ray	*22.8%*
	PC/server hard disks	*44.5%*
	Others (memory cards, floppy disks, mobile phone, PDSs, cameras, video games, etc.)	*> 1%*

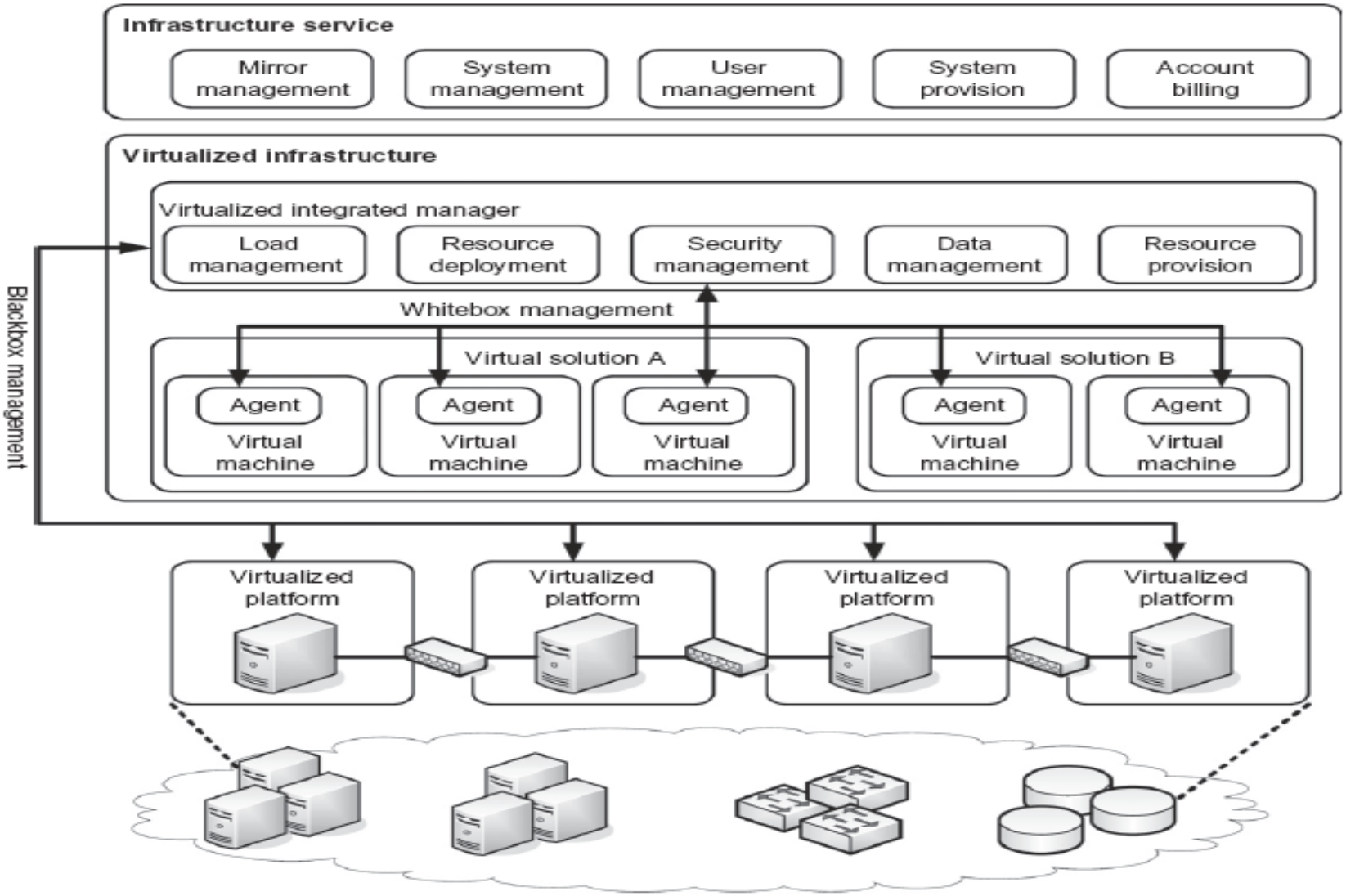


FIGURE 4.17

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

(Courtesy of Zhong-Yuan Qin, SouthEast University, China)

Other Public Clouds from Google, Microsoft, IBM, SalesForce, HP, SGI, etc.

Table 4.8
Public clouds offering PaaS services (August 2015)

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, Force.com	Apex, Eclipse-based IDE, web-based Wizard	Workflow, Excel-like, Web programming on demand	CRM and add-on app development for business
Microsoft Azure	.NET, Azure tools for MS Visual Studio	Dryad, Twister, .NET Framework	Enterprise and web applications
Amazon Elastic MapReduce	Hive, Pig, Cascading, Java, Ruby, Perl, Python, PHP, R, and C++	MapReduce, Hadoop, Spark	Data processing, e-mail, e-commerce, S3, and WorkDocs

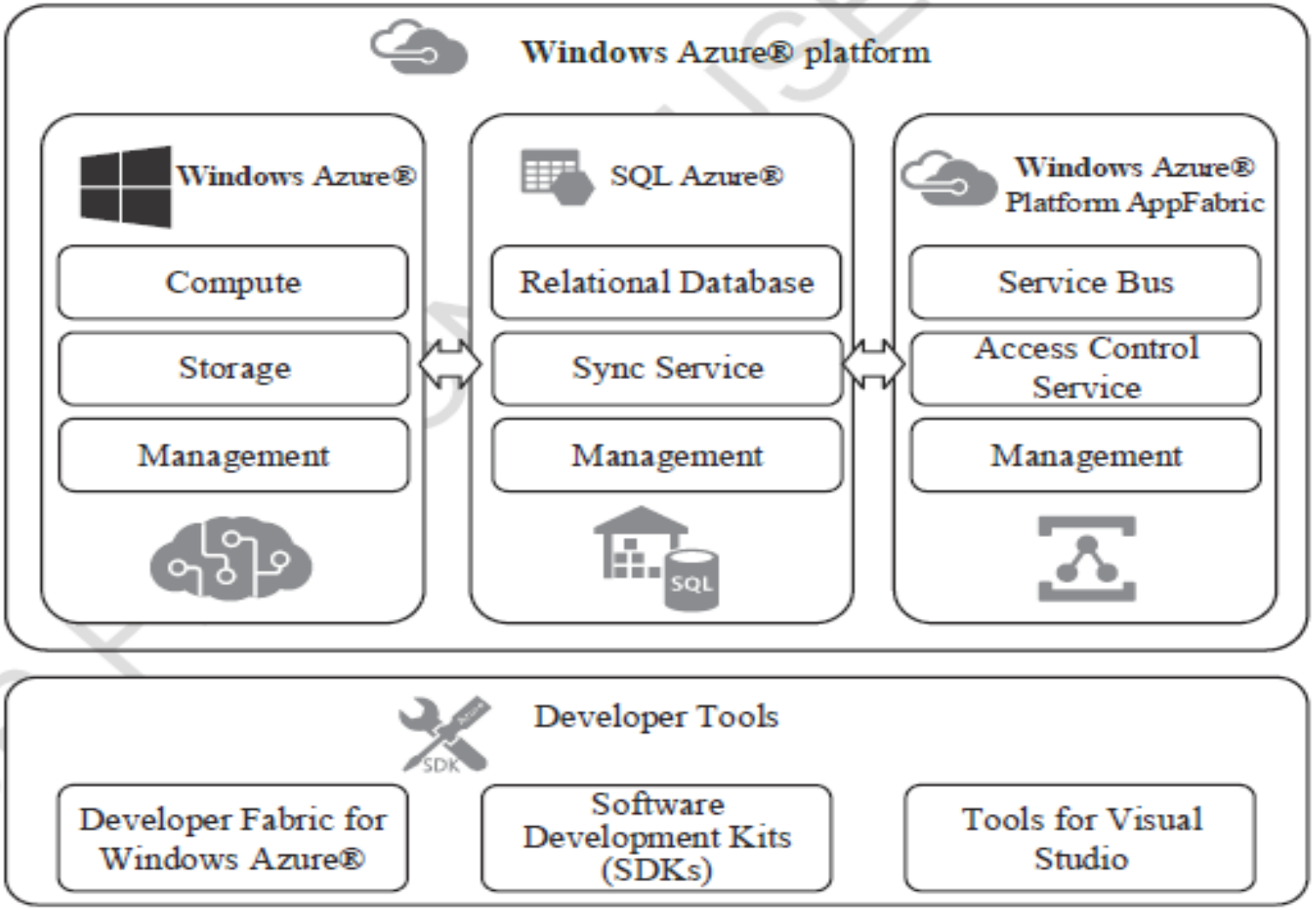


Figure 4.25
Microsoft Windows Azure platform for cloud computing. Courtesy of Microsoft, http://www.microsoft.com /windowsazure, 2016.

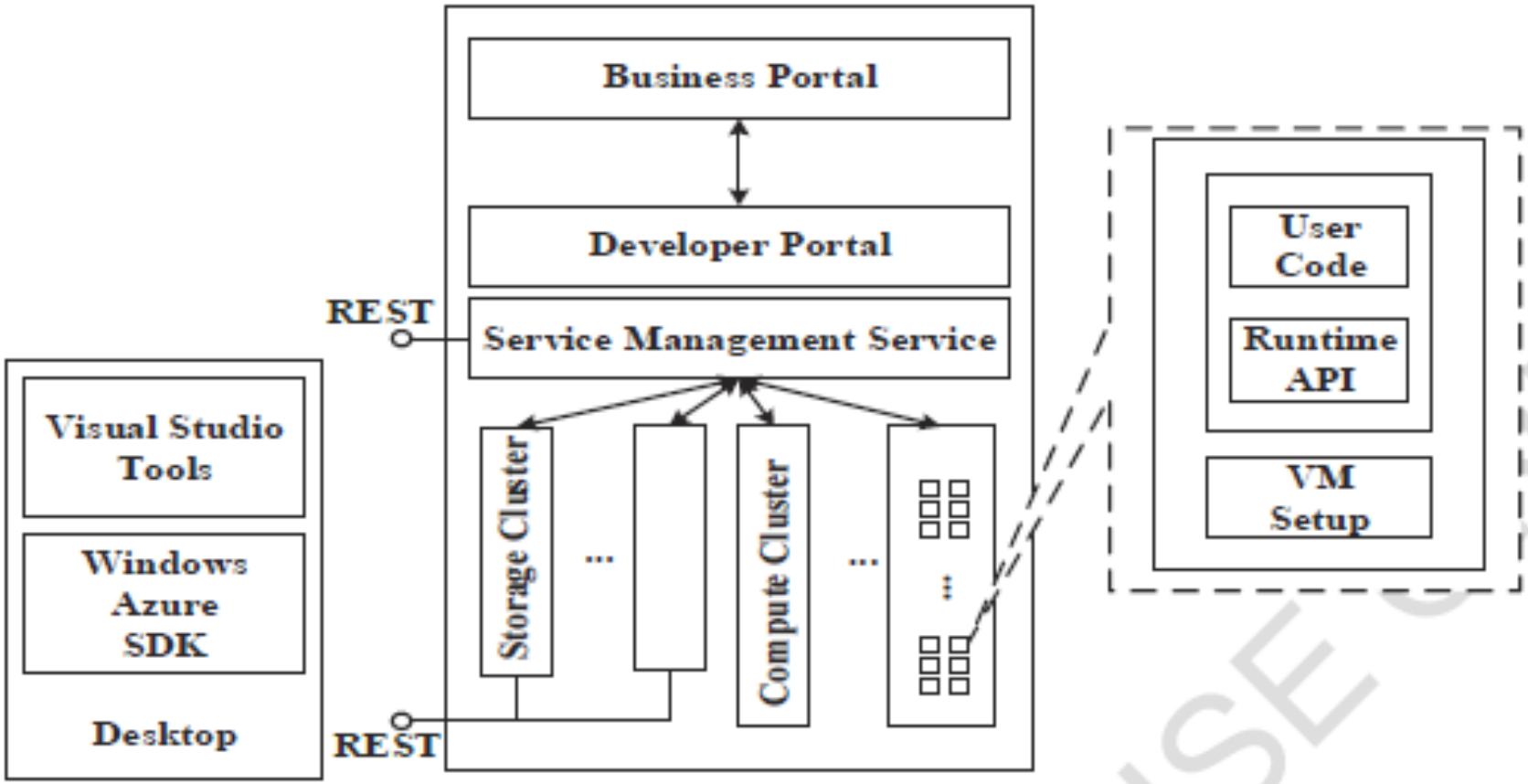


Figure 4.26
Microsoft Azure and its interfaces with business, developer, and client applications.

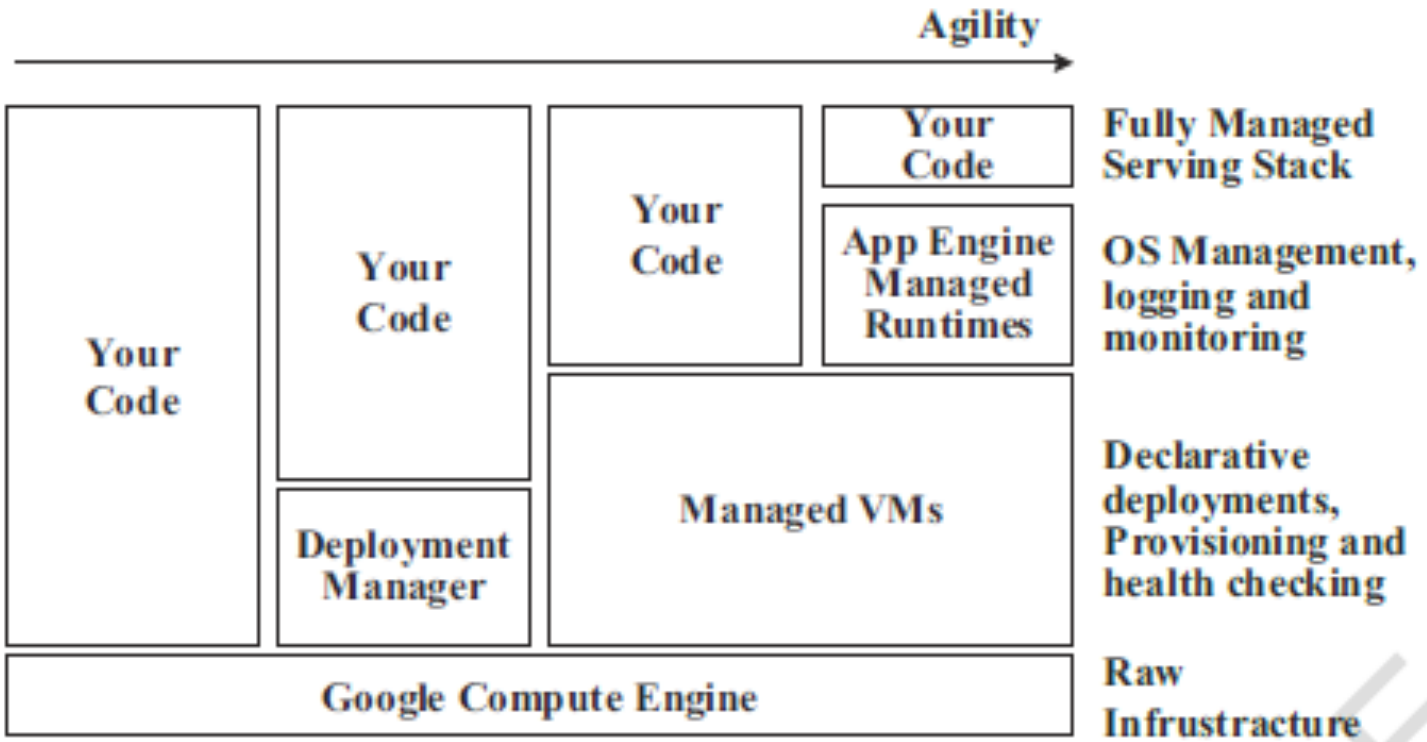


Figure 4.20

Building Google App Engine with increasing agility from raw infrastructure to Compute Engine, using OS support and managed VMs and runtimes to reduce user coding efforts. Courtesy of Google website, http://www.google.com, retrieved October 2016.

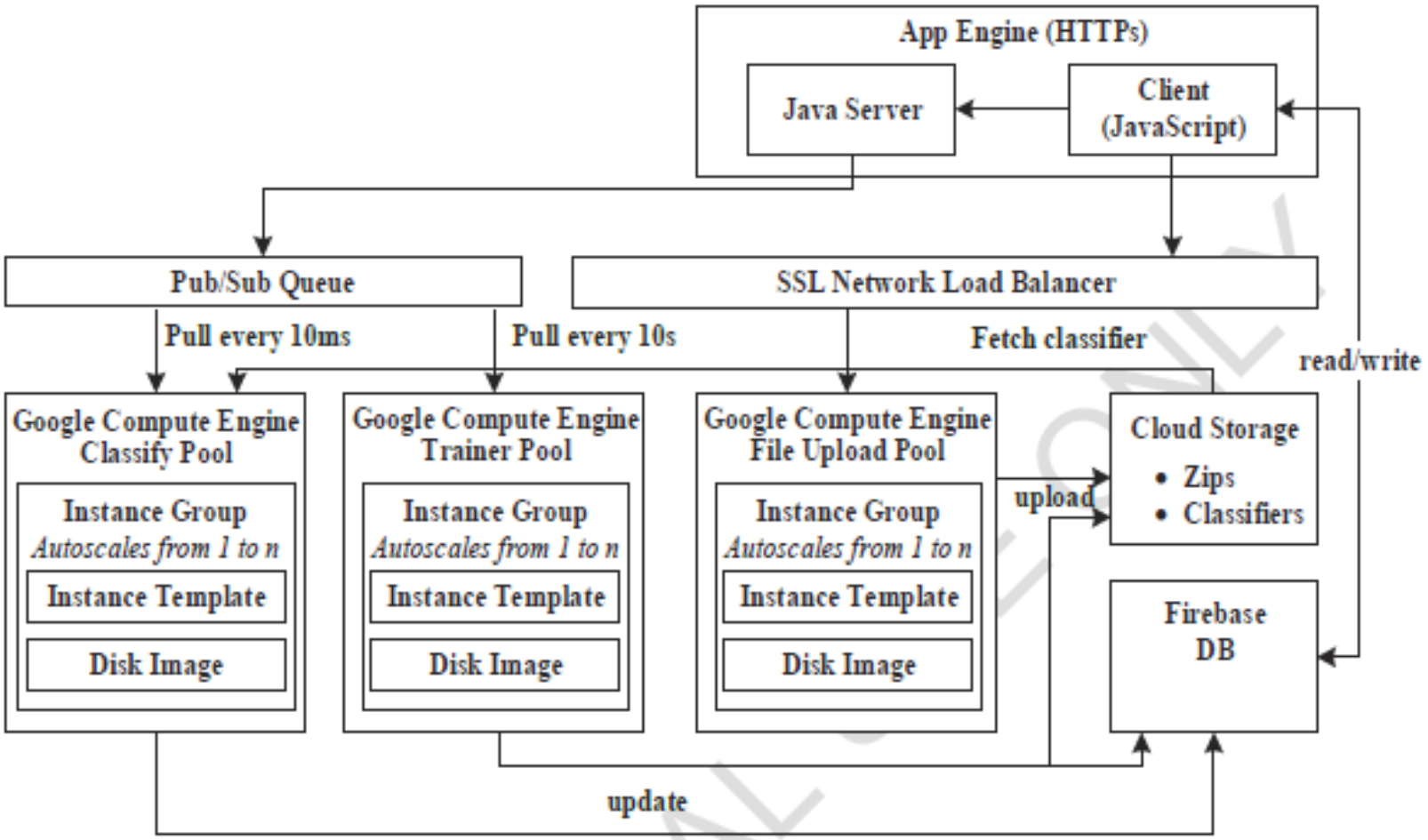


Figure 4.21
The architecture of Google Compute Engine as three functional pools and cloud storage to support Google App Engine applications development and executions.

SaaS Clouds and Service Offerings

Table 4.12
Four SaaS cloud platforms and their service offerings (August 2015)

Model	Amazon AWS	Google App Engine	Microsoft Azure	Salesforce
Platform Support	AWS EC2, S3, EMR, SNS, etc.	GAE, GFS, BigTable, MapReduce, etc.	Azure, .NET service, Dynamic CRM,	Force.com, Online CRM, Gifttag
SaaS Offerings	Elastic Beanstalk, CodeDeploy, OpsWorks, CodeCommit, Code-Pipeline, Mobile Analytics	Gmail, Docs, YouTube, WhatsApp	Live, SQL, Office 365 (OWA), Hotmail	Sales, service, market, data, collaboration, analytics
Security Features	CloudWatch, Trusted Advisor, Identity/Access Control	Chubby locks for security enforcement	Replicated data, rule-based access control	Admin./record security, use metadata API
APIs and Languages	API Gateway, Latin Pig	Web-based admin. console, Python	Azure portal, .NET Framework	Apex, Visualforce, AppExchange, SOSL, SOQL

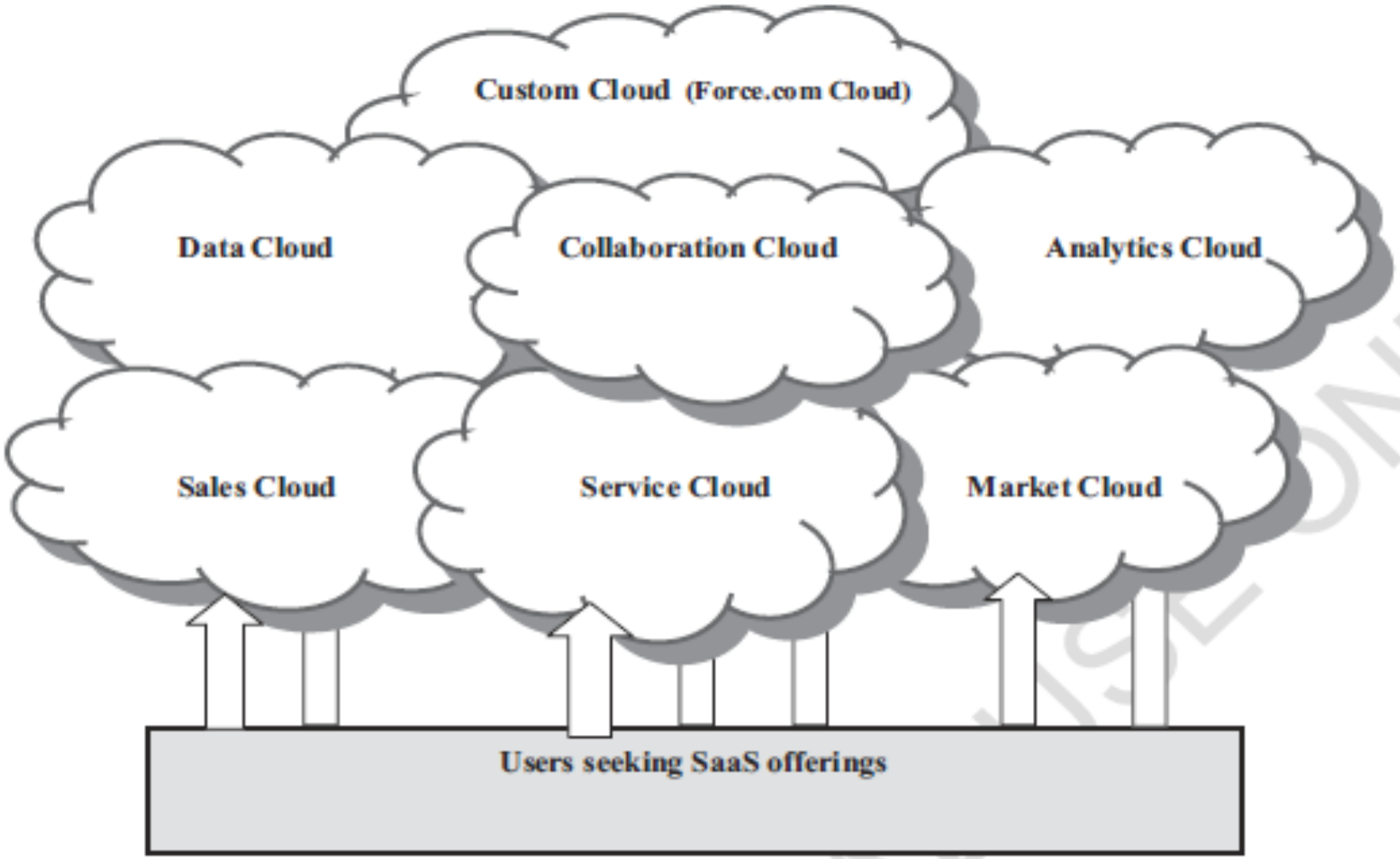


Figure 4.28
Seven Salesforce cloud service offerings: all for SaaS applications except the custom cloud offering PaaS applications.

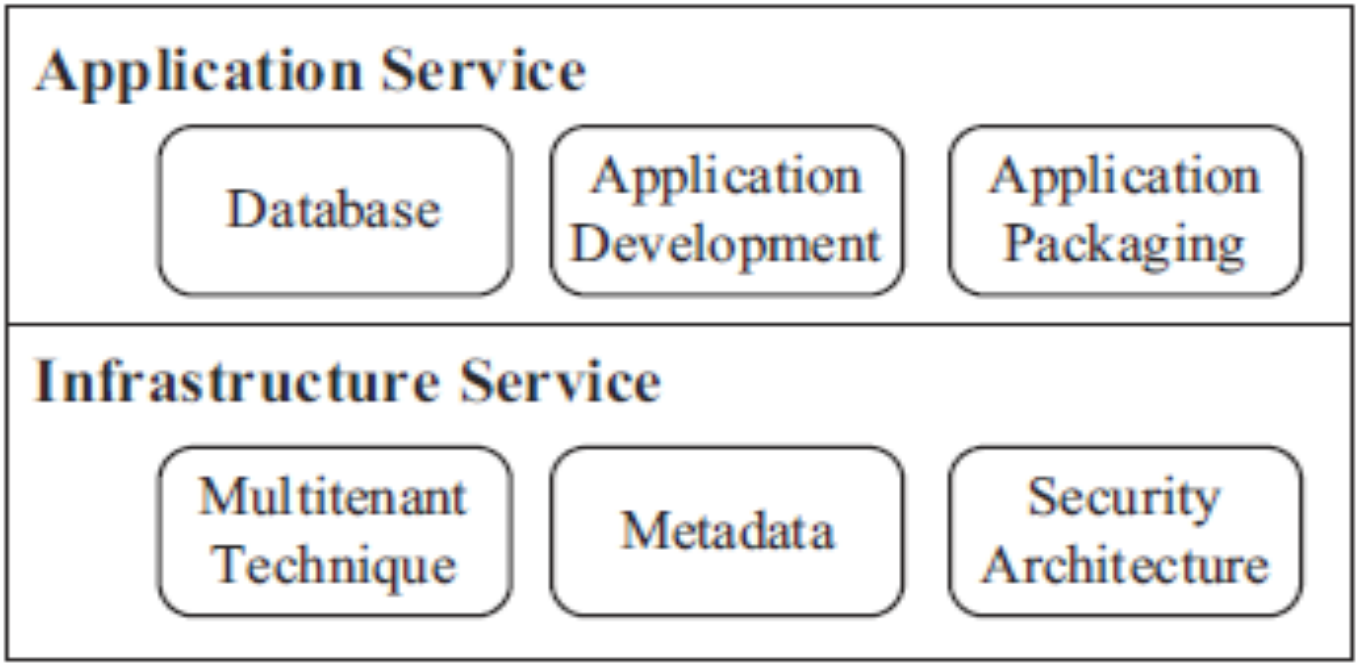


Figure 4.27
Salesforce Force.com cloud platform supporting both PaaS and SaaS applications.

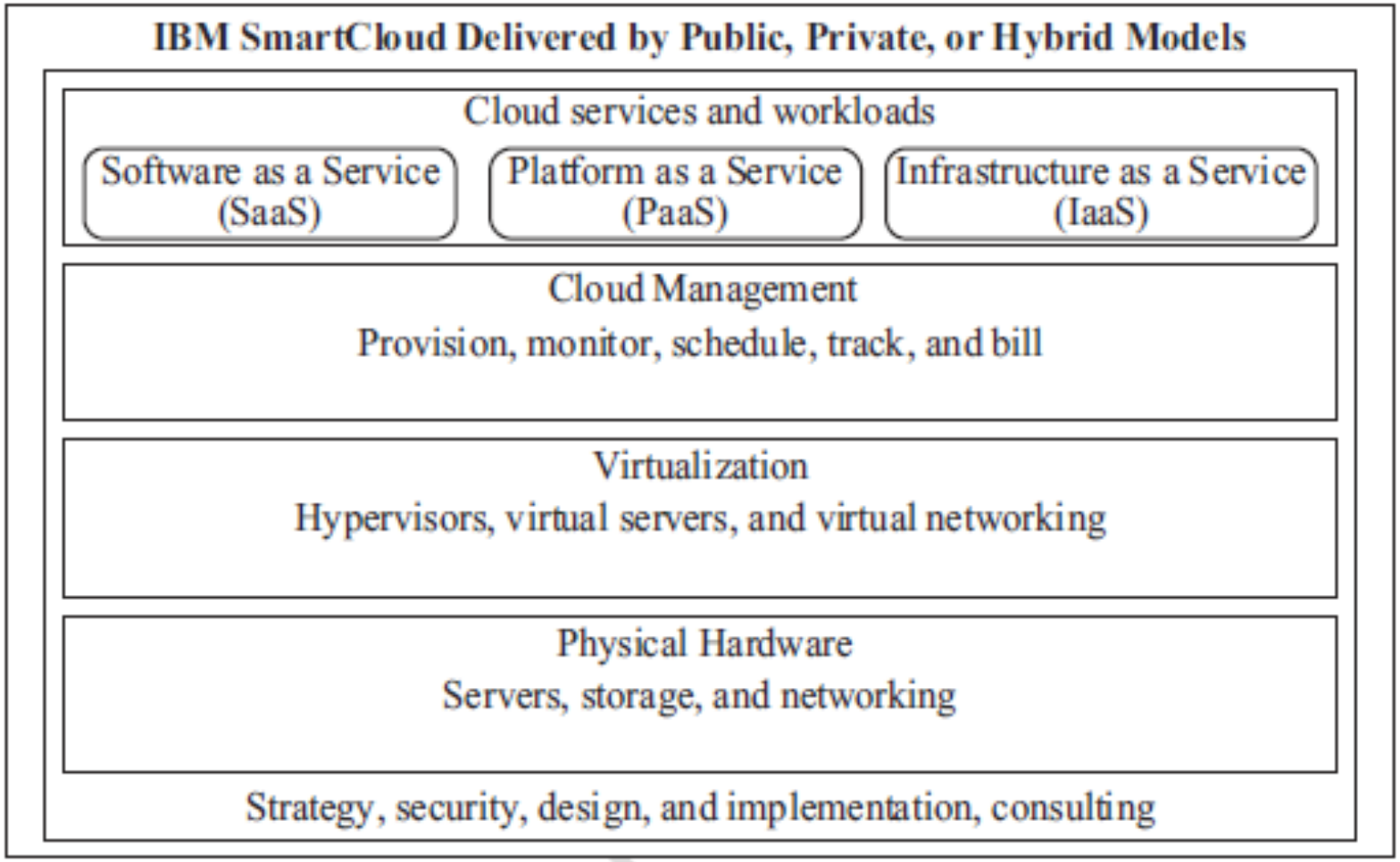


Figure 4.29
Architecture of an IBM SmartCloud offered as public, private, and hybrid platforms based on IBM Tivoli, DB2, WebSphere, and Xen-based virtualization.

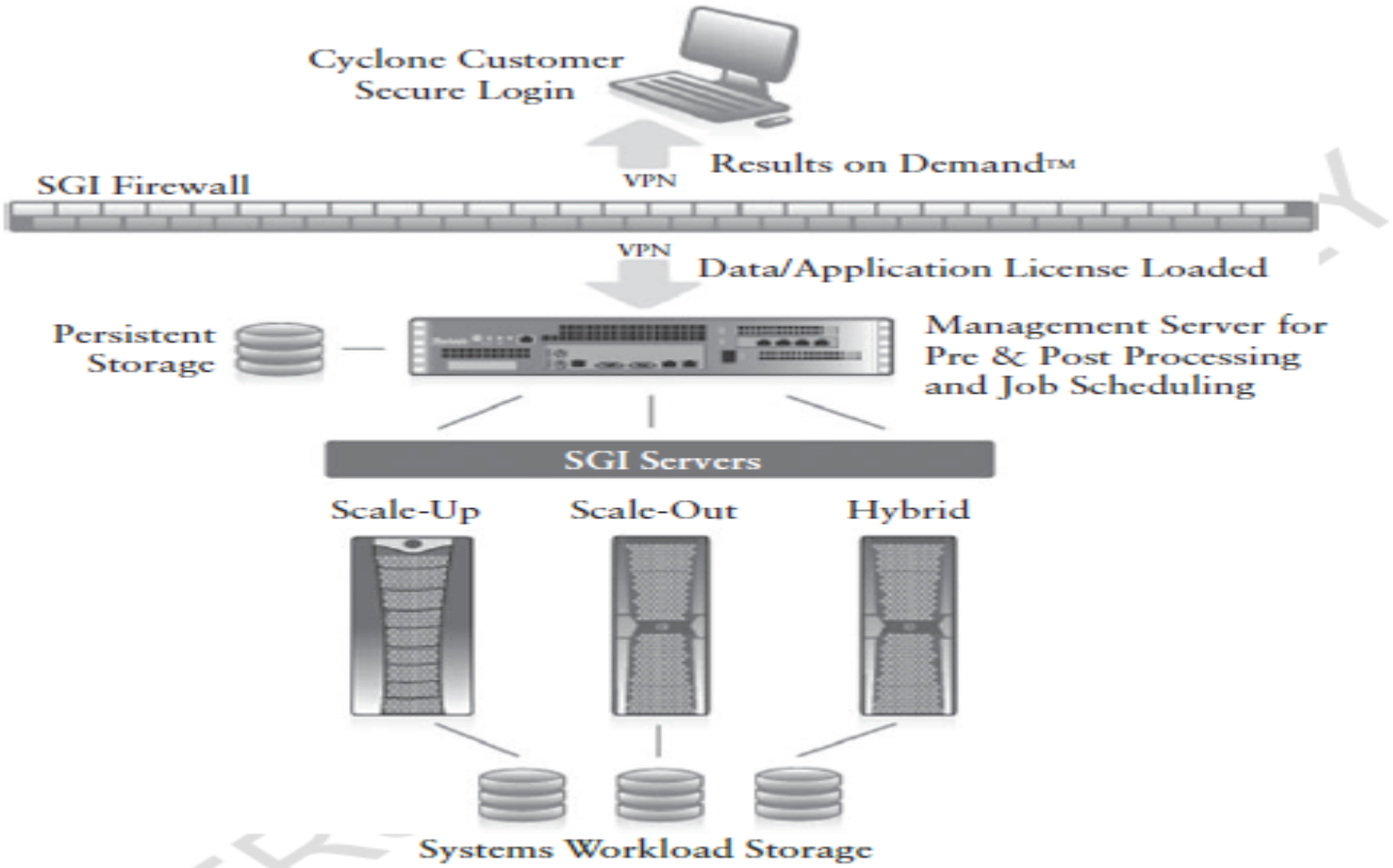


Figure 4.30
SGI Cyclone HPC cloud for enabling SaaS and IaaS applications.

The Nebula Cloud
Built at NASA Ames Research Center

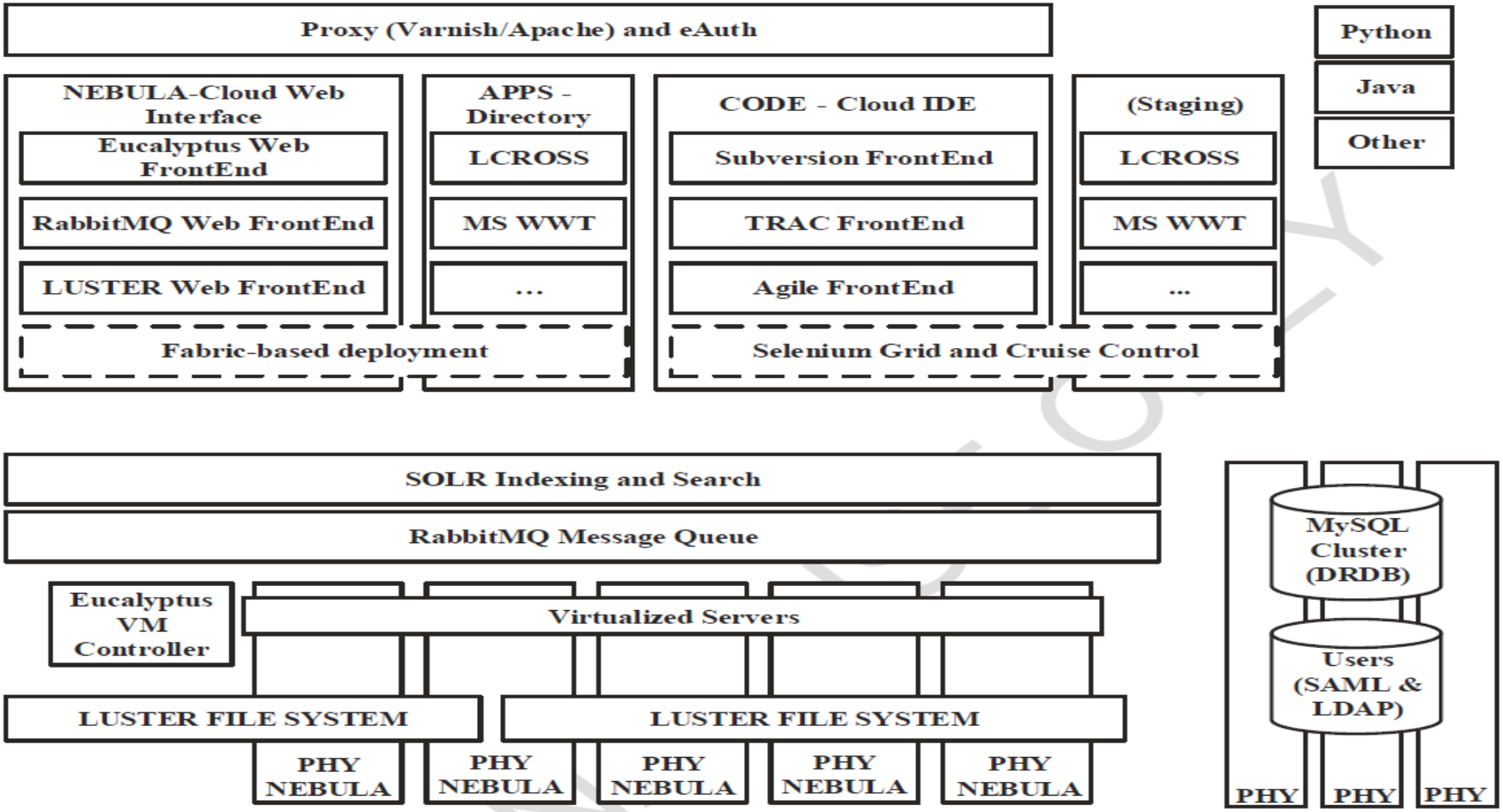


Figure 4.31
The Nebula cloud project at NASA, http://nebula.nasa.gov.

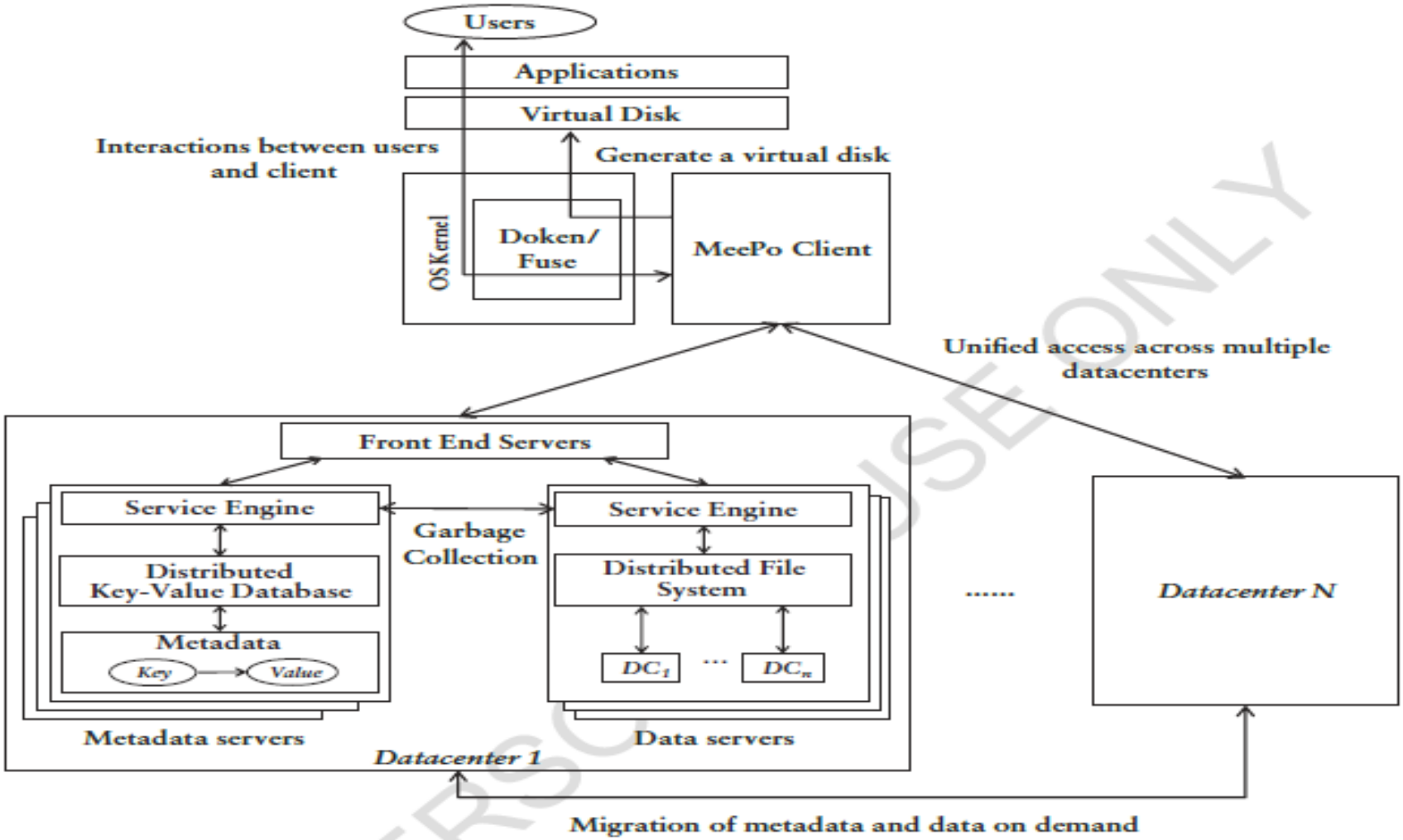


Figure 4.8
MeePo: A community cloud built at Tsinghua University over multiple data centers, where DC stands for data chunks. Courtesy of Wu et al., "Associative Big Data Sharing in Community Clouds—The MeePo Approach," *IEEE Cloud Computing Magazine* (January 2016).

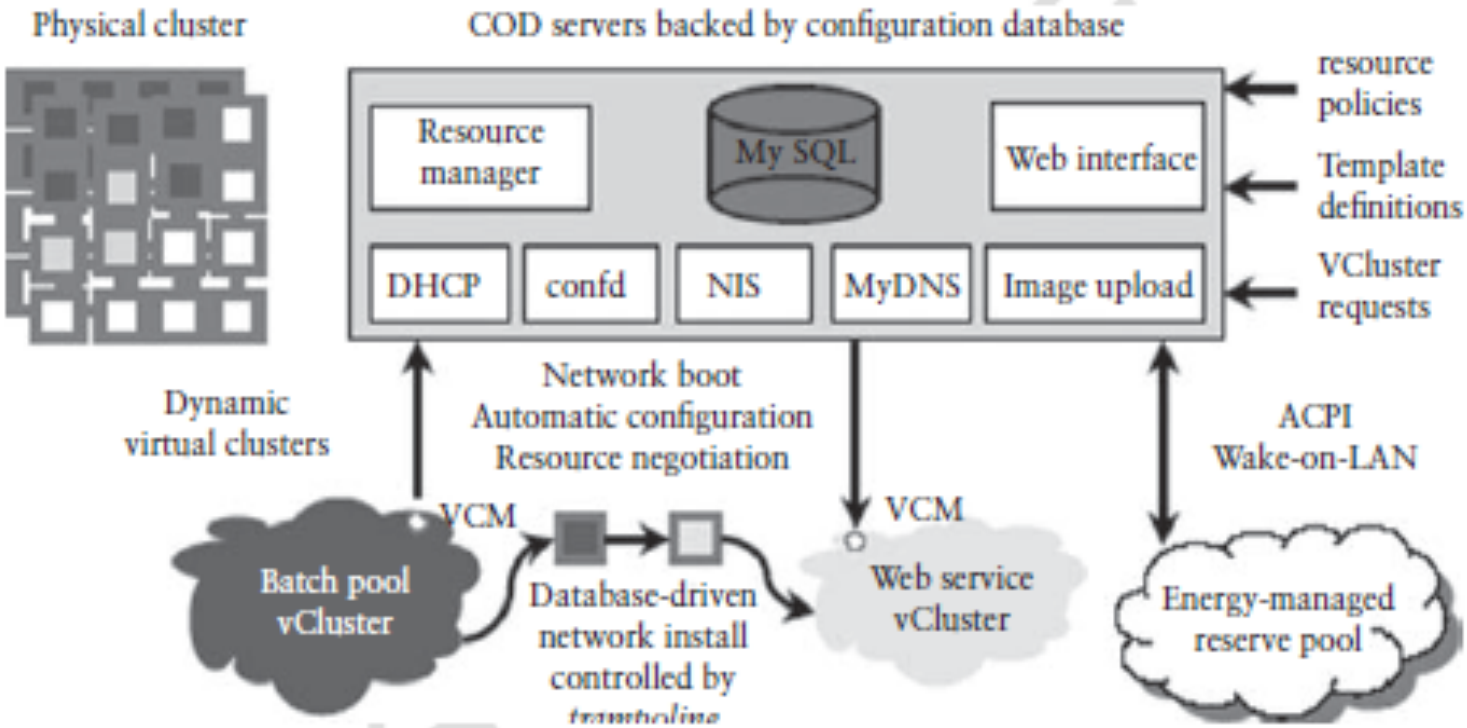


Figure 4.9

Cluster-on-Demand (COD) partitions a physical cluster into multiple virtual clusters. Courtesy of J. Chase, et al., "Dynamic Virtual Clusters in a Grid Site Manager," IEEE 12th Symposium on High-Performance Distributed Computing (HPDC), 2003.

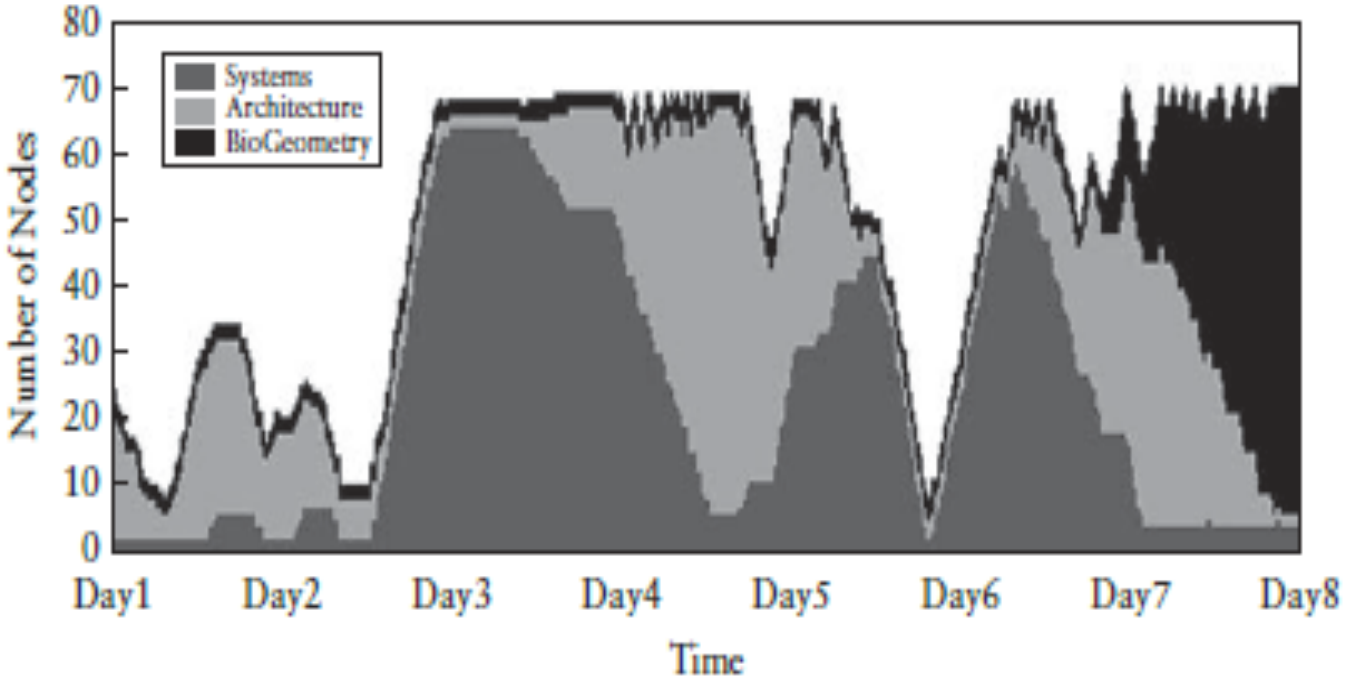


Figure 4.10

The cluster size variation in the COD for eight days at Duke University. Courtesy of J. Chase, et al., "Dynamic Virtual Clusters in a Grid Site Manager," IEEE 12th Symposium on High-Performance Distributed Computing (HPDC), Washington, DC, June 2003.

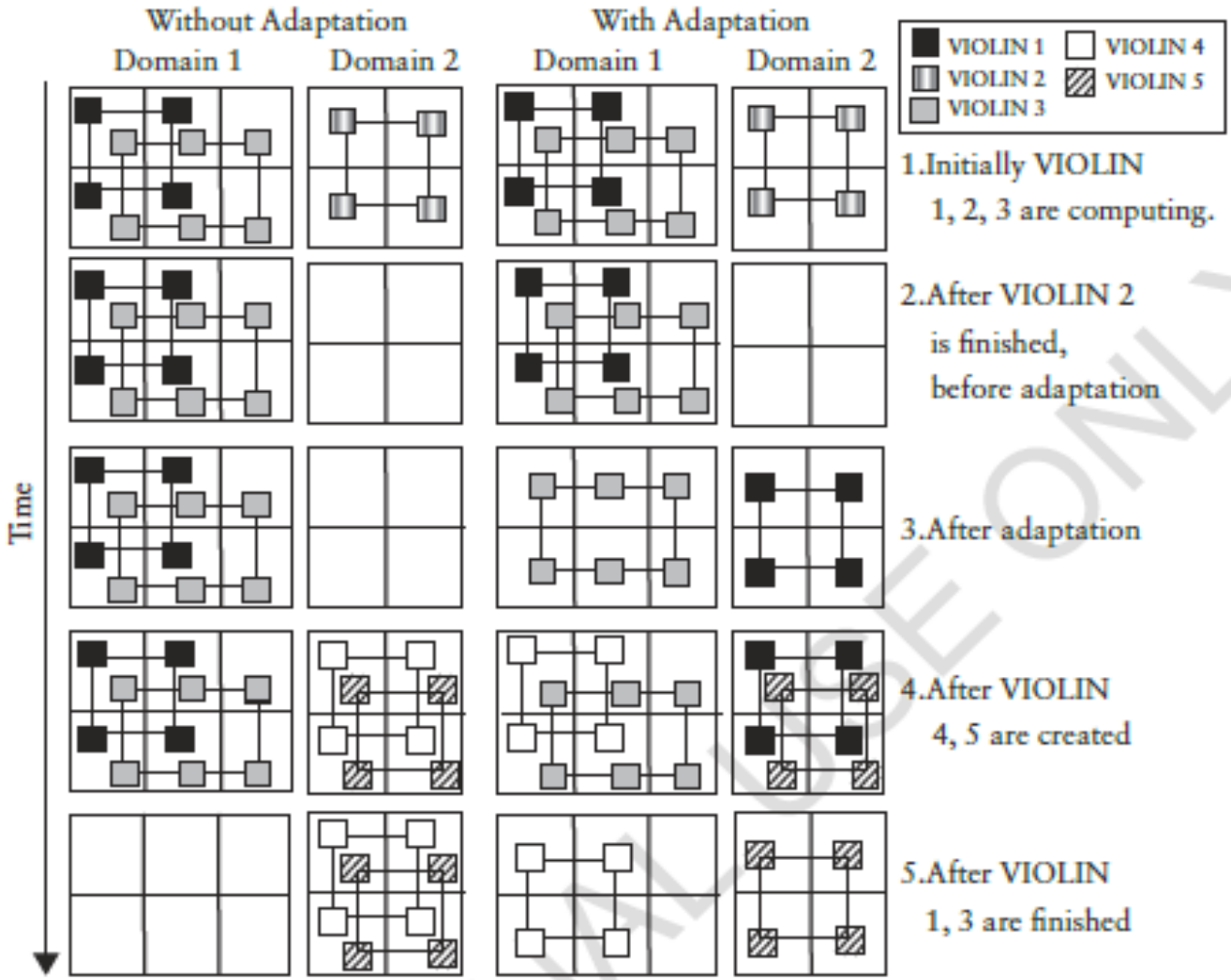
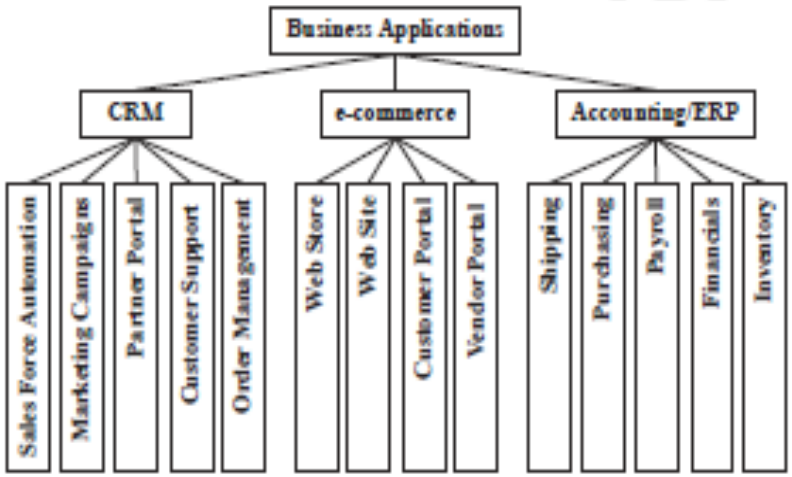
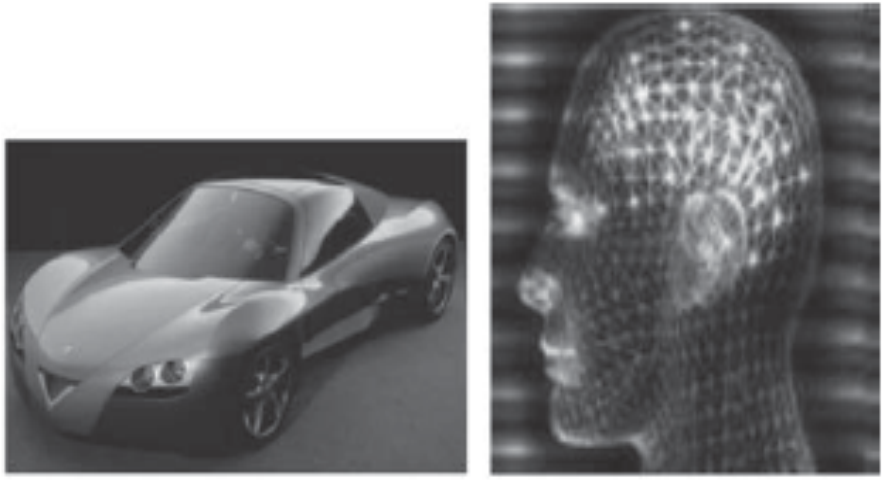


Figure 4.11
VIOLIN adaptation scenario of five virtual clusters on two hosted clusters. There are less idle nodes after the adaptation. Courtesy of P. Ruth et al., "Automatic Live Migration of Virtual Computational Environments in a Multi-Domain Infrastructure," *Technical Report*, Purdue University (2006).

Initial Cloud Applications



(a) Business applications of cloud services



(b) Scientific and technical applicatiuons

(c) Consumer, games, e-mail, mobile and social media applications

Figure 4.19
Typical cloud applications in business, science/technical, and social media areas.

AWS Software, Libraries, and API Support

Resource	Description
AWS SDKs	AWS SDKs include sample code, libraries, tools, documentation, and templates. To download the AWS SDKs, go to AWS Software Development Kits (SDKs).
Libraries	Developers can provide their own libraries, which you can find at the following AWS developer centers: • Java Developer Center • Mobile Developer Center • PHP Developer Center • Python Developer Center • Ruby Developer Center • Windows and .NET Developer Center
Amazon EC2 API	If you prefer, you can code directly to the Amazon EC2 API. For more information, see Making API Requests and Amazon Elastic Compute Cloud API Reference.

Public Data Sets for AWS Experiments

AWS provides a centralized repository of public data sets that can be seamlessly integrated into AWS cloud-based applications. AWS is hosting the public data sets at no charge for the community. Like all AWS services, users pay only for the compute and storage they use for their own applications. Three public Data Sets are below:

Common Crawl Corpus

A corpus of web crawl data composed of over 5 billion web pages. This data set is available on Amazon S3 and is released under the Common Crawl Terms of Use.

1000 Genomes Project

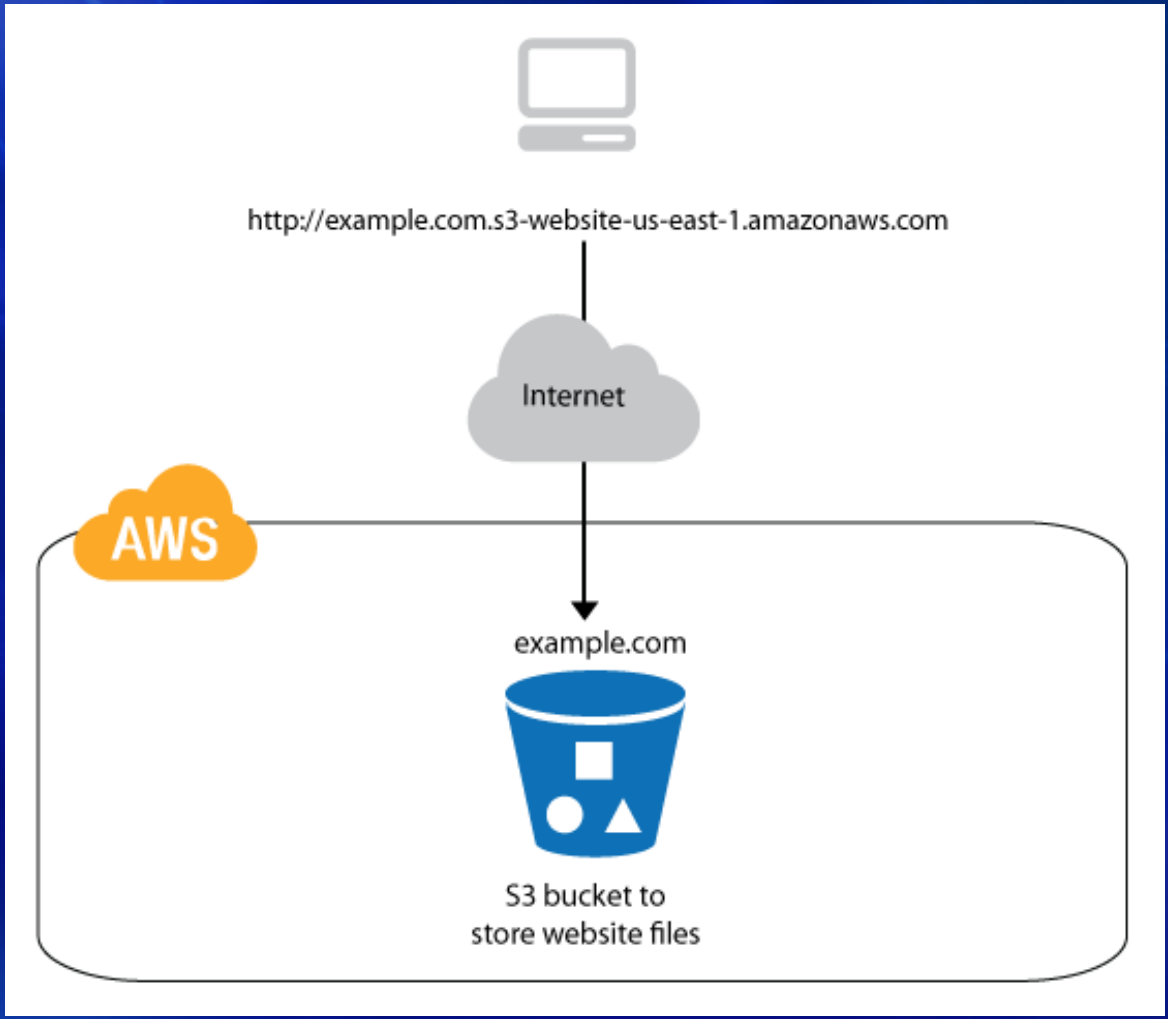
The 1000 Genomes Project, initiated in 2008, is an international public-private consortium that aims to build the most detailed map of human genetic variation available.

Google Books Ngrams

A data set containing Google Books n-gram corpuses. This data set is freely available on Amazon S3 in a Hadoop friendly file format and is licensed under a Creative Commons Attribution 3.0 Unported License. The original dataset is available from http://books.google.com/ngrams/

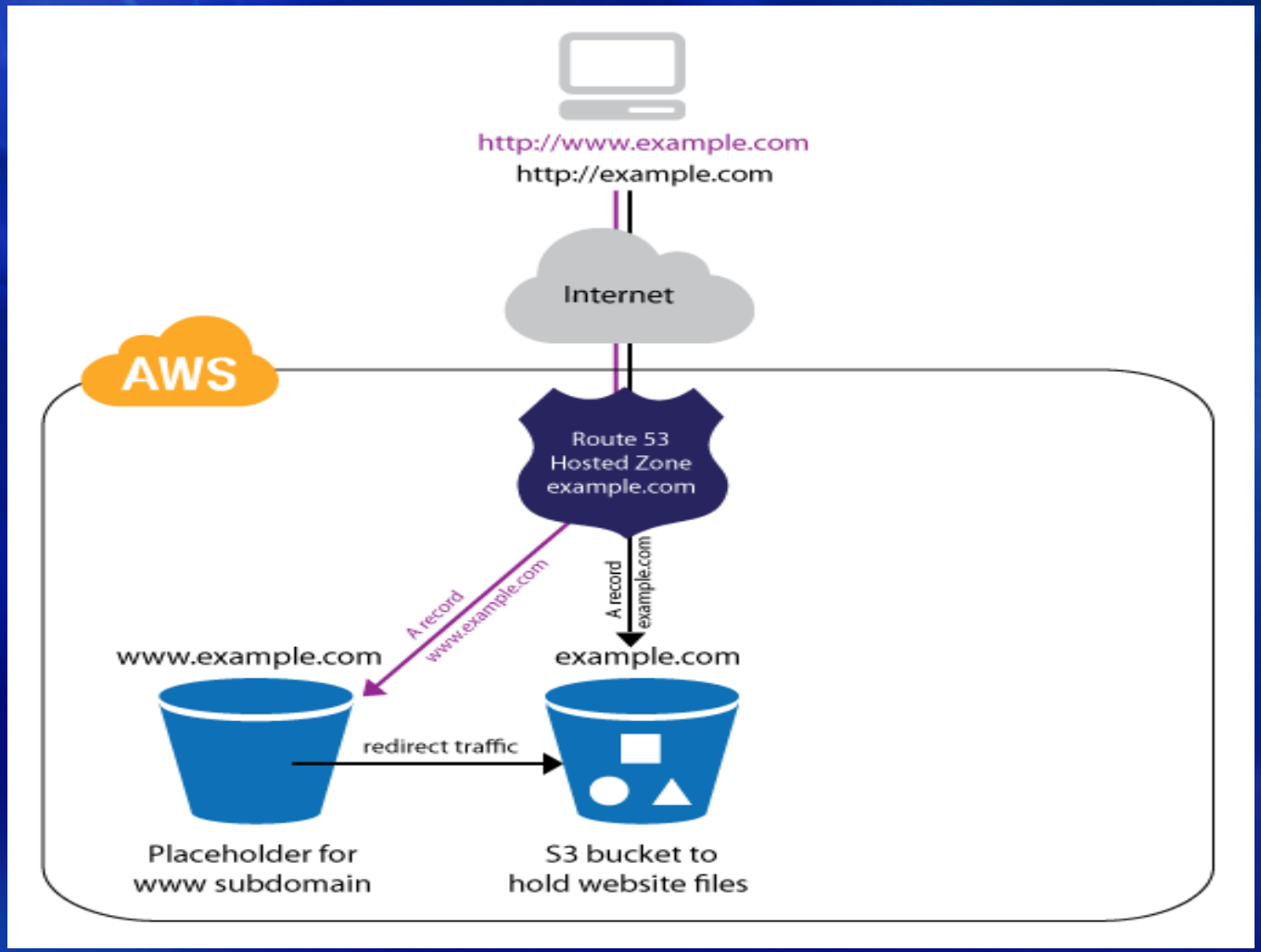
Web Hosting on The AWS

1. S3 to host a static website



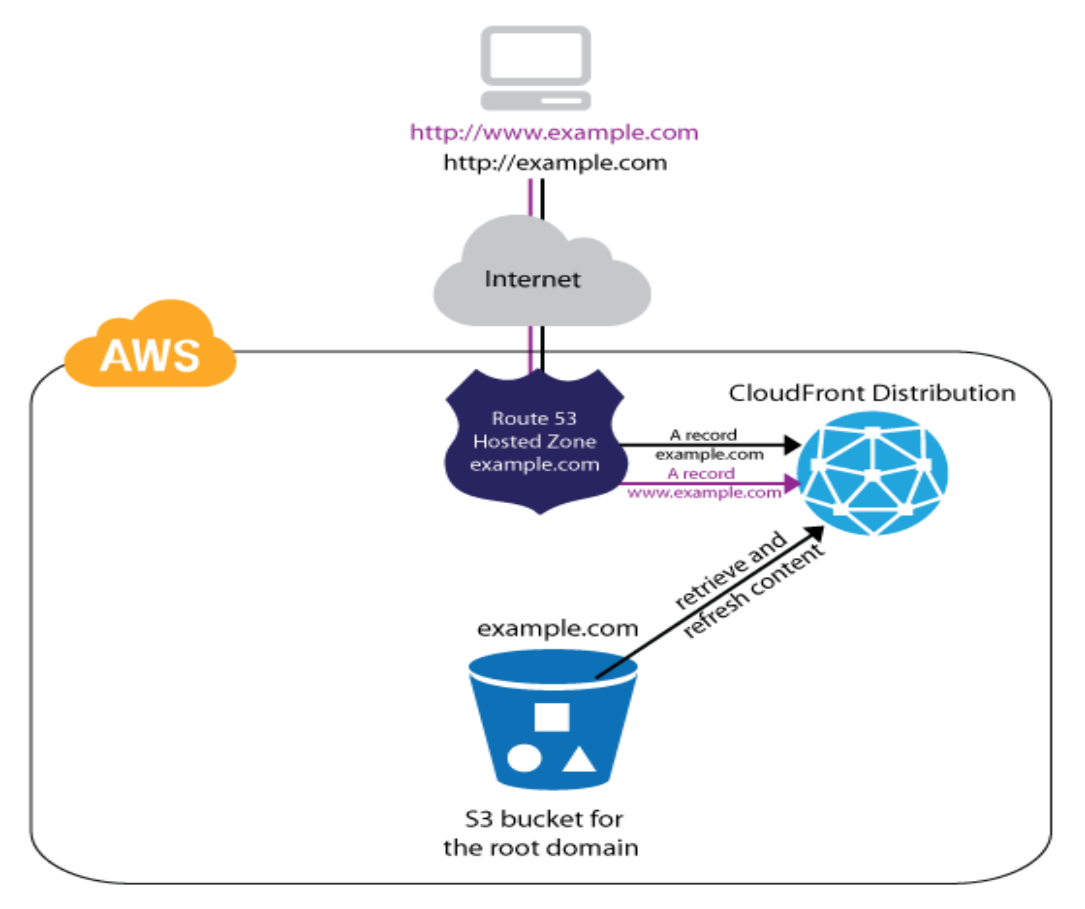
Steps to Create A Static Web Site Hosting on The AWS

2. Route S3 provides custom domain names

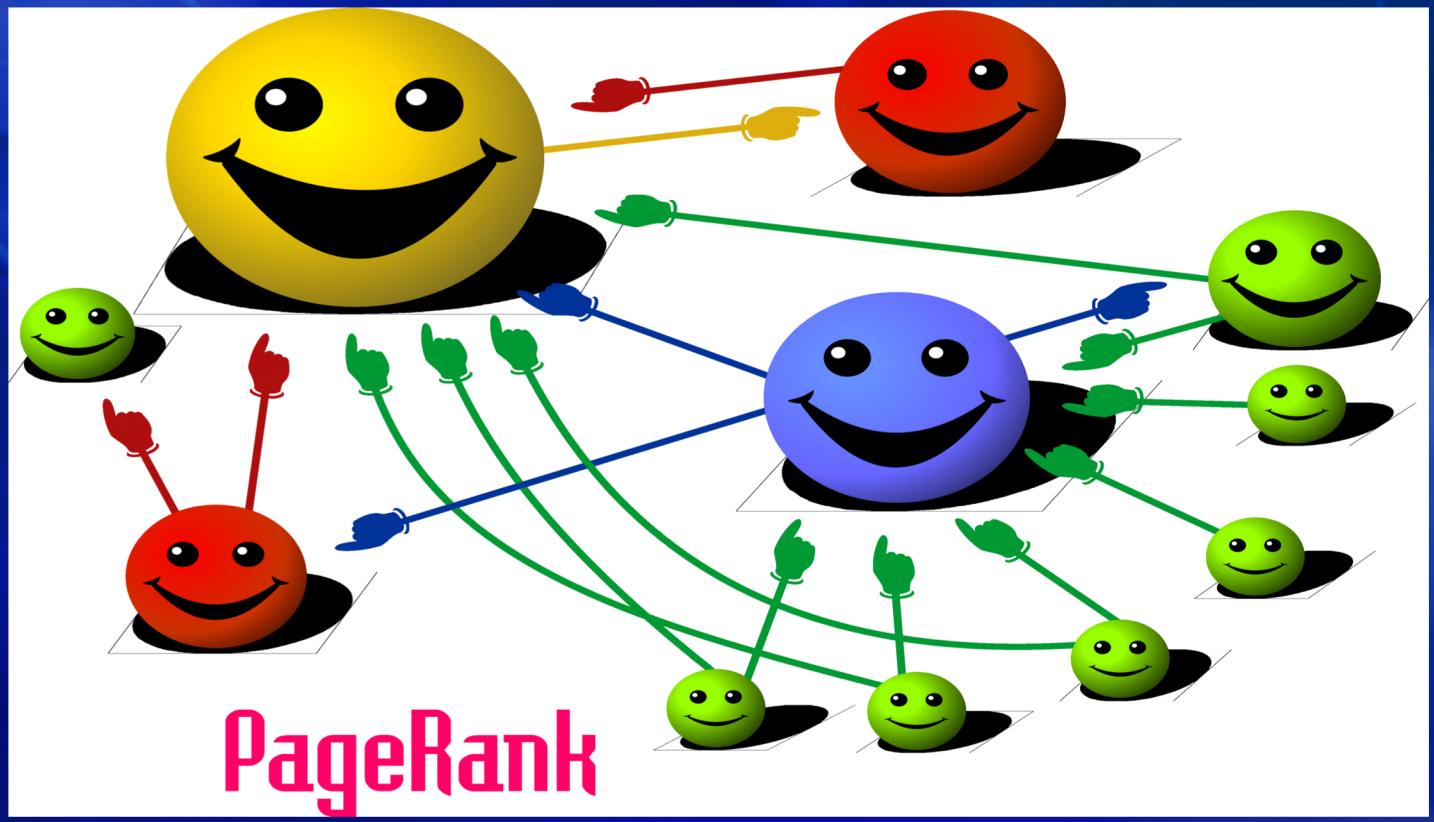


Steps to Create A Static Web Site Hosting on The AWS

3. CloudFront reduces the latency to access the website



PageRank Apps



PageRank for Search Engines and Web Services

- Program implemented by Google to rank any type of recursive "documents" using MapReduce
- Initially developed at Stanford Univ. by Google founders, Larry Page and Sergey Brin, in 1995
- The Google search engine has been extended to support cloud computing in 2007 for Web 2.0 services and beyond

Simulation of a "random-surfer" in the following steps:

- Begins with pair (URL, list-of-URLs)
- Maps to (URL, (PR, list-of-URLs)), For each *u in list-of-URLs*, returns *(u,* PR/|list-of-URLs|), as well as (*u, new-list-of-URLs)*
- Reduce receives (URL, list-of-URLs), and many (URL, value) pairs and calculates (URL, (new-PR, list-of-URLs))

AWS Term Project

HiBench Testing and New ML and IoT Applications on The AWS Cloud

(for 2 - hour lecture)

Project Objectives and Requirements

- Each team proposes one new application topic by extending from those topics suggested below. All cloud app topics are subject to approval before you start.
- We have created a Project Team Formation forum in Discussion Board under *Tools* in blackboard to help you find your teammates.
- You cannot repeat a project report from those done by previous students at USC. Extending from your previous experience is fine, as long as you report original results obtained from new experiments on the AWS cloud.
- Feel free to use any of the available SDK tools, benchmark, existing app codes, or data sets you can find in open sources. All borrowed codes, tools, data sets must identify the sources of information explicitly in the Project Report.

Important AWS Links To Visit and Documented Reports/Articles To Read

1. The main AWS site: http://aws.amazon.com
2. Just use http://aws.amazon.com/ec2/ to access ec2, or http://aws.amazon.com/s3/ for S3, or …../sqs/, …../sns/, or …./simpledb/, ……/sdk/, ……/fps/, etc.
3. Many PDF reports and articles, application examples, SDK tools, etc. can be found in the above web sites. You do not need an account to access these sites.
4. You need to establish your personal accounts at AWS, if you want to start using the AWS to do homeworks and this team project.

Some AWS Services You May Use

Category	Service	Description
Compute	EC2	EC2)provides resizable compute capacity in the cloud.
	Container Service	Manage Docker containers across a cluster of Amazon EC2 instances
	Elastic Beanstalk	An application container for deploying and managing applications.
	Lambda	AWS Lambda responds to events and manages the compute resources needed
Analytics Service	EMR	Apache Spark on Hadoop YARN is natively supported in Amazon EMR
	Data Pipeline:	Orchestration for Data-Driven Workflows
	Kinesis:	Work with Real-Time Streaming Data
	Machine Learning:	A managed service for building ML models and generating predictions or classifications
Database	RDS	RDS makes it easy to set up, operate, and scale the relational databases in the cloud.
	DynamoDB	This is a scalable NoSQL data store that manages distributed replicas of your data
	ElastiCache	This improves application performance by using an in-memory caching system.
	Redshift	Amazon Redshift is a fast, fully managed, petabyte-scale data warehouse
Mobile Services	Mobile Hub:	AWS Mobile Hub lets you quickly build, test, and monitor usage of your mobile apps.
	Cognito:	This provides user identity and data synchronization service across mobile devices.
	Device Farm:	AWS Device Farm helps you improve the quality of your Android, iOS and web apps .
	Mobile Analytics:	This service lets you easily collect, visualize, and understand app usage data at scale.
	SNS	A fast, flexible, fully managed push notification service .
IoT	AWS IoT	AWS IoT connected devices interact with cloud applications and other devices.
Storage Services	S3 Storage Service	S3 buckets are used to store and retrieve any amount of data as data objects .
	CloudFront	This distribute content to end users with low latency and high data transfer speeds.
	Elastic File System	A file storage service for Amazon Elastic Compute Cloud (Amazon EC2) instances.
	Glacier	Provides secure and durable storage for data archiving and backup.
	Snowball	This accelerates moving large data into and out of AWS using secure appliances

AWS System Setup

- Setup JDK, Hadoop-YARN, Spark runtime environment
- Download/checkout HiBench benchmark suite
- Run <HiBench_Root>/bin/build-all.sh to build HiBench
- Begin from HiBench V4.0, HiBench needs Python 2.x(>=2.6)

For minimum requirements: create & edit conf/99-user_defined_properties.conf :

- cd conf
- cp 99-user_defined_properties.conf.template 99-
- user_defined_properties.conf

Intel HiBench Benchmark for Testing Clouds

- HiBench is a benchmark specifically tailored for running Hadoop programs based on MapReduce paradigm. The suite was developed at Intel for measuring the speed, throughput, HDFS bandwidth, and resources utilization in sort, word count, page ranking, Bayesian classifier, and distributed I/O workload. https://github.com/intel-hadoop/HiBench

- Hadoop programs in HiBench are:

1. Sort 2. WordCount 3. TeraSort 4. Enhanced DFSIO
5. Nutch indexing 6. PageRank 7. Bayesian classification
8. K-means clustering 9. Hive Query

Reference Paper: Huang, S., Huang, J., Dai, J., and Xie, T., and Hong, B., "The HiBench Benchmark Suite: Characterization of The MapReduce-based Data Analysis, *Int'l Conf. on Data Engineering Workshops,* March 2010.

HiBench Component Programs

- Sort (sort): This workload sorts a text input data randomly generated using the TexWriter
- WordCount (wordcount):This workload counts the occurrence of of each word in the input text data using the TexWriter
- TeraSort (terasort): This is standard benchmark generated by Hadoop TeraGen program
- Sleep (sleep): This workload sleep an amount of seconds in each task to test framework scheduler
- Scan (scan), Join (join), Aggregate (aggregation): These workloads are for SQL query processing. It contains 5 Hive queries performing the typical OLAP quieres. Its input is automatically generated Web data

Example of WordCount in a Python Program on Amazon EMR

In this part, you will run a WordCount program using AWS EMR. The WordCount Python code has been uploaded to an Amazon S3 bucket (s3://elasticmapreduce/samples/wordcount/wordSplitter.py), and you can check out the source code here (https://aws.amazon.com/articles/Elastic-MapReduce/2273). This WordCount Python code is included in Amazon EMR.

The input dataset is uploaded to an Amazon S3 bucket: s3://elasticmapreduce/samples/wordcount/input. The way you use the Python code and input data is to set up their reference in the jobflow configuration . The WordCount counts the number of occurrences of each word in a given input document set.

A MapReduce sorts the outputs of the maps and generate the final counts using reduce tasks. The MapReduce framework operates exclusively on <key, value> pairs, that is, the framework views the input to the job as a set of <key, value> pairs and produces a set of <key, value> pairs as the output.

HiBench Component Programs

- **PageRank (pagerank):** This workload benchmarks PageRank algorithm implemented in Spark-MLLib/Hadoop examples. The data source is generated from Web data
- **Nutch indexing (nutchindexing):** Large-scale search indexing using MapReduce. It tests the index subsystem in Nutch serach engine
- **Bayesian classification (bayes):** This workload benchmarks Naïve Bayesian classification algorithm implemented in Spark-MLLib/Mahout examples for machine learning, data mining and knowledge discovery
- **K-means clustering (kmeans):** This tests the K-means clustering algorithm for knowledge discovery and data mining in Mahout 0.7/Spark MLLib
- **Enhanced DFSIO (dfsioe):** This tests the HDFS throughput of the Hadoop cluster by generating a large number of tasks performing writes and reads simultaneously. It measures the average I/O rate of each Map task and the aggregated throughput of HDFS cluster

AWS HiBench System Setup

- HiBench contains a set of Hadoop, Spark and streaming workloads
- Setup JDK, Hadoop-YARN, Spark runtime environment
- Download/checkout HiBench benchmark suite(https://github.com/intel-hadoop/HiBench)
- Run <HiBench_Root>/mvn -Dspark=2.1 -Dscala=2.11 clean package build HiBench
- Begin from HiBench V4.0, HiBench needs Python 2.x(>=2.6)

For HadoopBench Configuration: Configure hadoop.conf:
conf/hadoop.conf :

- cd conf
- cp conf/hadoop.conf.template conf/hadoop.conf

Suggested Topics for AWS App Design (1)

Topic 1: Web or mashup services hosted on AWS

By utilizing AWS Elastic Beanstalk, you could run your own web service in AWS. Developers can simply upload their application code and the service automatically handles all the details such as resource provisioning, load balancing, auto-scaling, and monitoring.

Elastic Beanstalk is ideal if you have a standard PHP, Java, Python, Ruby, Node.js, .NET, Go, or Docker application that can run on an app server with a database. Elastic Beanstalk uses Auto Scaling and Elastic Load Balancing supports highly variable amounts of traffic.

You can start small and scale up. For example, you could develop a benchmark web service like TPC-W http://www.tpc.org/tpcw/default.asp, which is a transactional web e-Commerce benchmark.

Suggested Topics for New AWS App Design (2)

Topic 2: Social group and community detection on the Cloud

Social media analytics are used to identify social groups and detect social communities. Some public datasets are given here: www.kdnuggets.com/2014/08/interesting-social-media-datasets.html.

A good survey by Xie, et al on *"Overlapping Community Detection in Dynamic Networks"* (http://www.cs.rpi.edu/~szymansk/papers/acm-cs.13.pdf)

Suggested Topics for AWS App Design (3)

Topic 3: Recommender System on AWS

You build a recommendation system like a music recommendation system. An example dataset is found here: http://www-etud.iro.umontreal.ca/~bergstrj/audioscrobbler_data.html

You need to search for algorithms such as the "Alternating Least Squares Recommender Algorithm" (https://mahout.apache.org/users/recommender/intro-als-hadoop.html).

You may build other systems for restaurant recommendation, job recommendation, apartment recommendation near USC, etc. This was a popular topic selected by many teams in the past.

Suggested Topics for New AWS App Design (4)

Topic 4: Cloud Security and Big Data Privacy Protection

For cloud security and data privacy protection, an example could be anomaly detection in the network traffic flowing in and out of a cloud system. Example KDD1999 data set from:
http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

Other datasets are summarized here:
https://www.quora.com/Where-can-I-get-the-latest-dataset-for-a-network-intrusion-detection-system

Suggested Topics for AWS App Design (5)

Topic 5: Specific Cloud IoT Apps on AWS.

The AWS IoT is a managed cloud platform that lets connected devices -- cars, light bulbs, sensor grids and more -- easily and securely interact with cloud applications and other devices.

- **Connect and manage your devices:** Connect devices to the cloud using the protocols HTTP, MQTT, or WebSockets. Devices can communicate with each other even if they are using different protocols
- **Process and act upon device data:** Filter, transform and act upon data from devices on the fly, based on business rules. AWS IoT applies Amazon DynamoDB, Kinesis, Machine Learning, and Lambda
- **Read and set device state at any time:** AWS IoT stores the latest device state that is read or set anytime. You can read or set a device's state even when the device is offline; or filter, transform and act upon data on the fly based on business rules

Suggested Topics for New AWS App Design (6)

Topic 6: Cloud-based Business and Financing App:

This topic deals with banking business and finance matters. For example, you may want to calculate the *Value at Risk* (VaR) for the stock market (https://en.wikipedia.org/wiki/Value_at_risk) through historical stock values and variations. You could get the stock data using Yahoo Finance API to get the CSV format data via the link: http://www.jarloo.com/yahoo_finance/

Topic 7: Cloud-based Healthcare and Medical App:

Machine learning models and algorithms can help analyze MRI images and medical prescription for predicting drug usage and continuity, feature extraction in medical records. They can be applied in brain tumoils, and cancer detection, etc. Health-care clouds are pursued heavily by IT industry, including the IBM Watson Project and the DeepMind Project at Google. They are used in preventing, detecting and recovery from physical diseases, but also on solving some mental disorder problems in emotion detection, suicide prevention, etc.

Suggested Topics for New AWS App Design (7)

***Topic 8: Genomics Big Data Applications on Clouds*:**

You could use ADAM which is a genomics analysis platform with specialized file formats built using Apache Avro, Apache Spark and Parquet. Source code : https://github.com/bigdatagenomics/adam

More info from the *Big Data Genomics* project: http://bdgenomics.org/ These are related to biological DNA sequence analysis.

Both genetic engineering and the drug industry are heavy in these area. You may need to process huge medical records using some data mining and machine learning techniques.

Example: Hosting an information sharing website on the AWS Cloud

- Your cloud service supports the storage, indexing, searching, classification, notification, replication, and secure sharing by friends, classmates, family members, etc. This system will enable the sharing and backup of personal or professional documents, photos, music and video services among trusted friends or peers.
- You will use the EC2 VM instances (or containers), S3 storage, CloudWatch, and notification (SNS) services. Some of the benchmark running experiments in Part 1 may help you develop this service system on AWS cloud.
- First implement the photo-service system on a personal computer (such as a notebook). Then explore the AWS cloud resources to upgrade the scope of services and the QoS including performance, privacy, protection, etc.
- Compare the relative performance of the local PC service and remote cloud services in terms of latency, response time, scalability, availability, and cost-effectiveness, etc.

Suggested Topics for New AWS App Design (8)

Topic 9: Machine Learning and Cognitive Apps

Speech and image understanding and computer vision and natural language processing are big part of today's AI and cognitive service industry. The Google Brain Team and X-Lab projects and many similar projects at Facebook, Microsoft, IBM, are devoted to these areas. Chapters 2, 6 , 7 and 9 are related to this topic. Some of the working examples in the text and homework problems are also relevant.

Topic 10: Cloud-Centric Mobile Applications

Mobile applications are mostly supported by public clouds like Apple's iCloud, etc. Wikipedia has a lot of coverage in these areas. The 5G mobile core network could be using cloud-controlled base stations. Many AWS services support mobile, IoT and smart machine applications. Mobile devices and pervasive computing cannot be separated.

Suggested Topics for New AWS App Design (9)

Topic 11: Fusion of SMACT in Specific AI Apps

SMACT technologies involve social networks, mobile systems, data analytics, clouds and IoT as introduced in Chapter 1. You could propose to use some social-media networks to collect big data and use AWS software to perform some machine learning and IoT sensing applications. Google, Facebook, Microsoft, IBM are all investing heavily in these areas. This would be a meaningful consolidation project for your team to integrate available technologies and put what you have learned in this course into real-life practice.

Topic 12: Geographical Information Services (GIS) on Clouds

GIS systems deal with big data collected by government and social service agencies. This has to do with weather, agriculture, transportation, travel agencies, express delivery services, etc. Remote sensing data from space is relevant to these services. Porting GIS on the cloud makes it easy for massive users to access the publically shared data. Go to Wikipedia to dig out more recent development on cloud-assisted GIS systems.

Homework problems and text examples that are relevant to your Term Project:

- Some of the listed topics are related or inspired by the following homework problems in Hwang's Text:
- Problems 1.4, 2.7, 2.8, 2.9, 2.13, 2.15, 3.13, 4.3, 4.5, 4.7, 4.11, 5.3, 5.4 5.5, 5.7, 5.8, 5.11, 6.5, 6.8 ~6.10, 7.3, 7.5, 8.3 ~ 8.8, 8.11 ~ 8.14, 9.1, 9.2, and 10.2~ 10.10. It is acceptable if you plan to use some of the homework codes and measured results
- You may review the following examples in Hwang's book:
- Examples 2.1 ~ 2.6, 3.6, 4.2 , 4.3, 4.8, 4.11, 4.12, 5.4, 5.5, 6.2, 6.3, 6.6, 7.3, 8.3, 8.5, 8.6, 8.11, 8.12, 9.7, 9.11~9.13, and 10.1 ~ 10.7. These examples may inspire you with some extended new cloud applications to be built in the project.

Technical Report for Your AWS Cloud Project

using the IEEE Conference Paper Format

1. Project Title must hit a hot topic - short, clear and eye-catching, authors (team members) and email contacts included. (5 %)
2. Executive Summary must state the project objectives, summarize technical findings and innovative contributions. (15%)
3. AWS Experiments Specification: Technically specify AWS experiments performed, application designs, AWS services applied, and analytical formulation with schematic diagram, flowchart, or algorithms, etc. (30%)
4. Experimental Settings: Explain the cloud hardware (VM instances), software tools, data sets, application code used or developed, and performance metrics used or defined and measured. (20%)
5. Professionally report performance results obtained, technical findings with scientific plots (figures) or tabulations, analysis, discussions, and conclusions with suggestions. (30%)