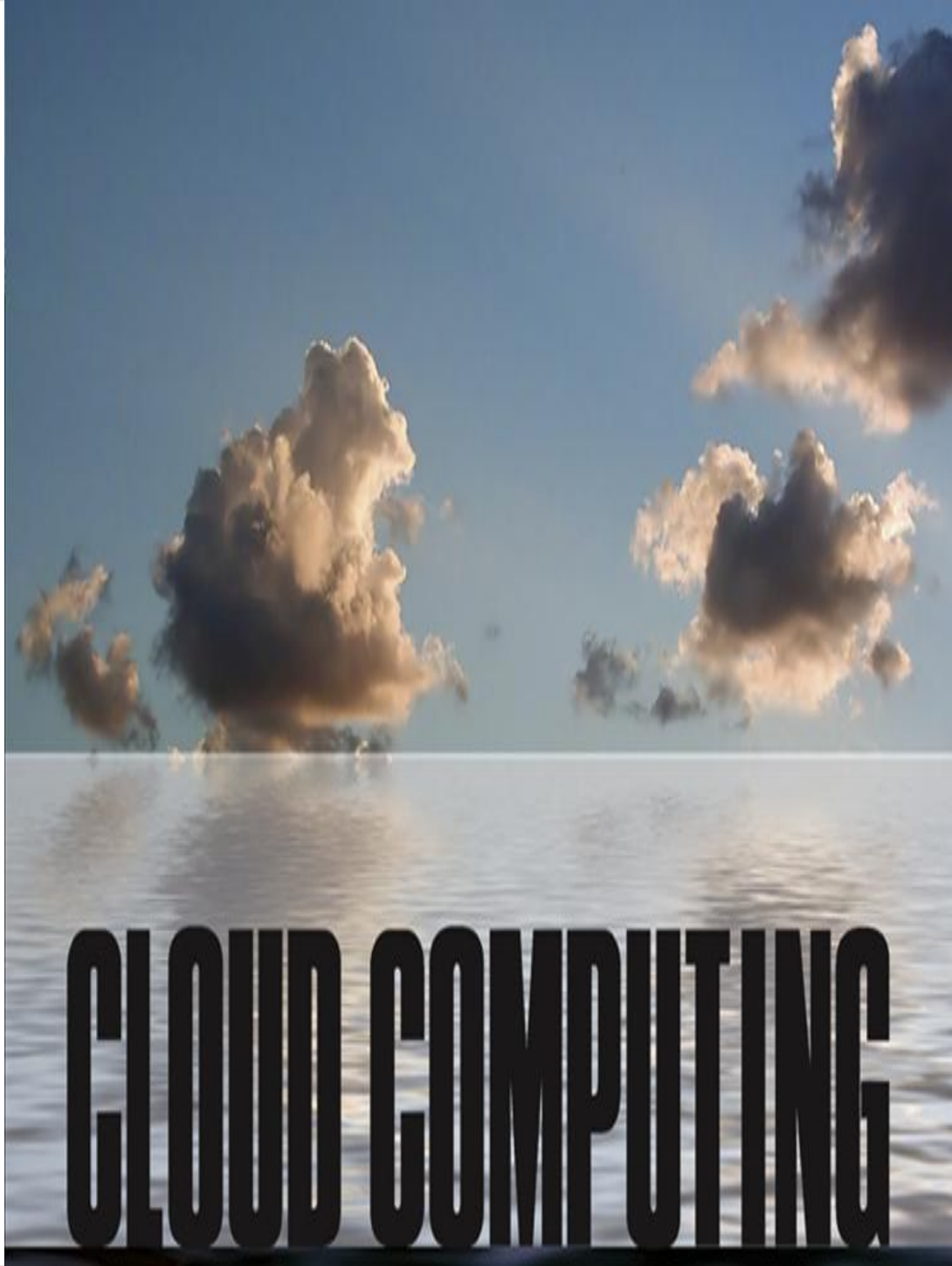
CLOUD COMPUTING AND SECURITY (BCS601)



	CLOUD COMPUTING		Semester	6
	Course Code	BCS601	CIE Marks	50
	Teaching Hours/Week (L: T:P: S)	3:0:2:0	SEE Marks	50
	Total Hours of Pedagogy	40	Total Marks	100
	Credits	04	Exam Hou3rs	3
	Examination type (SEE)	Theory/Practical		

Course objectives:

- Introduce the rationale behind the cloud computing revolution and the business drivers
- Understand various models, types and challenges of cloud computing
- Understand the design of cloud native applications, the necessary tools and the design tradeoffs.
- Realize the importance of Cloud Virtualization, Abstraction`s, Enabling Technologies and cloud security

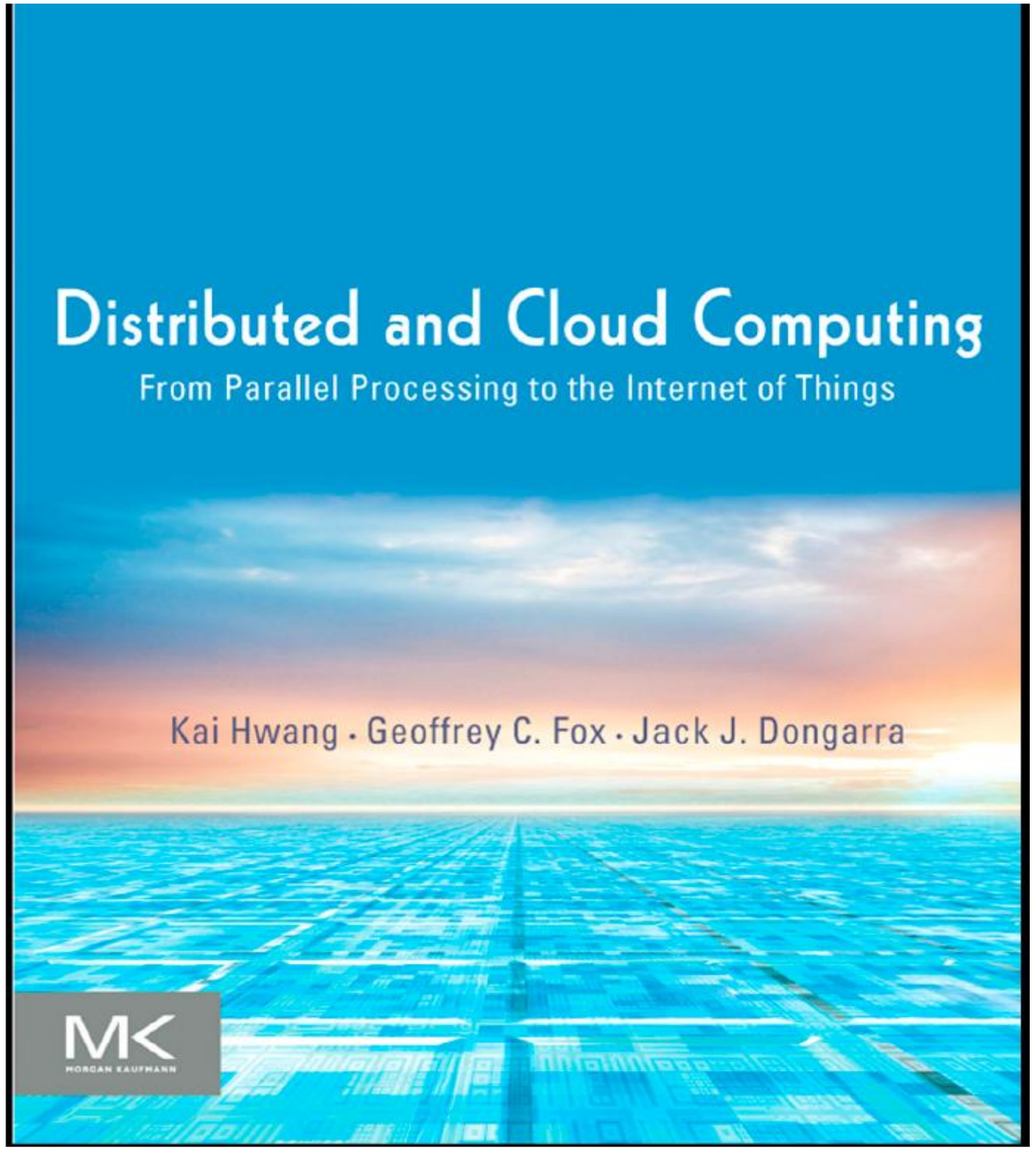
Course outcome (Course Skill Set)

At the end of the course, the student will be able to:

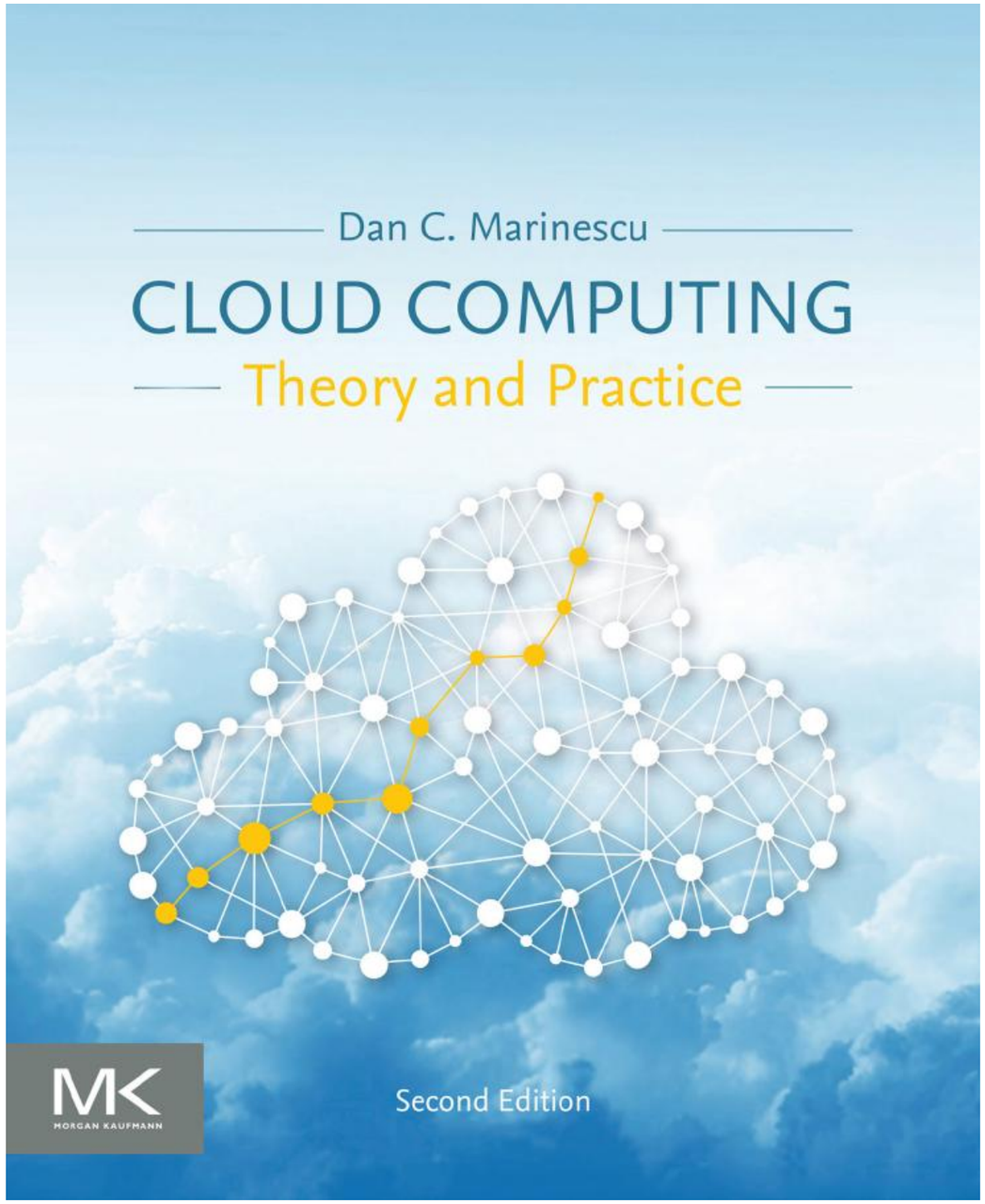
1. Describe various cloud computing platforms and service providers.
2. Illustrate the significance of various types of virtualization.
3. Identify the architecture, delivery models and industrial platforms for cloud computing based applications.
4. Analyze the role of security aspects in cloud computing.
5. Demonstrate cloud applications in various fields using suitable cloud platforms.

TEXT BOOKS

MODULE 1 to MODULE 5



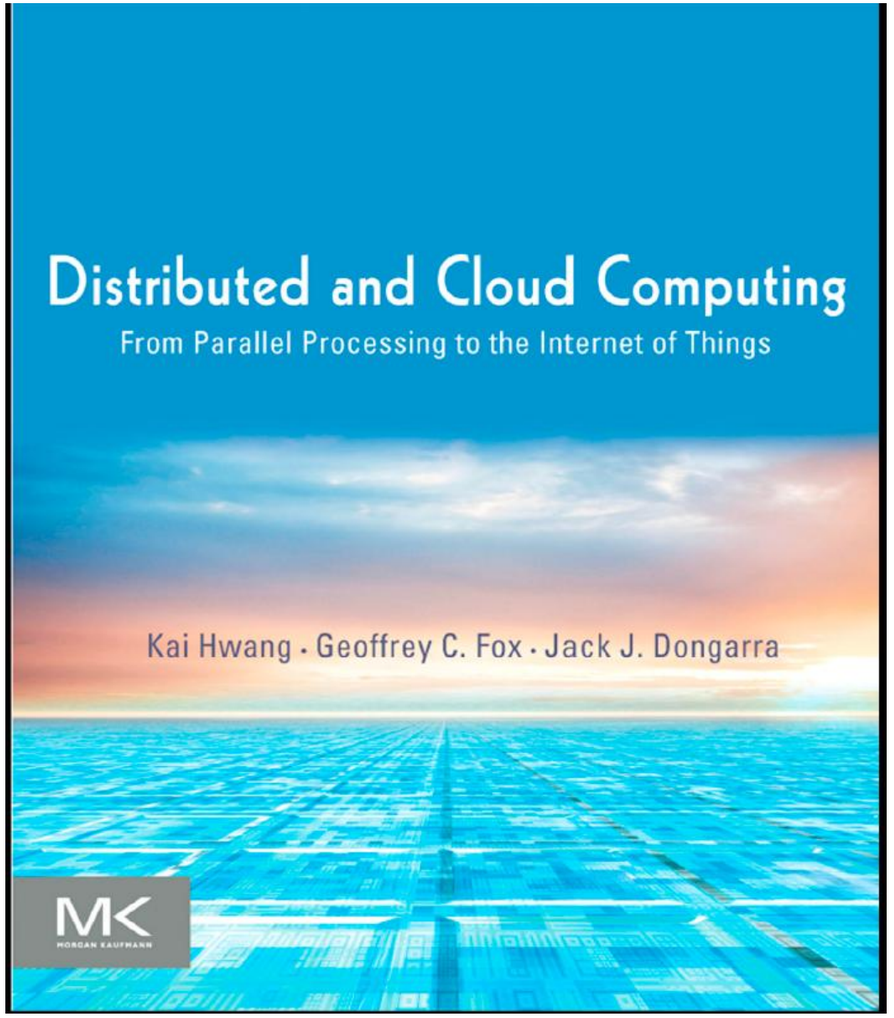
MODULE 4



Module-3

Cloud Platform Architecture over Virtualized Datacenters: Cloud Computing and Service Models, Data Center Design and Interconnection Networks, Architectural Design of Compute and Storage Clouds, Public Cloud Platforms: GAE, AWS and Azure, Inter-Cloud Resource Management.

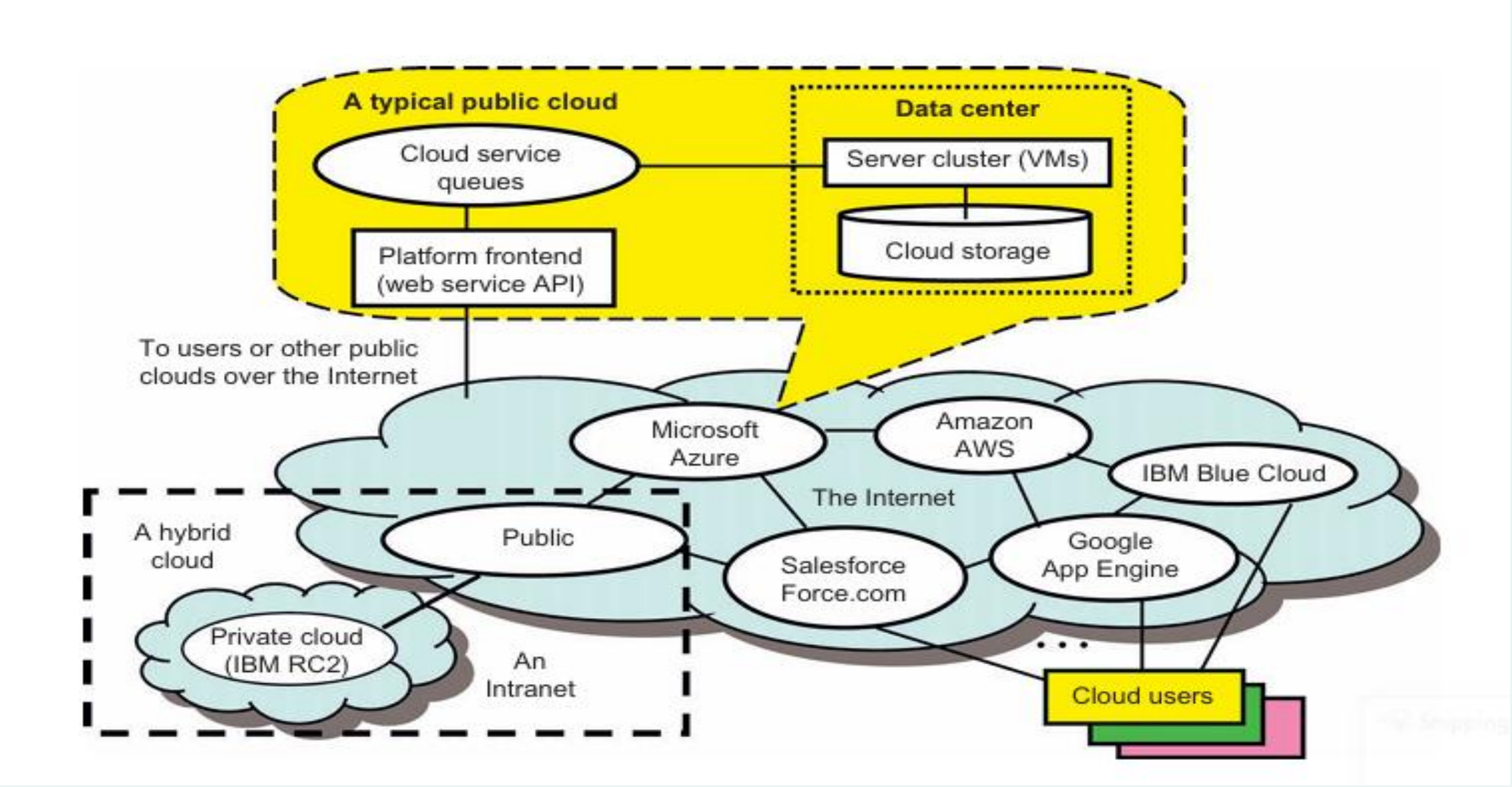
Textbook 1: Chapter 4: 4.1 to 4.5



CLOUD COMPUTING AND SERVICE MODELS

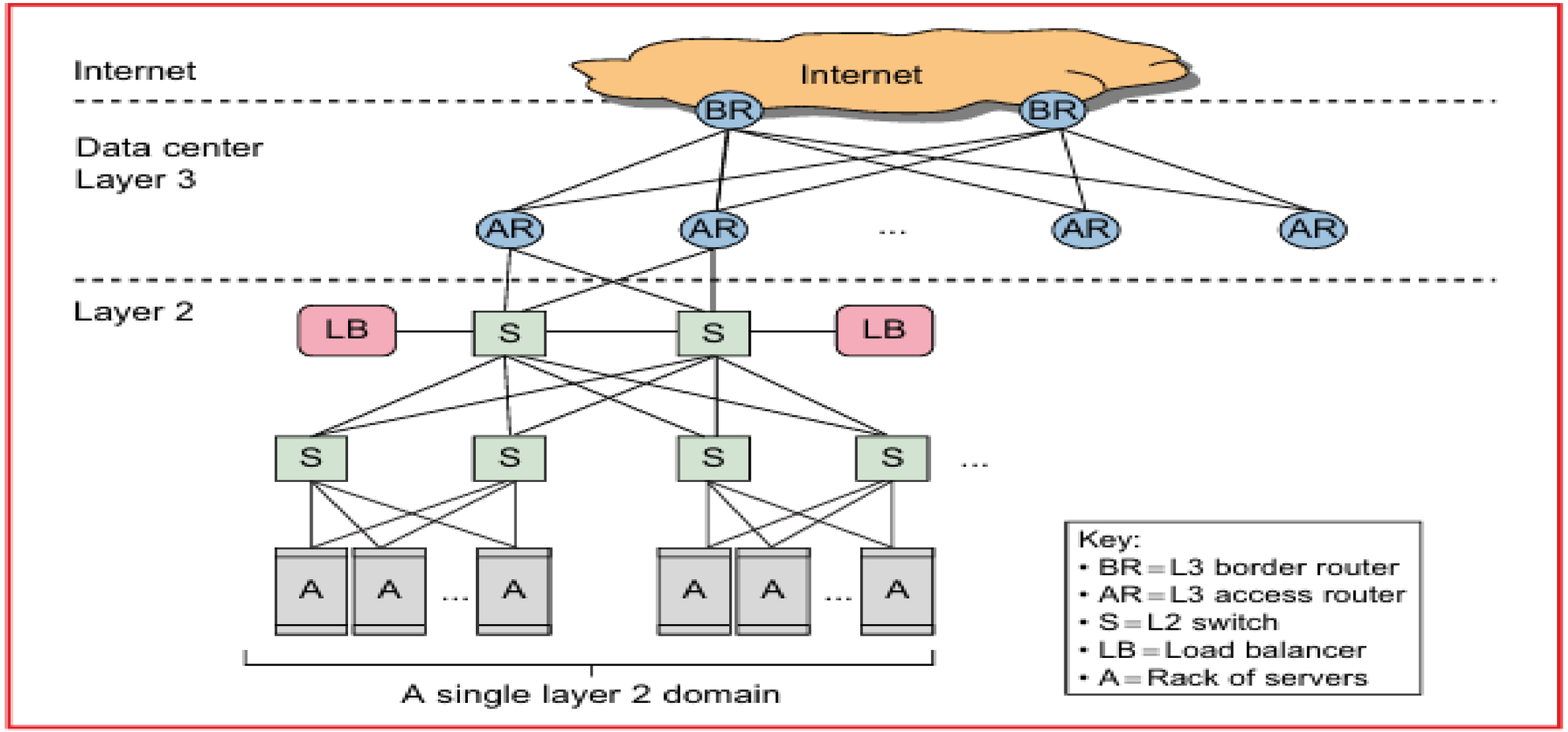
Public, Private, and Hybrid Clouds

Both public clouds and private clouds are developed in the Internet. As many clouds are generated by commercial providers or by enterprises in a distributed manner, they will be interconnected over the Internet to achieve scalable and efficient computing services. Commercial cloud providers such as Amazon, Google, and Microsoft created their platforms to be distributed geographically.



Data-Center Networking Structure

The server racks are at the bottom Layer 2, and they are connected through fast switches (S) as the hardware core. The data center is connected to the Internet at Layer 3 with many access routers (ARs) and border routers (BRs).



Cloud Ecosystem and Enabling Technologies

- **This computing model follows a pay as-you-go model.**
- **The cost is significantly reduced, because we simply rent computer resources without buying the computer in advance.**
- **All hardware and software resources are leased from the cloud provider without capital investment on the part of the users.**
- **Only the execution phase costs some money. The experts at IBM have estimated that an 80 percent to 95 percent saving results from cloud computing, compared with the conventional computing paradigm.**

Classical Computing	**Cloud Computing**
(Repeat the following cycle every 18 months)	(Pay as you go per each service provided)
Buy and own	**Subscribe**
Hardware, system software, applications to meet peak needs	- - - -
Install, configure, test, verify, evaluate, manage	**Use** (Save about 80-95% of the total cost)
- - - -	- - - -
Use	(Finally)
- - - -	**$ - Pay for what you use**
Pay $$$$$ (High cost)	based on the QoS

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.

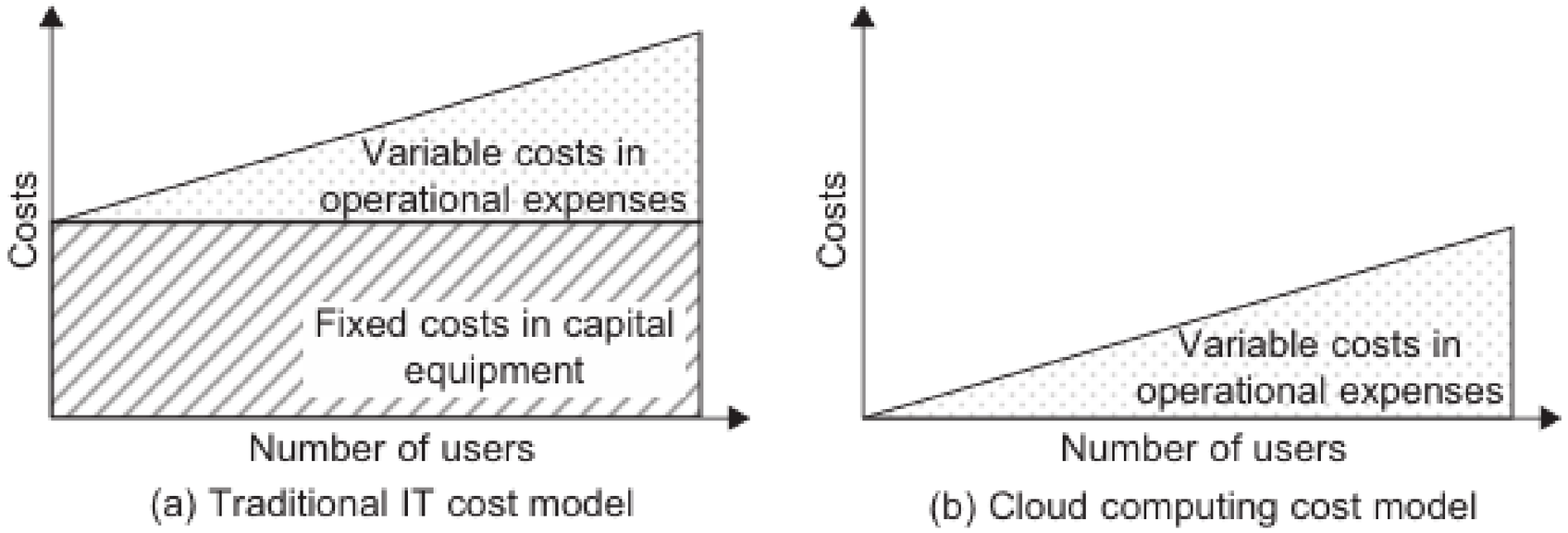
- Data privacy protection Can you trust data centers to handle your private data and records? This concern must be addressed to make clouds successful as trusted services.

- 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

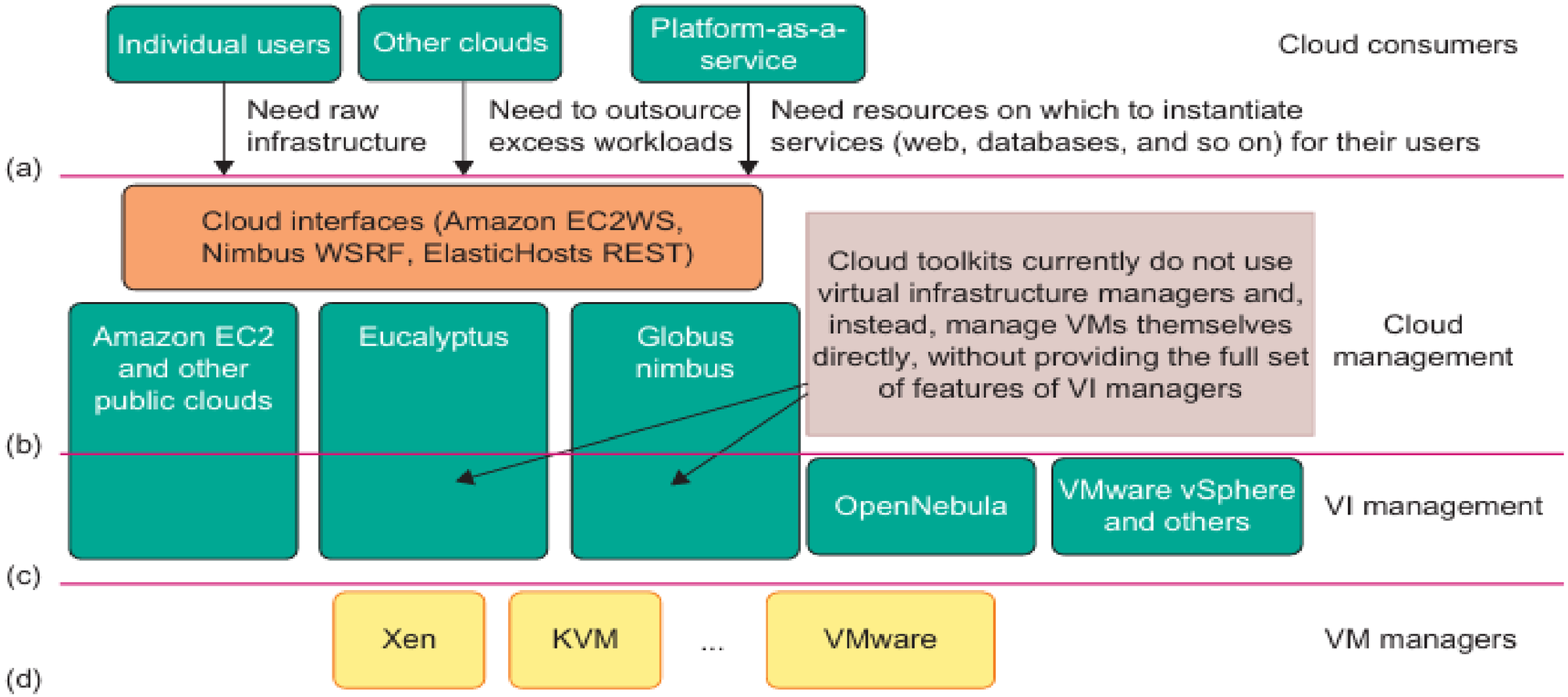
Cost Model

In traditional IT systems, the primary expense lies in the fixed capital investment required for hardware and infrastructure. While this fixed cost can be marginally reduced as the number of users grows, the operational costs—such as maintenance, energy, and administration—tend to increase significantly with user scale. As a result, the overall cost can escalate rapidly when accommodating a large user base. In contrast, cloud computing follows a pay-per-use model where computing tasks are outsourced to external data centers. This model eliminates the need for upfront capital investment in hardware, allowing users to incur only variable costs based on their actual usage



Cloud Ecosystems

Cloud ecosystem for building private clouds: (a) Consumers demand a flexible platform; (b) Cloud manager provides virtualized resources over an IaaS platform; (c) VI manager allocates VMs; (d) VM managers handle VMs installed on servers markets.



CLOUD ECOSYSTEM (CONTINUED)

four levels of ecosystem development in a private cloud.

1. At the user end, consumers demand a flexible platform.

2. At the cloud management level, the cloud manager provides virtualized resources over an IaaS platform.

3. At the virtual infrastructure (VI) management level, the manager allocates VMs over multiple server clusters.

1. Finally, at the VM management level, the VM managers handle VMs installed on individual host machines.

An ecosystem of cloud tools attempts to span both cloud management and VI management. Integrating these two layers is complicated by
the lack of open and standard interfaces between them.

These tools support dynamic placement and VM management on a pool of physical resources, automatic load balancing, server consolidation, and dynamic infrastructure resizing and partitioning. In addition to public clouds such as Amazon EC2, Eucalyptus and Globus Nimbus are open source tools for virtualization of cloud

Infrastructure-as-a-Service (IaaS)

The services provided over the cloud can be generally categorized into three different service models: namely IaaS, Platform as a Service (PaaS), and Software as a Service (SaaS). These form the three pillars on top of which cloud computing solutions are delivered to end users. All three models allow users to access services over the Internet, relying entirely on the infrastructures of cloud service providers. These models are offered based on various SLAs between providers and users.

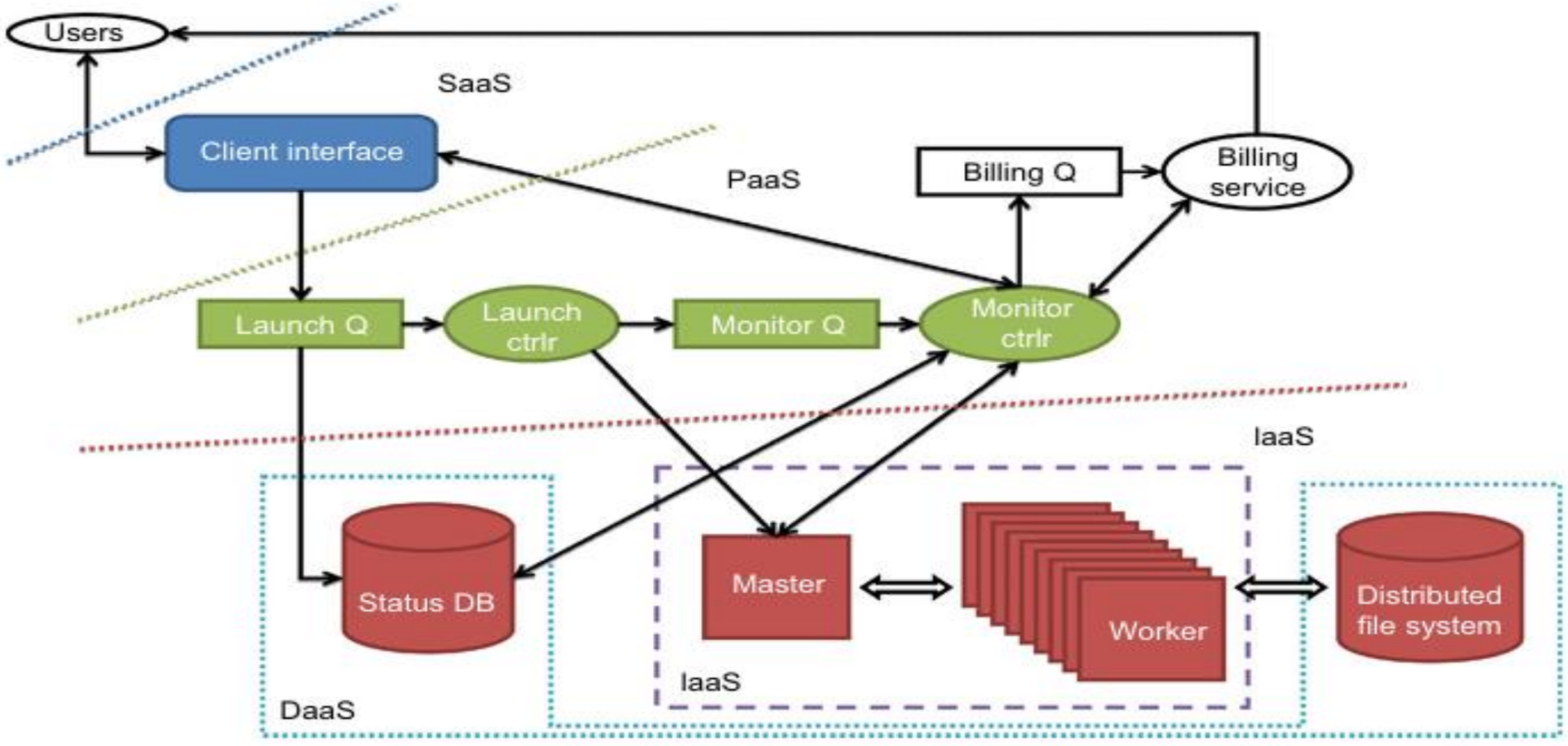
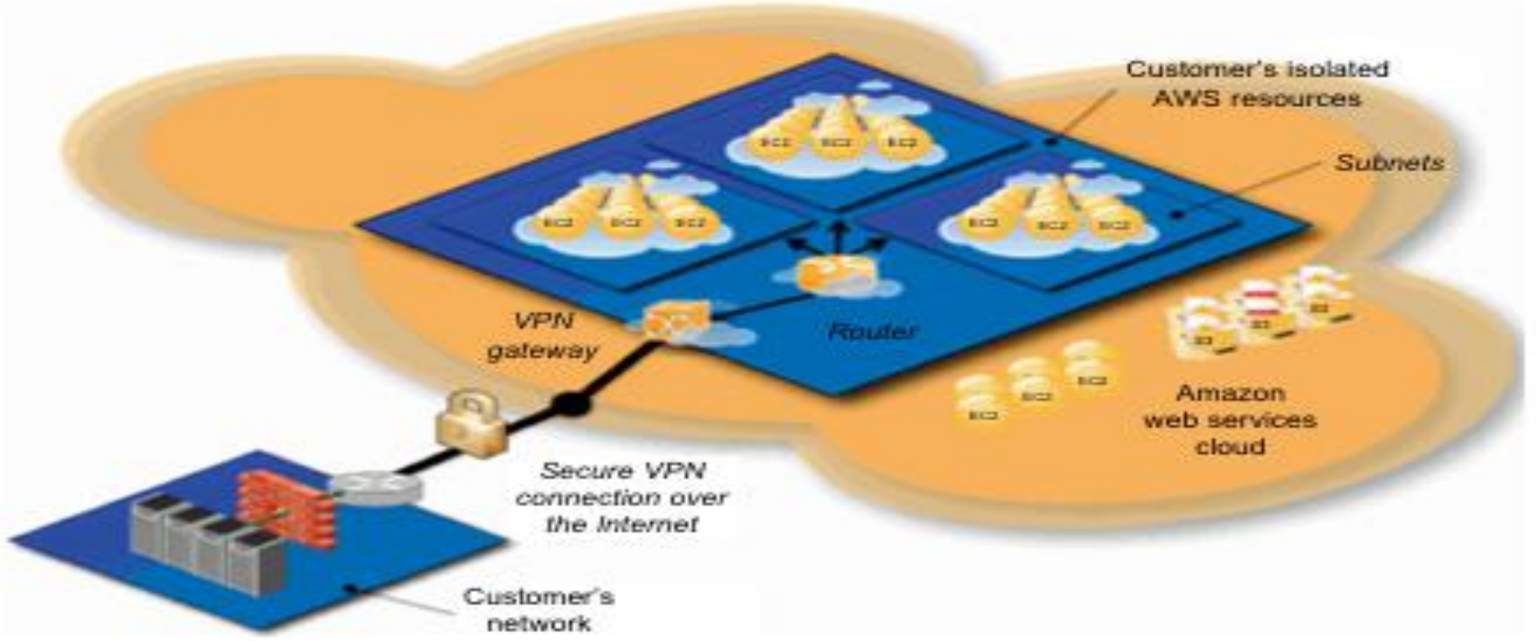


Table 4.1 Public Cloud Offerings of IaaS [10,18]

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) and Software-as-a-Service (SaaS)

To be able to develop, deploy, and manage 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. This has triggered the creation of the PaaS model to enable users to develop and deploy their user applications.

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 pro gramming languages and software tools supported by the provider (e.g., Java, Python, .NET)

Table 4.2 Five Public Cloud Offerings of PaaS [10,18]

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

Software as a Service (SaaS)

- This refers to browser-initiated application software over thousands of cloud customers.

- Services and tools offered by PaaS are utilized in construction of applications and management of their deployment on resources offered by IaaS providers. The SaaS model provides software applications as a service

- As a result, 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.

- The best examples of SaaS services include Google Gmail and docs, Microsoft SharePoint, and the 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.

Warehouse-Scale Data-Center Design

Data-Center Construction Requirements: 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.

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.

FIGURE 4.8

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

Cooling System of a Data-Center Room

The data-center room has raised floors for hiding cables, power lines, and cooling supplies.

The raised floor has a steel grid resting on stanchions about 2–4 ft above the concrete floor. The under-floor area is often used to route power cables to racks, but its primary use is to distribute cool air to the server rack. The CRAC (computer room air conditioning) unit pressurizes the raised floor plenum by blowing cold air into the plenum.

The cold air escapes from the plenum through perforated tiles that are placed in front of server racks. Racks are arranged in long aisles that alternate between cold aisles and hot aisles to avoid mixing hot and cold air. The hot air produced by the servers circulates back to the intakes of the CRAC units that cool it and then exhaust the cool air into the raised floor plenum again. Typically, the incoming coolant is at 12–14°C and the warm coolant returns to a chiller

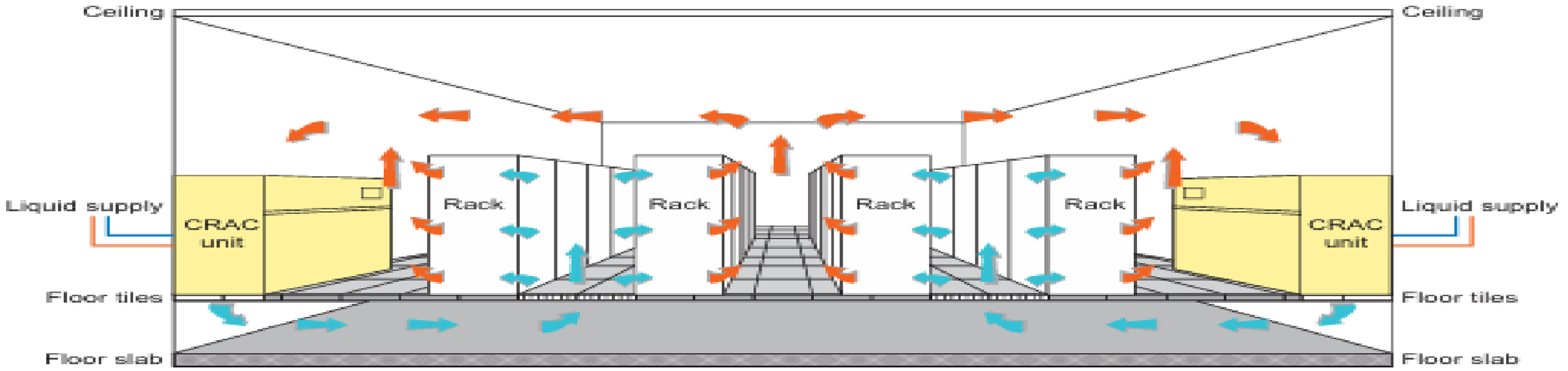


FIGURE 4.9

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

Data-Center Interconnection Networks

This network design must meet five special requirements: low latency, high bandwidth, low cost, message-passing interface (MPI) communication support, and fault tolerance

Application Traffic Support: The network topology should support all MPI communication patterns. Both point-to-point and collective MPI communications must be supported.

Network Expandability: The interconnection network should be expandable. With thousands or even hundreds of thousands of server nodes, the cluster network interconnection should be allowed to expand once more servers are added to the data center.

Fault Tolerance and Graceful Degradation: Fault tolerance of servers is achieved by replicating data and computing among redundant servers. Similar redundancy technology should apply to the network structure. Both software and hardware Network redundancy apply to cope with potential failures.

Switch-centric Data-Center Design: there are two approaches to building data-center-scale networks: One is switch Centric and the other is server-centric. In a switch-centric network, the switches are used to connect the server nodes. The server-centric design does modify the operating system running on the servers. Special drivers are designed for relaying the traffic.

Modular Data Center in Shipping Containers

GPU Programming Model

A modern data center is structured as a shipyard of server clusters housed in truck-towed containers. Figure 4.11 shows the housing of multiple sever racks in a truck-towed container in the SGI ICE Cube modular data center.

Inside the container, hundreds of blade servers are housed in racks sur rounding the container walls. An array of fans forces the heated air generated by the server racks to go through a heat exchanger, which cools the air for the next rack (detail in callout) on a continu ous loop. The SGI ICE Cube container can house 46,080 processing cores or 30 PB of storage per container.

This container-based data center was motivated by demand for lower power consumption, higher computer density, and mobility to relocate data centers to better locations with lower electricity costs, better cooling water supplies, and cheaper housing for maintenance engineers.

Container Data-Center Construction

- The data-center module is housed in a truck-towable container. The modular container design includes the network, computer, storage, and cooling gear. One needs to increase cooling efficiency by varying the water and airflow with better airflow management.

- The container must be designed to be weatherproof and easy to transport. Modular data-center construction and testing may take a few days to complete if all components are available and power and water supplies are handy.

- The container must be designed to be weatherproof and easy to transport.

- Modular data-center construction and testing may take a few days to complete if all components are available and power and water supplies are handy.

Interconnection of Modular Data Centers

The BCube is commonly used inside a server container. The containers are considered the building blocks for data centers.

The proposed network was named MDCube (for Modularized Datacenter Cube). This network connects multiple BCube containers by using high-speed switches in the Bcube.

Figure shows how a 2D MDCube is constructed from nine BCube1 containers.

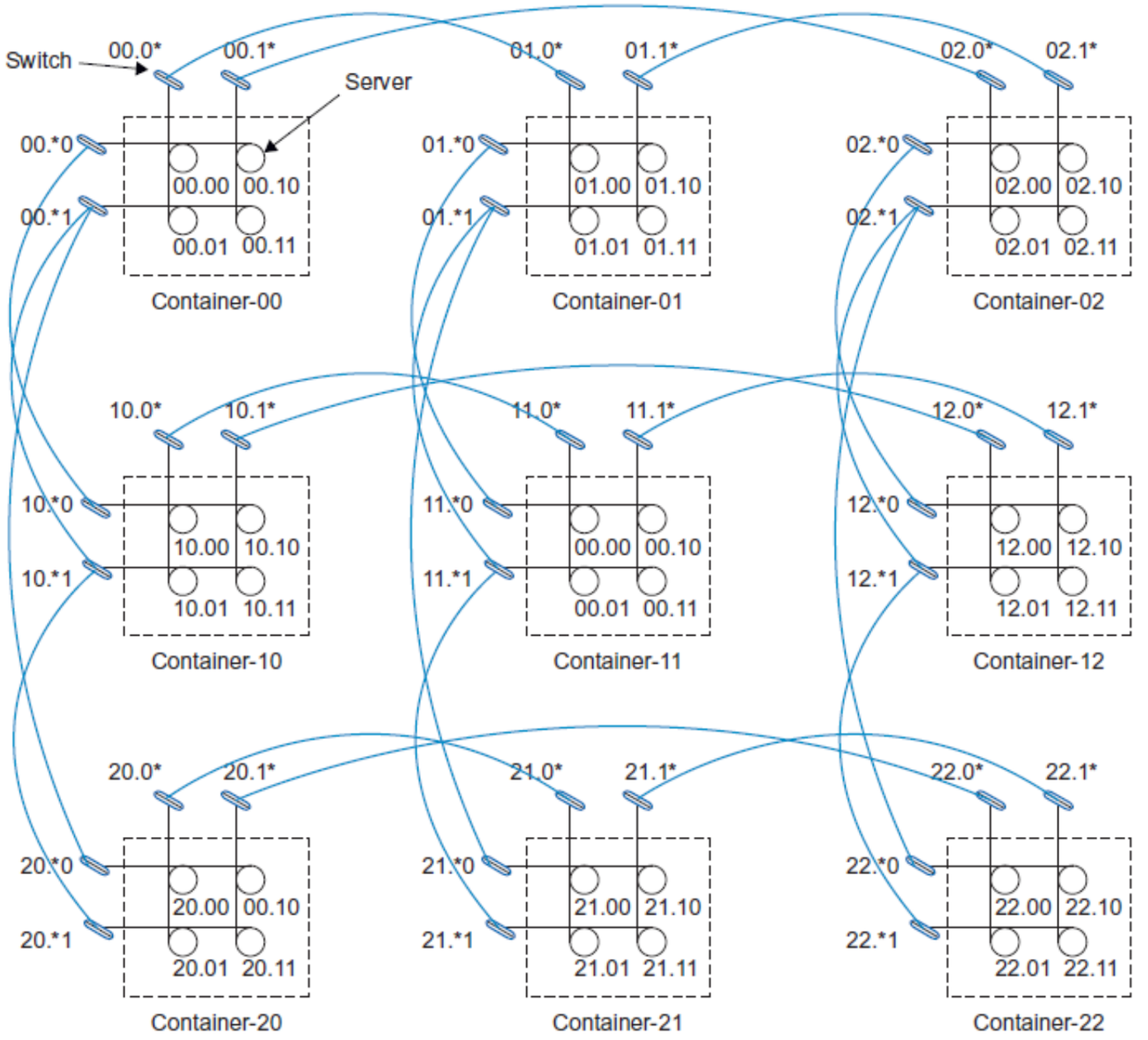


FIGURE 4.13

A 2D MDCube constructed from nine BCube containers.

(Courtesy of Wu, et al. [82])

Data-Center Management Issues

- **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.**

ARCHITECTURAL DESIGN OF COMPUTE AND STORAGE CLOUDS

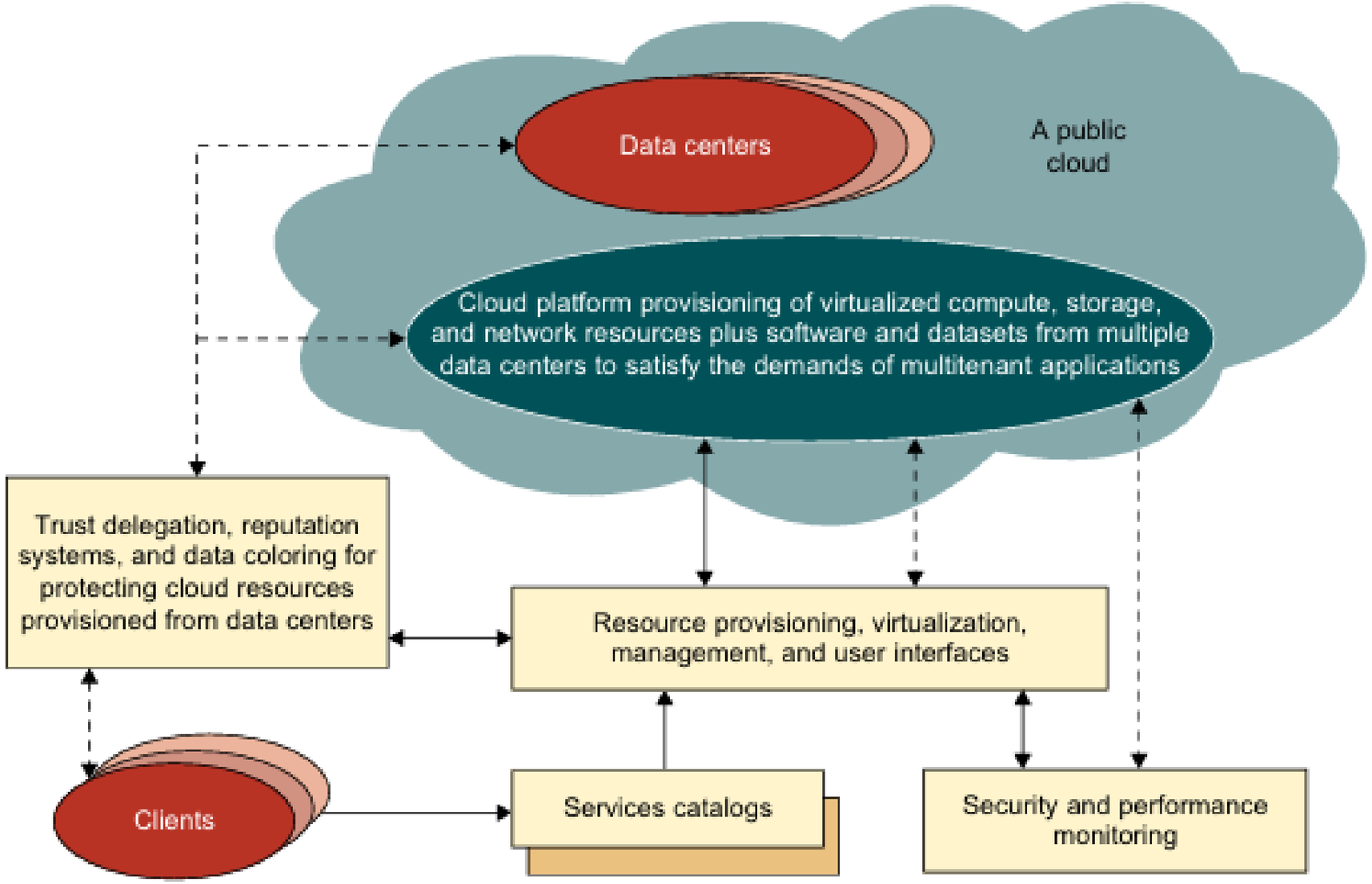
- Cloud Platform Design Goals Scalability, virtualization, efficiency, and reliability are four major design goals of a cloud computing platform. Clouds support Web 2.0 applications. Cloud management receives the user request, finds the correct resources, and then calls the provisioning services which invoke the resources in the cloud. The cloud management software needs to support both physical and virtual machines. Data can be put into multiple locations. For example, user e-mail can be put in three disks which expand to different geographically separate data centers. In such a situation, even if one of the data centers crashes, the user data is still accessible.

- Enabling Technologies for Clouds Cloud users are able to demand more capacity at peak demand, reduce costs, experiment with new services, and remove unneeded capacity, whereas service providers can increase system utilization via multiplexing, virtualization, and dynamic resource provisioning. Clouds are enabled by the progress in hardware, software, and networking technologies summarized

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 Cloud Architecture



A Generic Cloud Architecture (CONTINUED)

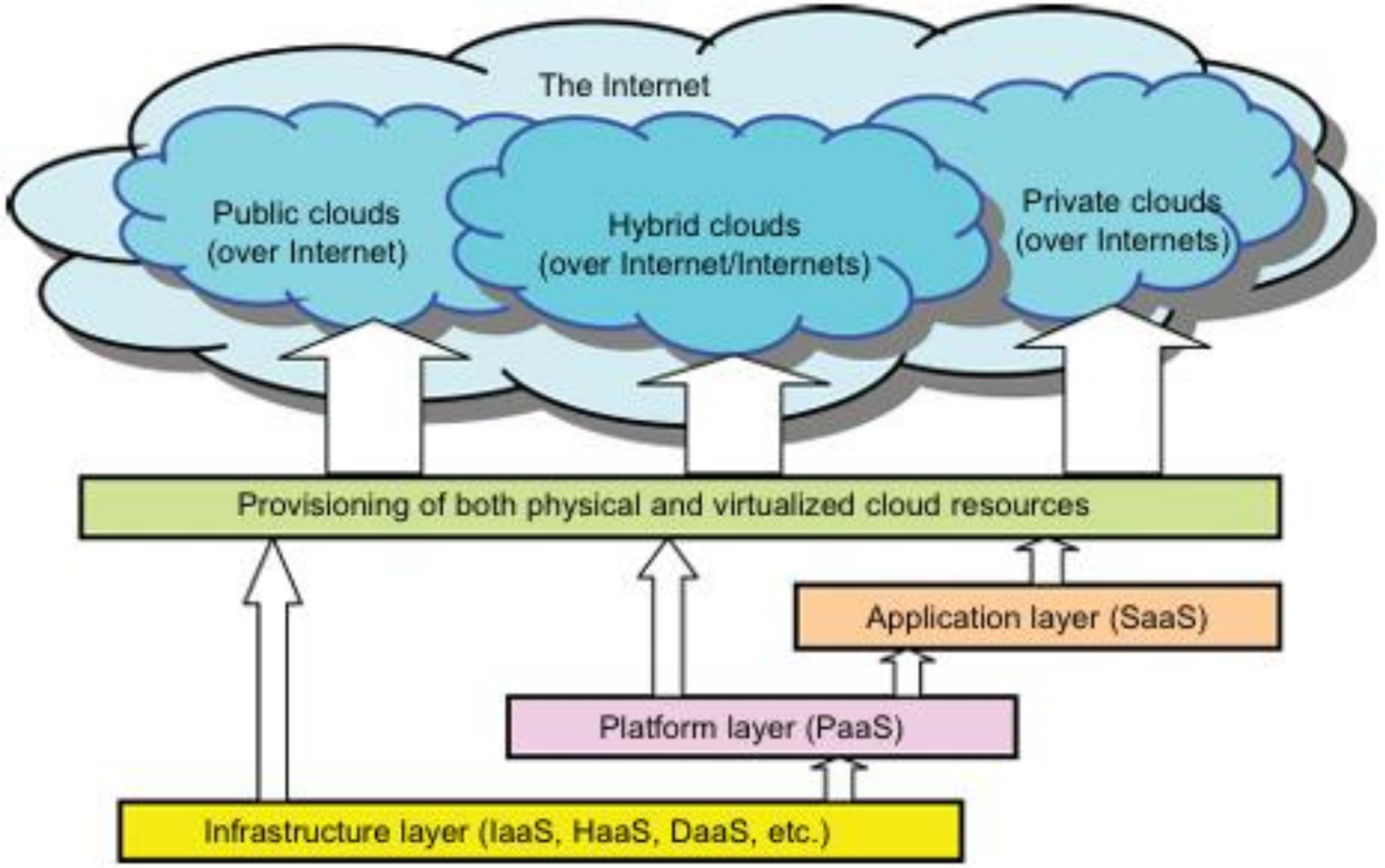
- **Massive Server Clusters: Internet cloud is made of large clusters of servers (physical or virtual).**
- **Dynamic Provisioning: Resources (servers, software, databases) are provisioned/deprovisioned on demand.**
- **User Interfaces & APIs: Users request services via UIs; APIs allow developers to use cloud capabilities.**
- **Distributed Storage: Requires distributed file systems and storage to manage large-scale data.**
- **Data Center Backbone: Cloud resources are hosted in third-party owned/operated data centers.**
- **Software as a Service (SaaS): Software is delivered as a service, abstracting underlying infrastructure.**
- **High Trust Required: Trust is essential due to vast data handled in remote data centers.**
- **Additional Resources: Includes SANs, databases, firewalls, security devices, etc.**
- **Monitoring & Metering: Tools track resource usage and performance.**
- **Automation & Resource Management: Cloud software manages nodes, detects failures, and automates maintenance.**
- **Global Data Centers: Companies like Google & Microsoft operate thousands of servers worldwide, often near hydroelectric power sources for efficiency.**
- **Performance Focus: Priority is on performance/price ratio and reliability, not just raw speed.**
- **Cloud Types:**

  Private Cloud: Easier to manage.
  Public Cloud: Easier to access.
  Hybrid Cloud: Increasingly popular due to need for both internal and external integration.

- **Security: A major concern across all types of cloud computing platforms.**

Layered Cloud Architectural Development

The architecture of a cloud is developed at three layers: infrastructure, platform, and application.These three development layers are implemented with virtualization and standardization of hardware and software resources provisioned in the cloud. The services to public, private, and hybrid clouds are conveyed to users through networking support over the Inter net and intranets involved.



(Continued)

The Infrastructure layer comprises virtualized compute, storage, and network resources, abstracting physical hardware to offer users greater flexibility. Virtualization enables automated resource provisioning and streamlines infrastructure management. Above this, the platform layer provides a reusable environment of software resources, allowing users to develop applications, test workflows, and monitor execution performance effectively.

The Platform should be able to assure users that they have scalability, dependability, and security protection. In a way, the virtualized cloud platform serves as a "system middleware" between the infrastructure and application layers of the cloud.The Internet Public clouds (over Internet) Hybrid clouds (over Internet/Internets) Private clouds (over Internets) Application layer (SaaS) Platform layer (PaaS) Infrastructure layer (IaaS, HaaS, DaaS, etc.) Provisioning of both physical and virtualized cloud resources.

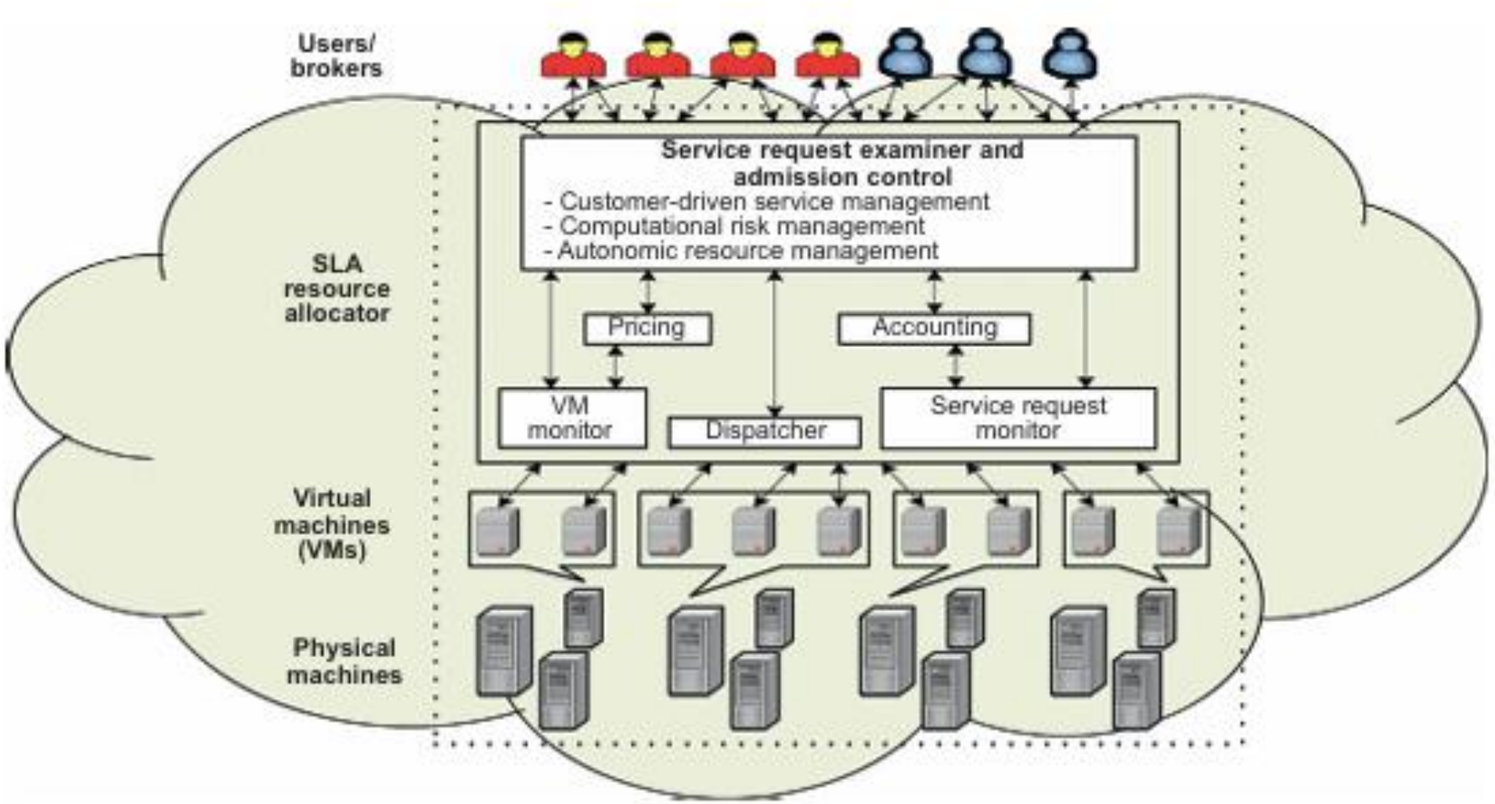
The Application layer consists of a comprehensive set of software modules designed to support Software as a Service (SaaS) applications. It encompasses services commonly used in daily office tasks, such as information retrieval, document processing, calendar management, and authentication. In addition to personal productivity, this layer is widely utilized by enterprises for business functions like marketing, sales, customer relationship management (CRM), financial transactions, and supply chain operations. Importantly, cloud services are not always confined to a single layer; many applications leverage resources across multiple layers to deliver their full functionality.

Market-Oriented Cloud Architecture

Users or brokers acting on user's behalf submit service requests from anywhere in the world to the data center and cloud to be processed.

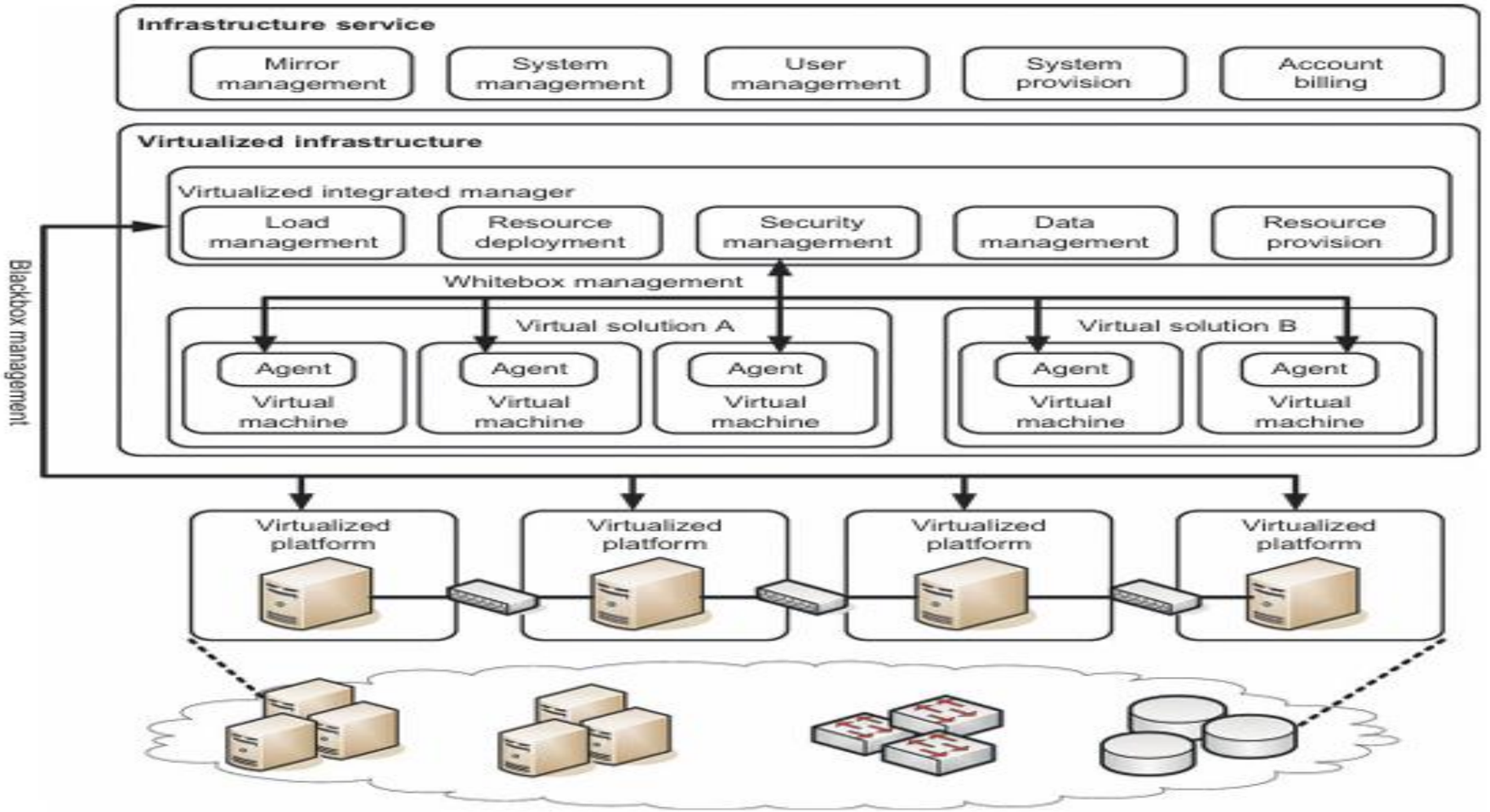
The SLA resource allocator acts as the interface between the data center/cloud service provider and external users/brokers. It requires the interaction of the following mechanisms to support SLA-oriented resource management.

When a service request is first submitted the service request examiner interprets the submitted request for QoS requirements before determining whether to accept or reject the request.



Virtualization Support and Disaster Recovery

Virtualization of servers on a shared cluster can consolidate web services. As the VMs are the containers of cloud services, the provisioning tools will first find the corresponding physical machines and deploy the VMs to those nodes before scheduling the service to run on the virtual nodes. the infrastructure needed to virtualize the servers in a data center for implementing specific cloud applications.



1.Hardware Virtualization

System virtualization software is a special kind of software which simulates the execution of hardware and runs even unmodified operating systems. Cloud computing systems use virtualization software as the running environment for legacy software such as old operating systems and unusual applications. Virtualization software is also used as the platform for developing new cloud applications that enable developers to use any operating systems and programming environments they like. The development environment and deployment environment can now be the same, which eliminates some runtime problems

Table 4.4 Virtualized Resources in Compute, Storage, and Network Clouds [4]

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 in Public Clouds - The VMware tools apply to workstations, servers, and virtual infrastructure. The Microsoft tools are used on PCs and some special servers. The Xen Enterprise tool applies only to Xen-based servers. Everyone is interested in the cloud; the entire IT industry is moving toward the vision of the cloud.

Storage Virtualization for Green Data Centers - Recent surveys from both IDC and Gartner confirm the fact that virtualization had a great impact on cost reduction from reduced power consumption in physical computing systems. This alarming situation has made the IT industry become more energy-aware. With little evolution of alternate energy resources, there is an imminent need to conserve power in all computers.. Green data centers and benefits of storage virtualization are considered to further strengthen the synergy of green computing.

Virtualization for IaaS - (1) System administrators consolidate workloads of underutilized servers in fewer servers; (2) VMs have the ability to run legacy code without interfering with other APIs; (3) VMs can be used to improve security through creation of sandboxes for running applications with questionable reliability; And (4) virtualized cloud platforms can apply performance isolation, letting providers offer some guarantees and better QoS to customer applications.

VM Cloning for Disaster Recovery -Traditional disaster recovery from one physical machine to another is rather slow, complex, and expensive. The cloning of VMs offers an effective solution. The idea is to make a clone VM on a remote server for every running VM on a local server. Among all the clone VMs, only one needs to be active. The migrated VM can run on a shared Internet connection. Only updated data and modified states are sent to the suspended VM to update its state. The Recovery Property Objective (RPO) and Recovery Time Objective (RTO) are affected by the number of snapshots taken.

Architectural Design Challenges

Challenge 1—Service Availability and Data Lock-in Problem - To achieve HA, one can consider using multiple cloud providers. Another availability obstacle is distributed denial of service (DDoS) attacks. SaaS providers the opportunity to defend against DDoS attacks by using quick scale-ups. mitigating data lock-in concerns, standardization of APIs enables a new usage model in which the same software infrastructure can be used in both public and private clouds.

Challenge 2—Data Privacy and Security Concerns-Current cloud offerings are essentially public (rather than private) networks, exposing the system to more attacks. Many obstacles can be overcome immediately with well-understood technologies such as encrypted storage, virtual LANs, and network middleboxes.

Challenge 3—Unpredictable Performance and Bottlenecks-Internet applications continue to become more data-intensive. If we assume applications to be "pulled apart" across the boundaries of clouds, this may complicate data placement and transport.Cloud users and providers have to think about the implications of placement and traffic at every level of the system, if they want to minimize costs.

Challenge 4—Distributed Storage and Widespread Software Bugs-Data centers must meet programmers' expectations in terms of scalability, data durability, and HA. Data consistence checking in SAN-connected data centers is a major challenge in cloud computing.

Challenge 5—Cloud Scalability, Interoperability, and Standardization-The pay-as-you-go model applies to storage and network bandwidth; both are counted in terms of the number of bytes used. Computation is different depending on virtualization level Open virtualization Format (OVF) describes an open, secure, portable, efficient, and extensible format for the packaging and distribution of VMs. It also defines a format for distributing software to be deployed in VMs.

Challenge 6—Software Licensing and Reputation Sharing-The primary opportunity is either for open source to remain popular or simply for commercial software companies to change their licensing structure to better fit cloud computing. One can consider using both pay-for-use and bulk-use licensing schemes to widen the business coverage.

PUBLIC CLOUD PLATFORMS: GAE, AWS, AND AZURE

1.Public Clouds and Service Offerings

Five levels of cloud players.

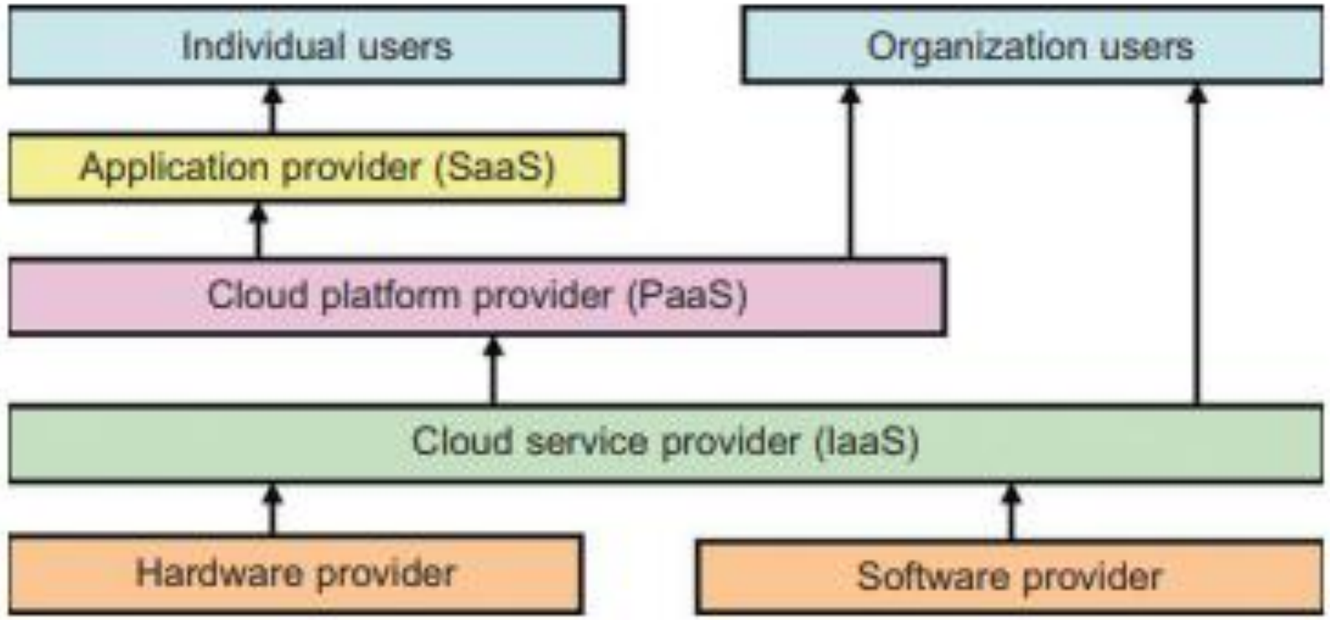


FIGURE 4.19

Roles of individual and organizational users and their interaction with cloud providers under various cloud service models.

Five Major Cloud Platforms and Their Service Offerings

Table 4.5 Five Major Cloud Platforms and Their Service Offerings [36]

Model	IBM	Amazon	Google	Microsoft	Salesforce
PaaS	BlueCloud, WCA, RC2		App Engine (GAE)	Windows Azure	Force.com
IaaS	Ensembles	AWS		Windows Azure	
SaaS	Lotus Live		Gmail, Docs	.NET service, Dynamic CRM	Online CRM, Gifttag
Virtualization		OS and Xen	Application Container	OS level/ Hypel-V	
Service Offerings	SOA, B2, TSAM, RAD, Web 2.0	EC2, S3, SQS, SimpleDB	GFS, Chubby, BigTable, MapReduce	Live, SQL Hotmail	Apex, visual force, record security
Security Features	WebSphere2 and PowerVM tuned for protection	PKI, VPN, EBS to recover from failure	Chubby locks for security enforcement	Replicated data, rule-based access control	Admin./record security, uses metadata API
User Interfaces		EC2 command-line tools	Web-based admin. console	Windows Azure portal	
Web API	Yes	Yes	Yes	Yes	Yes
Programming Support	AMI		Python	.NET Framework	

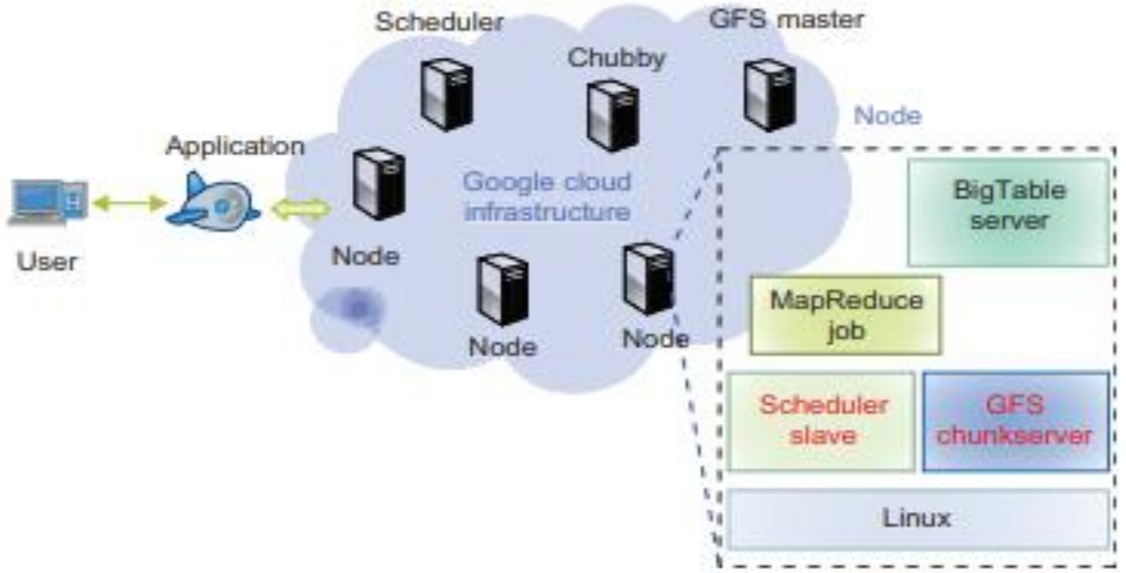
2.Google App Engine (GAE)- The Google platform is based on its search engine expertise. Google has hundreds of data centers and has installed more than 460,000 servers worldwide. For example, 200 Google data centers are used at one time for a number of cloud applications. Data items are stored in text, images, and video and are replicated to tolerate faults or failures. Here we discuss Google's App Engine (GAE) which offers a PaaS platform supporting various cloud and web applications.

Google Cloud Infrastructure : Google pioneered cloud services in Gmail, Google Docs, and Google Earth, among other applications.. GAE platform specializes in supporting scalable (elastic) web applications. GAE enables users to run their applications on a large number of data centers associated with Google's search engine operations

GAE Architecture

GFS is used for storing large amounts of data. MapReduce is for use in application program development. Chubby is used for distributed application lock services.

A typical cluster configuration can run the Google File System, MapReduce jobs, and BigTable servers for structure data.



Functional Modules of GAE

- The data store offers object-oriented, distributed, structured data storage services based on BigTable techniques. The data store secures data management operations.

- The application runtime environment offers a platform for scalable web programming and execution. It supports two development languages: Python and Java.

- The software development kit (SDK) is used for local application development. The SDK allows users to execute test runs of local applications and upload application code.

- The administration console is used for easy management of user application development cycles, instead of for physical resource management.

- The GAE web service infrastructure provides special interfaces to guarantee flexible use and management of storage and network resources by GAE.

GAE Applications : Well-known GAE applications include the Google Search Engine, Google Docs, Google Earth, and Gmail. These applications can support large numbers of users simultaneously. Users can interact with Google applications via the web interface provided by each application. Third-party application providers can use GAE to build cloud applications for providing services

Amazon Web Services (AWS)

Amazon has been a leader in providing public cloud services. Amazon applies the IaaS model in providing its services. EC2 provides the virtualized platforms to the host VMs where the cloud application can run. S3 (Simple Storage Service) provides the object-oriented storage service for users. EBS (Elastic Block Service) provides the block storage interface which can be used to support traditional applications

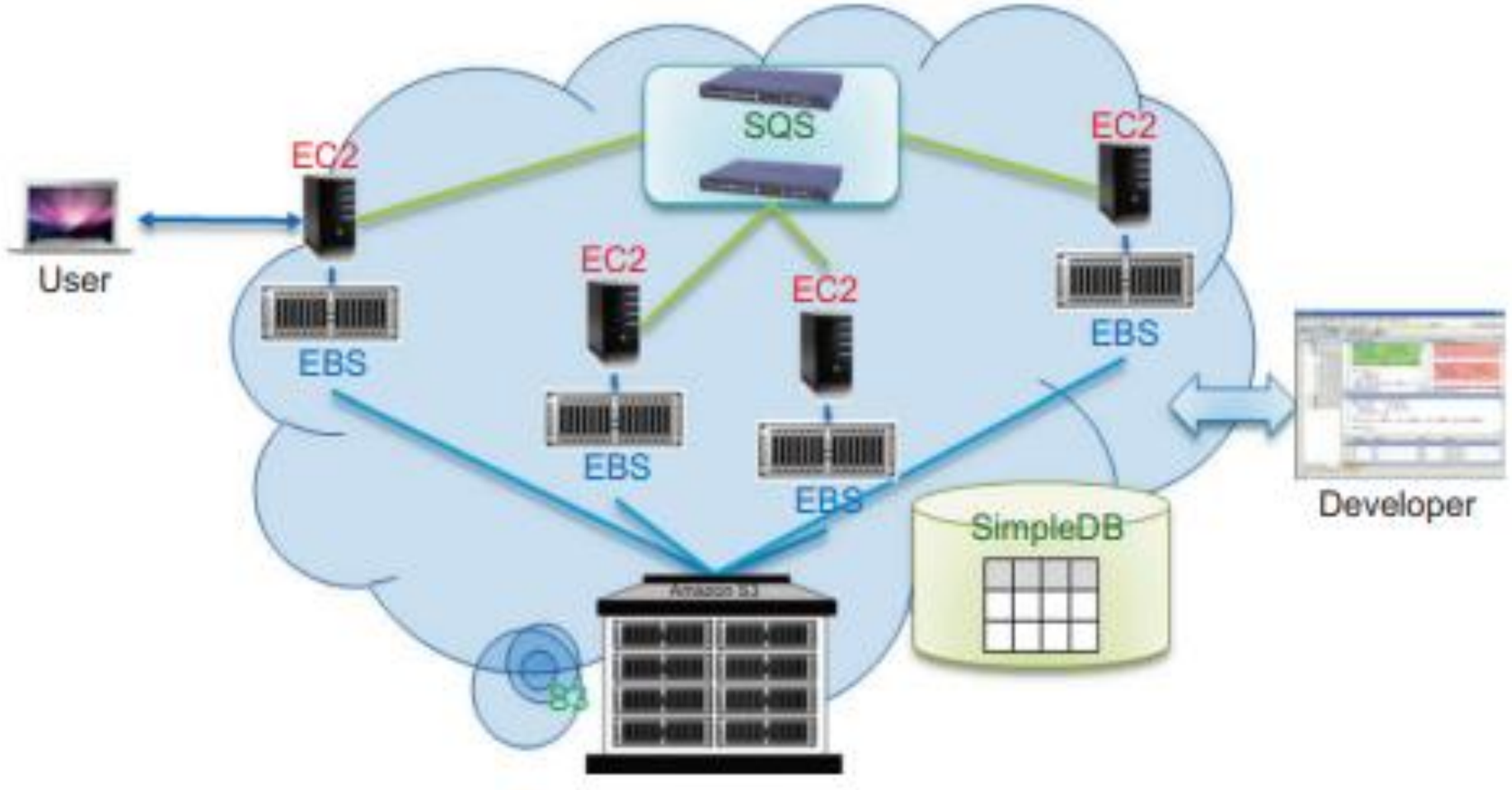


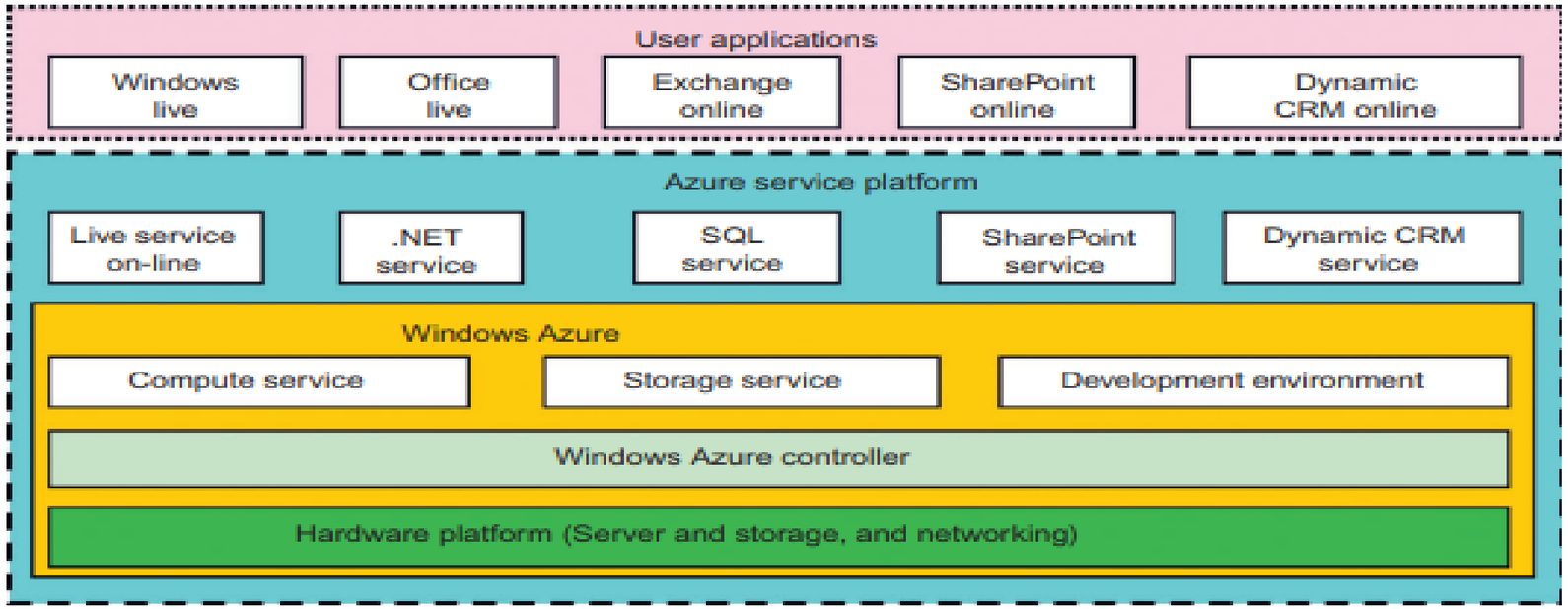
Table 4.6 AWS Offerings in 2011

Service Area	Service Modules and Abbreviated Names
Compute	Elastic Compute Cloud (EC2), Elastic MapReduce, Auto Scaling
Messaging	Simple Queue Service (SQS), Simple Notification Service (SNS)
Storage	Simple Storage Service (S3), Elastic Block Storage (EBS), AWS Import/Export
Content Delivery	Amazon CloudFront
Monitoring	Amazon CloudWatch
Support	AWS Premium Support
Database	Amazon SimpleDB, Relational Database Service (RDS)
Networking	Virtual Private Cloud (VPC) (Example 4.1, Figure 4.6), Elastic Load Balancing
Web Traffic	Alexa Web Information Service, Alexa Web Sites
E-Commerce	Fulfillment Web Service (FWS)
Payments and Billing	Flexible Payments Service (FPS), Amazon DevPay
Workforce	Amazon Mechanical Turk

(Courtesy of Amazon, http://aws.amazon.com [3])

Microsoft Windows Azure

In 2008, Microsoft launched a Windows Azure platform to meet the challenges in cloud computing. This platform is built over Microsoft data centers



- 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.

INTER-CLOUD RESOURCE MANAGEMENT

1.Extended Cloud Computing Services

Six layers of cloud services, ranging from hardware, network, and collocation to infrastructure, platform, and software applications.

Cloud players are divided into three classes: (1) cloud service providers and IT administrators, (2) software developers or vendors, and (3) end users or business users. These cloud players vary in their roles under the IaaS, PaaS, and SaaS models. From the software vendors' perspective, application performance on a given cloud platform is most important. From the providers' perspective, cloud infrastructure performance is the primarv concern

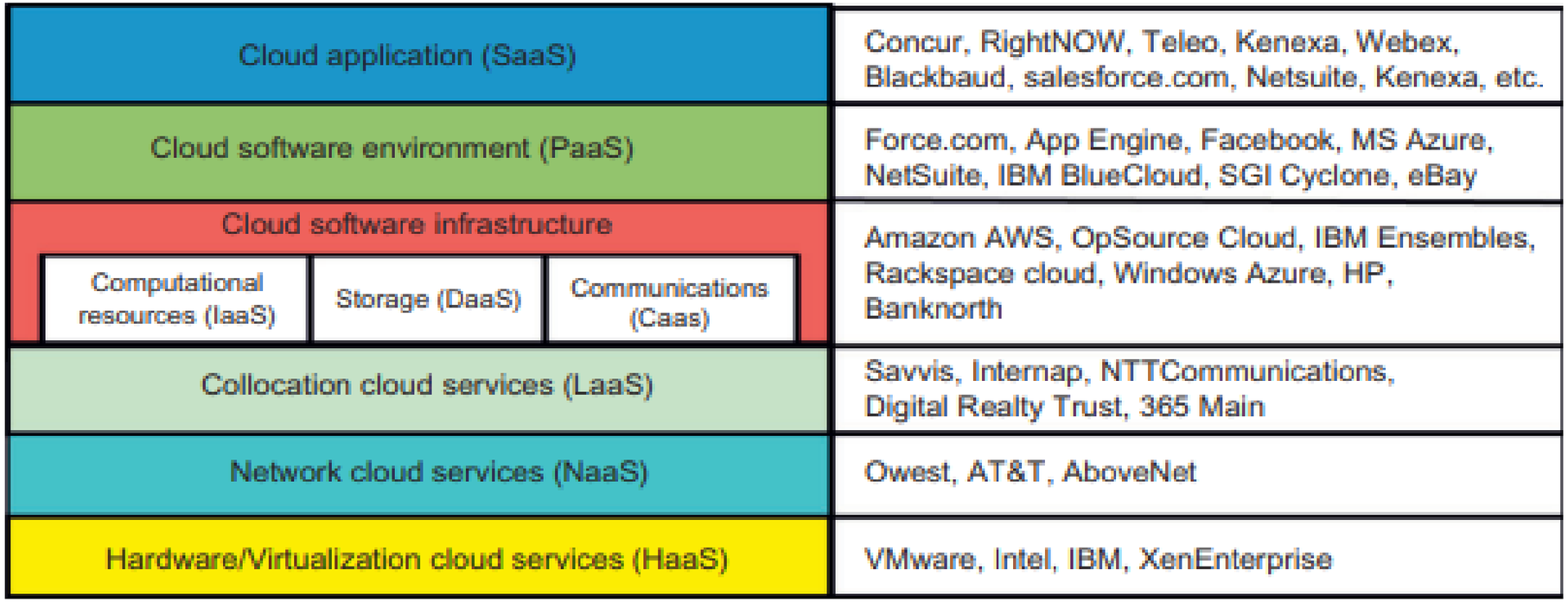


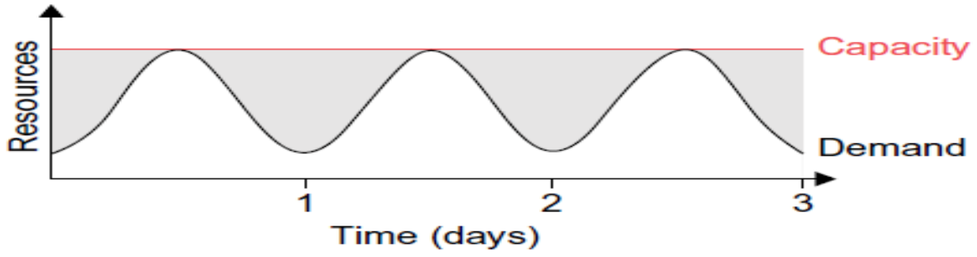
Table 4.7 Cloud Differences in Perspectives of Providers, Vendors, and Users

Cloud Players	IaaS	PaaS	SaaS
IT administrators/cloud providers	Monitor SLAs	Monitor SLAs and enable service platforms	Monitor SLAs and deploy software
Software developers (vendors)	To deploy and store data	Enabling platforms via configurators and APIs	Develop and deploy software
End users or business users	To deploy and store data	To develop and test web software	Use business software

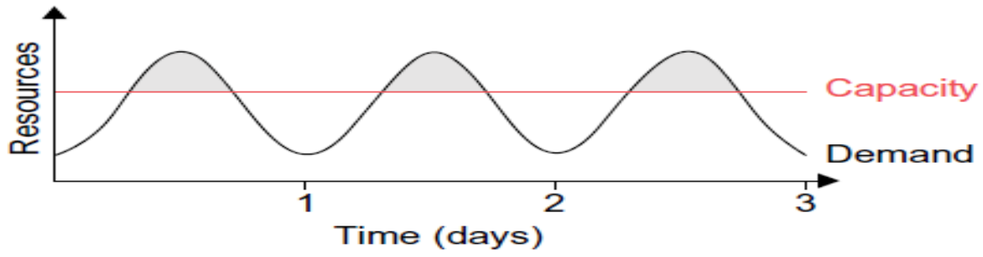
Resource Provisioning and Platform Deployment

Provisioning of Compute Resources (VMs) : 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.

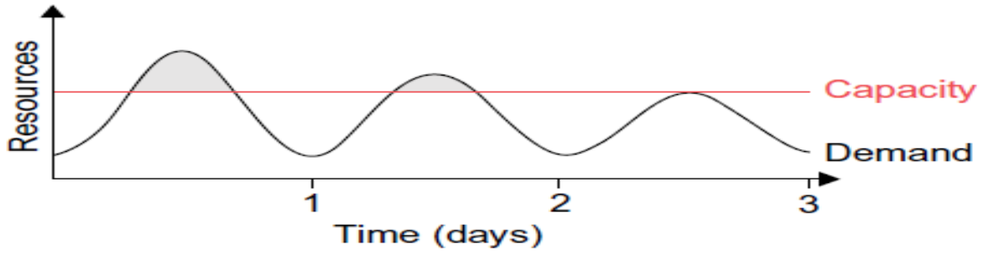
Resource Provisioning Methods: In case (a), overprovisioning with the peak load causes heavy resource waste (shaded area). In case (b), underprovisioning (along the capacity line) of resources results in losses by both user and provider in that paid demand by the users (the shaded area above the capacity) is not served and wasted resources still exist for those demanded areas below the provisioned capacity. In case (c), the constant provisioning of resources with fixed capacity to a declining user demand could result in even worse resource waste. The user may give up the service by canceling the demand, resulting in reduced revenue for the provider.



(a) Provisioning for peak load



(b) Underprovisioning 1



(c) Underprovisioning 2

- **Event-Driven Resource Provisioning**

This method adds or removes computing instances based on the current utilization level of the allocated resources. 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.

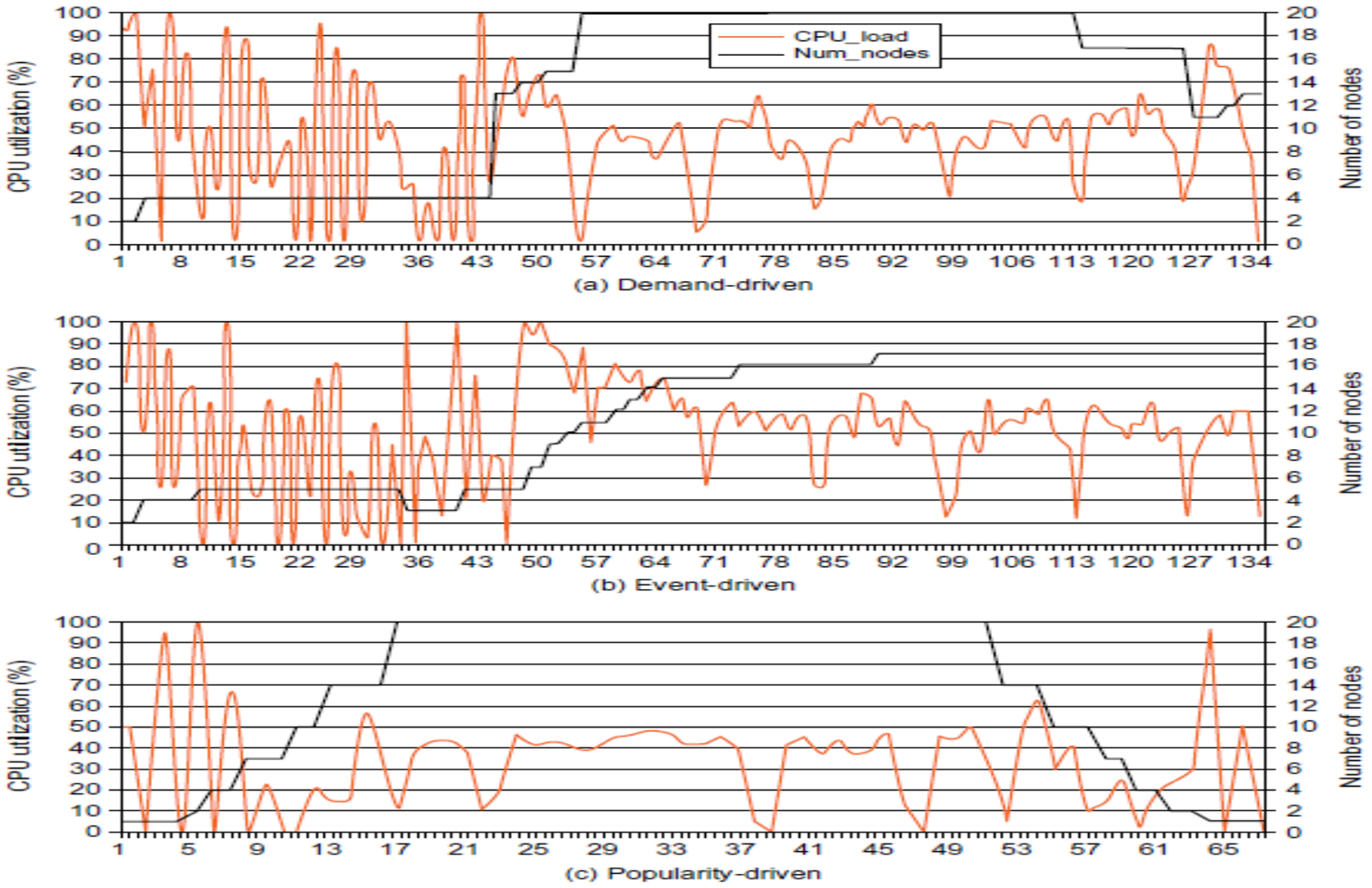


FIGURE 4.25

EC2 performance results on the AWS EC2 platform, collected from experiments at the University of Southern California using three resource provisioning methods.

(Courtesy of Ken Wu, USC)

- **Event-Driven Resource Provisioning:** This scheme adds or removes machine instances based on a specific time event. 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.

- **Popularity-Driven Resource Provisioning :** In this method, the Internet searches for popularity of certain applications and creates the instances by popularity demand. The scheme anticipates increased traffic with popularity

- **Dynamic Resource Deployment:** a scenario is illustrated by which an intergrid gateway (IGG) allocates resources from a local cluster to deploy applications in three steps: (1) requesting the VMs, (2) enacting the leases, and (3) deploying the VMs as requested. Under peak demand, this IGG interacts with another IGG that can allocate resources from a cloud computing provider.

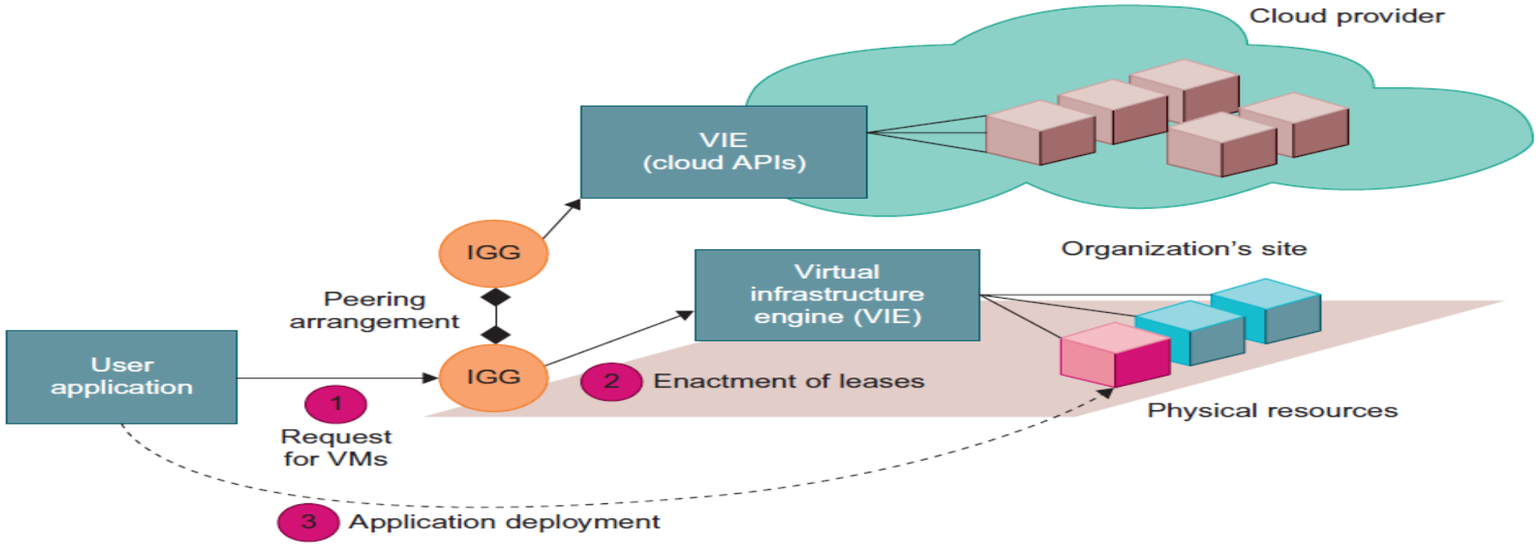


FIGURE 4.26

Cloud resource deployment using an IGG (intergrid gateway) to allocate the VMs from a Local cluster to interact with the IGG of a public cloud provider.

Provisioning of Storage Resources

Table 4.8 Storage Services in Three Cloud Computing Systems

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

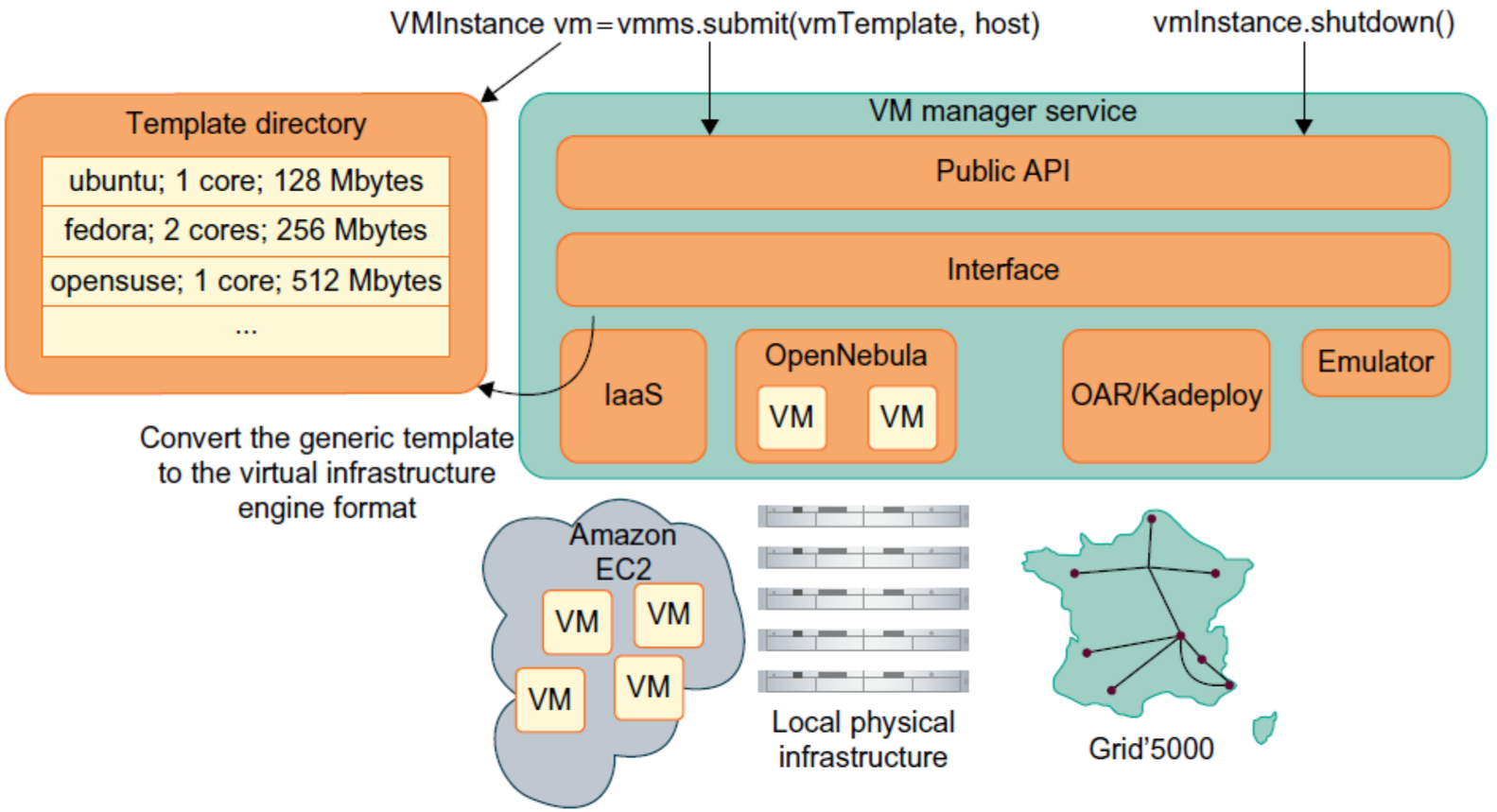


FIGURE 4.27

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

- **Independent Service Management**
- **Running Third-Party Applications**
- **Virtual Machine Manager**
- **Virtual Machine Templates**

A VM template is analogous to a computer's configuration and contains a description for a VM with the following static information:

The number of cores or processors to be assigned to the VM

The amount of memory the VM requires

The kernel used to boot the VM's operating system

The disk image containing the VM's file system

The price per hour of using a VM

The disk image that contains the VM's file system

The address of the physical machine hosting the VM

The VM's network configuration

The required information for deployment on an IaaS provider

□

Distributed VM Management

Cloud customers face challenges in determining the optimal location for hosting services, as they may be unaware of where their consumers are located. Additionally, SaaS providers may struggle to meet the QoS expectations of users from various regions.

To address this, a solution is needed for the seamless federation of data centers across multiple cloud providers, enabling dynamic scaling of applications to meet QoS requirements. Since no single cloud provider can establish data centers globally, SaaS providers must rely on multiple infrastructure providers to support their global consumer base. This need is particularly prevalent in enterprises with worldwide operations, such as Internet services and media hosting.

The Cloud bus Project at the University of Melbourne has proposed the Inter Cloud architecture, which supports the brokering and exchange of cloud resources to scale applications across different cloud environments.

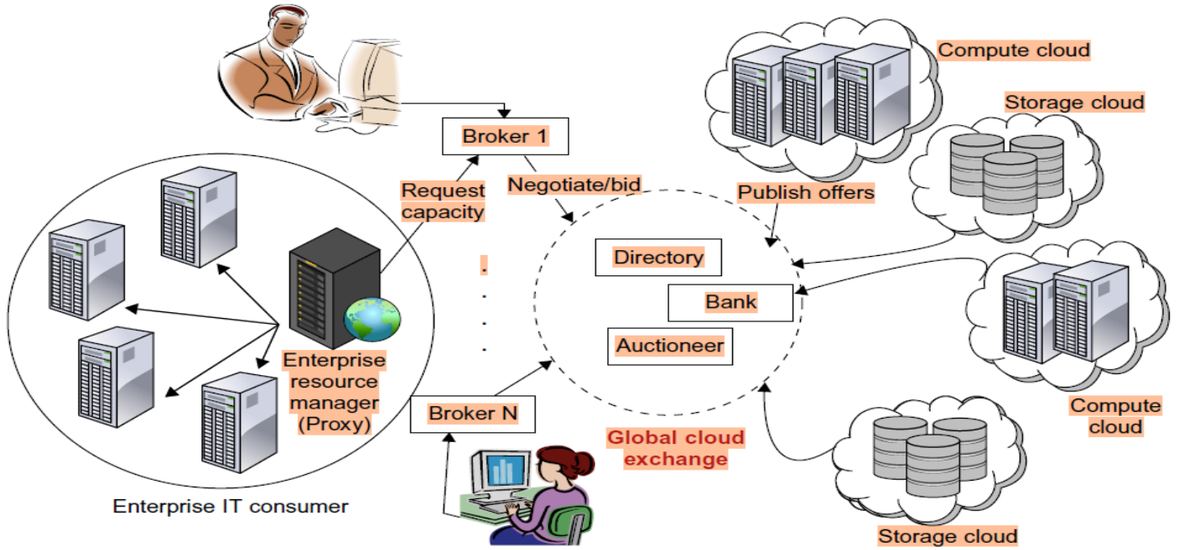


FIGURE 4.30

Inter-cloud exchange of cloud resources through brokering.