UNIT-I

SYSTEM MODELING, CLUSTERING AND VIRTUALIZATION:

1. DISTRIBUTED SYSTEM MODELS AND ENABLING TECHNOLOGIES

The Age of Internet Computing

- Billions of people use the Internet every day. As a result, supercomputer sites and large data centers must provide high-performance computing services to huge numbers of Internet users concurrently. Because of this high demand, the Linpack Benchmark for *high-performance computing (HPC)* applications is no longer optimal for measuring system performance.
- The emergence of computing clouds instead demands *high-throughput computing (HTC)* systems built with parallel and distributed computing technologies . We have to upgrade data centers using fast servers, storage systems, and high-bandwidth networks. The purpose is to advance network-based computing and web services with the emerging new technologies.

The Platform Evolution

- Computer technology has gone through five generations of development, with each generation lasting from 10 to 20 years. Successive generations are overlapped in about 10 years. For instance, from 1950 to 1970, a handful of mainframes, including the IBM 360 and CDC 6400, were built to satisfy the demands of large businesses and government organizations.

- From 1960 to 1980, lower-cost mini- computers such as the DEC PDP 11 and VAX Series became popular among small businesses and on college campuses.

- From 1970 to 1990, we saw widespread use of personal computers built with VLSI microproces- sors. From 1980 to 2000, massive numbers of portable computers and pervasive devices appeared in both wired and wireless applications. Since 1990, the use of both HPC and HTC systems hidden in.

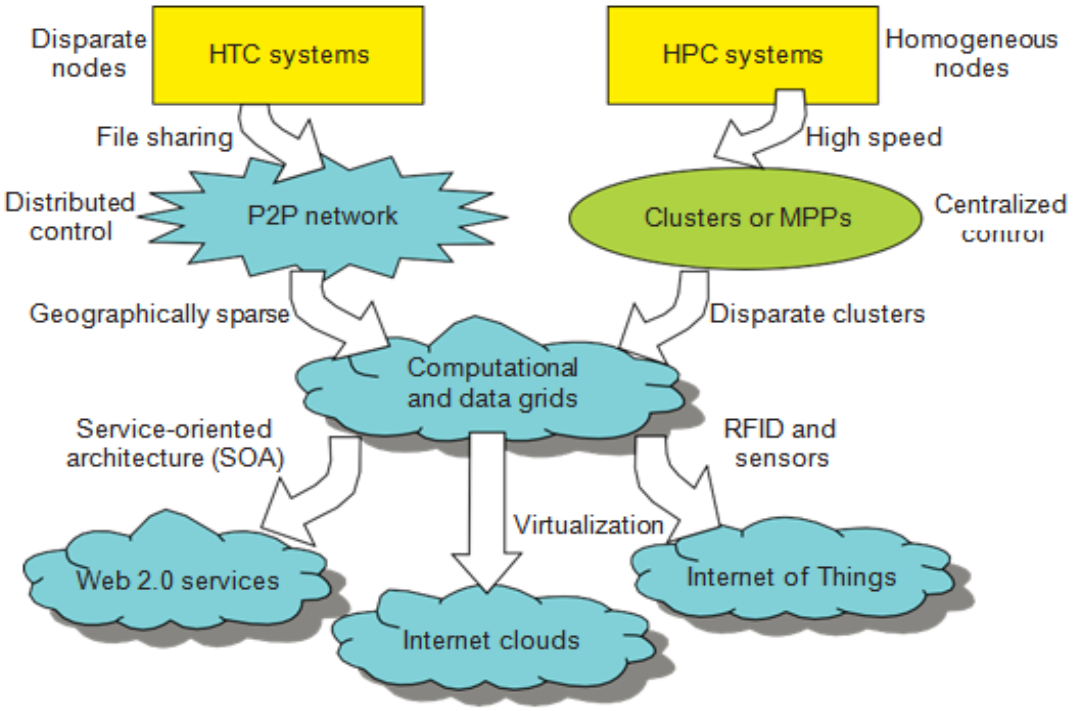


Fig 1. Evolutionary trend toward parallel, distributed, and cloud computing with clusters, MPPs, P2P networks, grids, clouds, web services, and the Internet of Things.

High-Performance Computing

- For many years, HPC systems emphasize the raw speed performance. The speed of HPC systems has increased from Gflops in the early 1990s to now Pflops in 2010. This

improvement was driven mainly by the demands from scientific, engineering, and manufacturing communities.

- For example,the Top 500 most powerful computer systems in the world are measured by floating-point speed in Linpack benchmark results. However, the number of supercomputer users is limited to less than 10% of all computer users.

- Today, the majority of computer users are using desktop computers or large servers when they conduct Internet searches and market-driven computing tasks.

High-Throughput Computing

- The development of market-oriented high-end computing systems is undergoing a strategic change from an HPC paradigm to an HTC paradigm. This HTC paradigm pays more attention to high-flux computing.

- The main application for high-flux computing is in Internet searches and web services by millions or more users simultaneously. The performance goal thus shifts to measure *high throughput* or the number of tasks completed per unit of time. HTC technology needs to not only improve in terms of batch processing speed, but also address the acute problems of cost, energy savings, security, and reliability at many data and enterprise computing centers. This book will address both HPC and HTC systems to meet the demands of all computer users.

Three New Computing Paradigms

- A Figure 1. illustrates, with the introduction of SOA, Web 2.0 services become available. Advances in virtualization make it possible to see the growth of Internet clouds as a new computing paradigm. The maturity of *radio-frequency identification (RFID)*, *Global Positioning System (GPS)*, and sensor technologies has triggered the development of the *Internet of Things (IoT)*.

Computing Paradigm Distinctions

- The high-technology community has argued for many years about the precise definitions of centralized computing, parallel computing, distributed computing, and cloud computing. In general, distributed computing is the opposite of centralized computing. The field of parallel computing overlaps with distributed computing to a great extent, and cloud computing overlaps with distributed, centralized, and parallel computing. The following list defines these terms more clearly; their architectural and operational differences are discussed further in subsequent chapters.

- Centralized computing. This is a computing paradigm by which all computer resources are centralized in one physical system. All resources (processors, memory, and storage) are fully shared and tightly coupled within one integrated OS. Many data centers and supercomputers are centralized systems, but they are used in parallel, distributed, and cloud computing applications

- Parallel computing In parallel computing, all processors are either tightly coupled with centralized shared memory or loosely coupled with distributed memory. Some authors refer to this discipline as parallel processing. Interprocessor communication is accomplished through shared memory or via message passing. A computer system capable of parallel computing is commonly known as a parallel computer . Programs running in a parallel computer are called parallel programs. The process of writing parallel programs is often referred to as parallel programming.

- Distributed computing This is a field of computer science/engineering that studies distributed systems. A distributed system consists of multiple autonomous computers, each having its own private memory, communicating through a computer network. Information exchange in a distributed

system is accomplished through message passing. A computer program that runs in a distributed system is known as a distributed program. The process of writing distributed programs is referred to as distributed programming.

- Cloud computing An Internet cloud of resources can be either a centralized or a distributed computing system. The cloud applies parallel or distributed computing, or both. Clouds can be built with physical or virtualized resources over large data centers that are centralized or distributed. Some authors consider cloud computing to be a form of utility computing or service computing

2. COMPUTER CLUSTERS FOR SCALABLE PARALLEL COMPUTING

Technologies for network-based systems

- With the concept of scalable computing under our belt, it's time to explore hardware, software, and network technologies for distributed computing system design and applications. In particular, we will focus on viable approaches to building distributed operating systems for handling massive parallelism in a distributed environment.

Cluster Development Trends Milestone Cluster Systems

- Clustering has been a hot research challenge in computer architecture. Fast communication, job scheduling, SSI, and HA are active areas in cluster research. Table 2.1 lists some milestone cluster research projects and commercial cluster products. Details of these old clusters can be found in

Project	Special Features That Support Clustering
DEC VAXcluster (1991)	A UNIX cluster of symmetric multiprocessing (SMP) servers running the VMS OS with extensions, mainly used in HA applications
U.C. Berkeley NOW Project (1995)	A serverless network of workstations featuring active messaging, cooperative filing, and GLUnix development
Rice University TreadMarks (1996)	Software-implemented distributed shared memory for use in clusters of UNIX workstations based on page migration
Sun Solaris MC Cluster (1995)	A research cluster built over Sun Solaris workstations; some cluster OS functions were developed but were never marketed sucoessfully
Tandem Himalaya Cluster (1994)	A scalable and fault-tolerant cluster for OLTP and database processing, built with nonstop operating system support
IBM SP2 Server Cluster (1996)	An AIX server cluster built with Power2 nodes and the Omega network, and supported by IBM LoadLeveler and MPI extensions
Google Search Engine Cluster (2003)	A 4,000-node server cluster built for Internet search and web service applications, supported by a distributed file system and fault tolerance
MOSIX (2010) www.mosix.org	A distributed operating system for use in Linux clusters, multiclusters, grids, and clouds; used by the research community

TABLE 2.1:MILE STONE CLUSTER RESEARCH PROJECTS

Fundamental Cluster Design Issues

- Scalable Performance: This refers to the fact that scaling of resources (cluster nodes, memory capacity, I/O bandwidth, etc.) leads to a proportional increase in performance. Of course, both scale-up and scale down capabilities are needed, depending on application demand or cost effectiveness considerations. Clustering is driven by scalability.

- Single-System Image (SSI): A set of workstations connected by an Ethernet network is not necessarily a cluster. A cluster is a single system. For example, suppose a workstation has a 300 Mflops/second processor, 512 MB of memory, and a 4 GB disk and can support 50 active users and 1,000 processes.

- By clustering 100 such workstations, can we get a single system that is equivalent to one huge workstation, or a mega-station, that has a 30 Gflops/second processor, 50 GB of memory, and a

400 GB disk and can support 5,000 active users and 100,000 processes? SSI techniques are aimed at achieving this goal.

- Internode Communication: Because of their higher node complexity, cluster nodes cannot be packaged as compactly as MPP nodes. The internode physical wire lengths are longer in a cluster than in an MPP. This is true even for centralized clusters.

- A long wire implies greater interconnect network latency. But more importantly, longer wires have more problems in terms of reliability, clock skew, and cross talking. These problems call for reliable and secure communication protocols, which increase overhead. Clusters often use commodity networks (e.g., Ethernet) with standard protocols such as TCP/IP.

- Fault Tolerance and Recovery: Clusters of machines can be designed to eliminate all single points of failure. Through redundancy, a cluster can tolerate faulty conditions up to a certain extent.

- Heartbeat mechanisms can be installed to monitor the running condition of all nodes. In case of a node failure, critical jobs running on the failing nodes can be saved by failing over to the surviving node machines. Rollback recovery schemes restore the computing results through periodic check pointing.

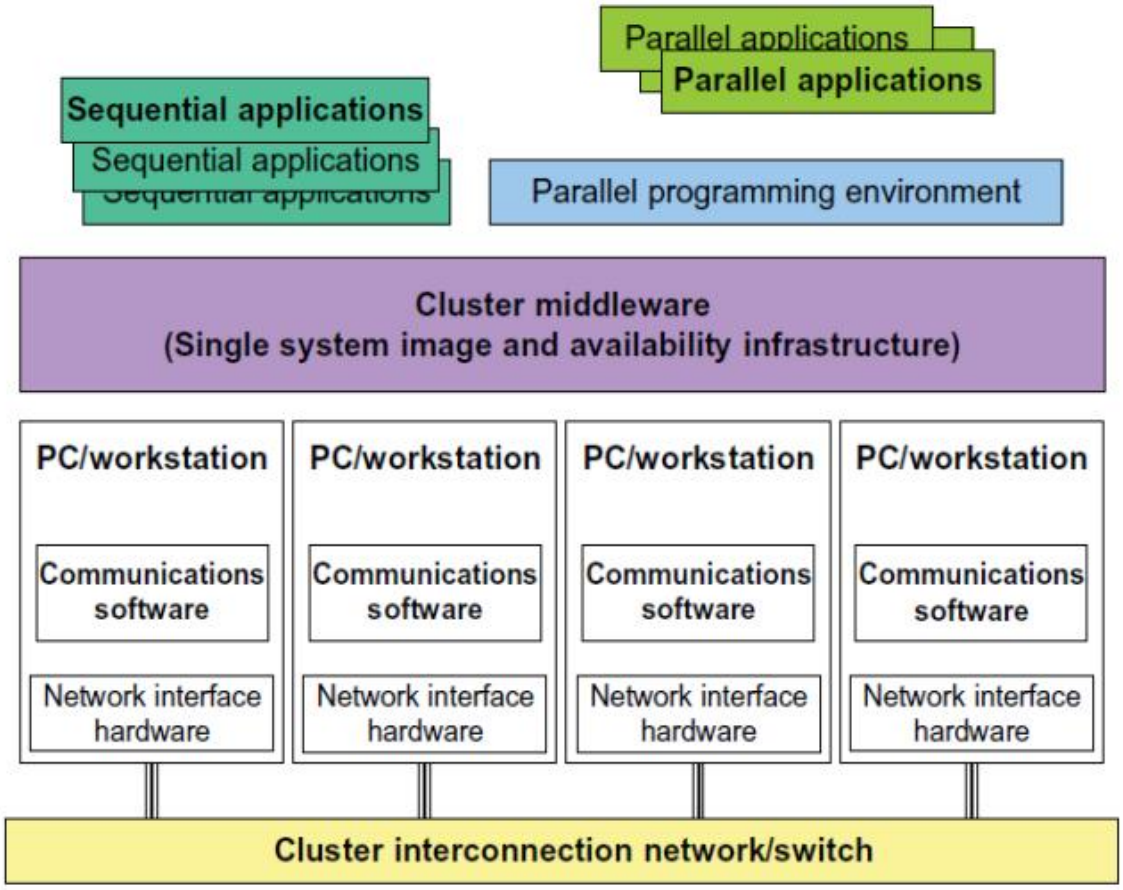


Fig2.1: Architecture of Computer Cluster

3. IMPLEMENTATION LEVELS OF VIRTUALIZATION

- Virtualization is a computer architecture technology by which multiple virtual machines (VMs) are multiplexed in the same hardware machine. The idea of VMs can be dated back to the 1960s . The purpose of a VM is to enhance resource sharing by many users and improve computer performance in terms of resource utilization and application flexibility.

- Hardware resources (CPU, memory, I/O devices, etc.) or software resources (operating system and software libraries) can be virtualized in various functional layers. This virtualization technology has been revitalized as the demand for distributed and cloud computing increased sharply in recent years .

- The idea is to separate the hardware from the software to yield better system efficiency. For example, computer users gained access to much enlarged memory space when the concept of virtual memory was introduced. Similarly, virtualization techniques can be applied to enhance the use of compute engines, networks, and storage. In this chapter we will discuss VMs and their applications for building distributed systems. According to a 2009 Gartner Report, virtualization was the top strategic technology poised to change the computer industry. With sufficient storage, any computer platform can be installed in another host computer, even if they use proc.

Levels of Virtualization Implementation

- A traditional computer runs with a host operating system specially tailored for its hardware architecture ,After virtualization, different user applications managed by their own operating systems (guest OS) can run on the same hardware, independent of the host OS.

- This is often done by adding additional software, called a virtualization layer .This virtualization layer is known as hypervisor or virtual machine monitor (VMM). The VMs are shown in the upper boxes, where applications run with their own guest OS over the virtualized CPU, memory, and I/O resources.

- The main function of the software layer for virtualization is to virtualize the physical hardware of a host machine into virtual resources to be used by the VMs, exclusively. This can be implemented at various operational levels, as we will discuss shortly.

- The virtualization software creates the abstraction of VMs by interposing a virtualization layer at various levels of a computer

system. Common virtualization layers include the instruction set architecture (ISA) level, hardware level, operating system level, library support level, and application level.

UNIT – II

FOUNDATIONS

1. INTRODUCTION TO CLOUD COMPUTING

- Cloud is a parallel and distributed computing system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements (SLA) established through negotiation between the service provider and consumers.
- Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for an optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized Service Level Agreements.

ROOTS OF CLOUD COMPUTING

- The roots of clouds computing by observing the advancement of several technologies, especially in hardware (virtualization, multi-core chips), Internet technologies (Web services, service-oriented architectures, Web 2.0), distributed computing (clusters, grids), and systems management (autonomic computing, data center automation).

From Mainframes to Clouds

- We are currently experiencing a switch in the IT world, from in-house generated computing power into utility-

supplied computing resources delivered over the Internet as Web services. This trend is similar to what occurred about a century ago when factories, which used to generate their own electric power, realized that it is was cheaper just plugging their machines into the newly formed electric power grid.

- Computing delivered as a utility can be defined as "on demand delivery of infrastructure, applications, and business processes in a security-rich, shared, scalable, and based computer environment over the Internet for a fee".

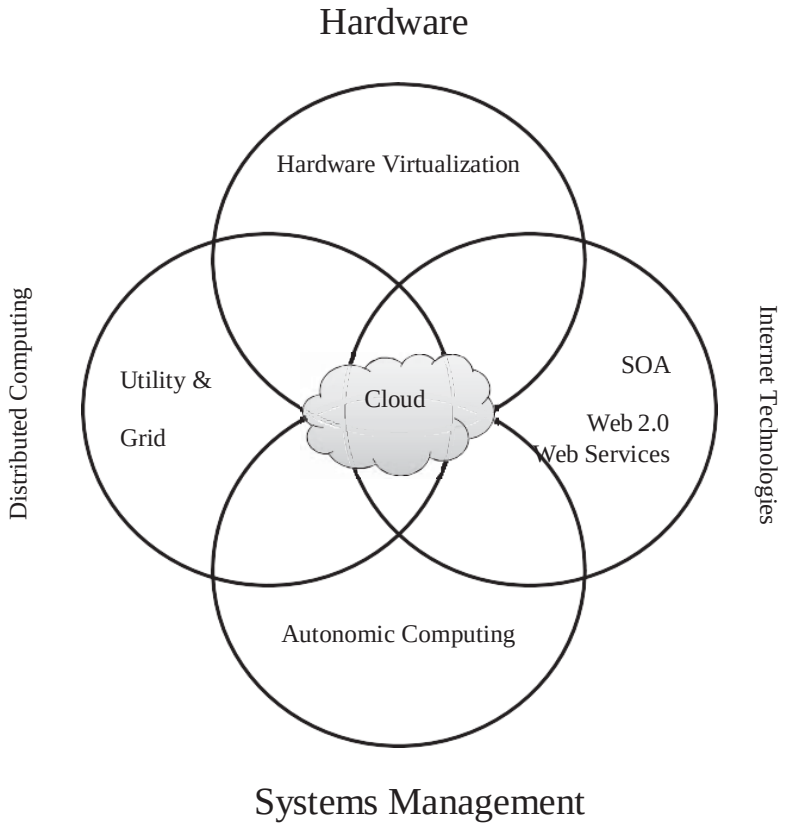


FIGURE 1.1. Convergence of various advances leading to the advent of cloud computing.

- This model brings benefits to both consumers and providers of IT services. Consumers can attain reduction on IT-related costs by choosing to obtain cheaper services from external providers as opposed to heavily investing on IT infrastructure and personnel hiring. The "on-demand" component of this model allows consumers to adapt their IT usage to rapidly increasing or unpredictable computing needs.

- Providers of IT services achieve better operational costs; hardware and software infrastructures are built to provide multiple solutions and serve many users, thus increasing efficiency and ultimately leading to faster return on investment (ROI) as well as lower total cost of ownership (TCO).

- The mainframe era collapsed with the advent of fast and inexpensive microprocessors and IT data centers moved to collections of commodity servers. Apart from its clear advantages, this new model inevitably led to isolation of workload into dedicated servers, mainly due to incompatibilities

Between software stacks and operating systems.

- These facts reveal the potential of delivering computing services with the speed and reliability that businesses enjoy with their local machines. The benefits of economies of scale and high utilization allow providers to offer computing services for a fraction of what it costs for a typical company that generates its own computing power.

SOA, WEB SERVICES, WEB 2.0, AND MASHUPS

- The emergence of Web services (WS) open standards has significantly con-tributed to advances in the domain of

software integration. Web services can glue together applications running on different messaging product platforms, enabling information from one application to be made available to others, and enabling internal applications to be made available over the Internet.

- Over the years a rich WS software stack has been specified and standardized, resulting in a multitude of technologies to describe, compose, and orchestrate services, package and transport messages between services, publish and discover services, represent quality of service (QoS) parameters, and ensure security in service access.

- WS standards have been created on top of existing ubiquitous technologies such as HTTP and XML, thus providing a common mechanism for delivering services, making them ideal for implementing a service-oriented architecture (SOA).

- The purpose of a SOA is to address requirements of loosely coupled, standards-based, and protocol-independent distributed computing. In a SOA, software resources are packaged as "services," which are well-defined, self- contained modules that provide standard business functionality and are independent of the state or context of other services. Services are described in a standard definition language and have a published interface.

- The maturity of WS has enabled the creation of powerful services that can be accessed on-demand, in a uniform way. While some WS are published with the intent of serving end-user applications, their true power resides in its interface being accessible by other services. An enterprise application that follows the SOA paradigm is a collection of services that together perform complex business logic.

- In the consumer Web, information and services may be

programmatically aggregated, acting as building blocks of complex compositions, called *service mashups*. Many service providers, such as Amazon, del.icio.us, Facebook, and Google, make their service APIs publicly accessible using standard protocols such as SOAP and REST.

- In the Software as a Service (SaaS) domain, cloud applications can be built as compositions of other services from the same or different providers. Services such user authentication, e-mail, payroll management, and calendars are examples of building blocks that can be reused and combined in a business solution in case a single, ready-made system does not provide all those features. Many building blocks and solutions are now available in public marketplaces.

- For example, Programmable Web is a public repository of service APIs and mashups currently listing thousands of APIs and mashups. Popular APIs such as Google Maps, Flickr, YouTube, Amazon eCommerce, and Twitter, when combined, produce a variety of interesting solutions, from finding video game retailers to weather maps. Similarly, Salesforce.com's offers AppExchange, which enables the sharing of solutions developed by third-party developers on top of Salesforce.com components.

GRID COMPUTING

- Grid computing enables aggregation of distributed resources and transparently access to them. Most production grids such as TeraGrid and EGEE seek to share compute and storage resources distributed across different administrative domains, with their main focus being speeding up a broad range of scientific applications, such as climate modeling, drug design, and protein analysis.

- A key aspect of the grid vision realization has been building standard Web services-based protocols that allow

distributed resources to be "discovered, accessed, allocated, monitored, accounted for, and billed for, etc., and in general managed as a single virtual system." The Open Grid Services Archi- tecture (OGSA) addresses this need for standardization by defining a set of core capabilities and behaviors that address key concerns in grid systems.

UTILITY COMPUTING

- In utility computing environments, users assign a "utility" value to their jobs, where utility is a fixed or time-varying valuation that captures various QoS constraints (deadline, importance, satisfaction). The valuation is the amount they are willing to pay a service provider to satisfy their demands. The service providers then attempt to maximize their own utility, where said utility may directly correlate with their profit. Providers can choose to prioritize

Hardware Virtualization

- Cloud computing services are usually backed by large-scale data centers composed of thousands of computers. Such data centers are built to serve many users and host many disparate applications. For this purpose, hardware virtualization can be considered as a perfect fit to overcome most operational issues of data center building and maintenance.

- The idea of virtualizing a computer system's resources, including processors, memory, and I/O devices, has been well established for decades, aiming at improving sharing and utilization of computer systems.

- Hardware virtua-lization allows running multiple operating systems and software stacks on a single physical platform. As depicted in Figure 1.2, a software

layer, the virtual machine monitor (VMM), also called a hypervisor, mediates access to the physical hardware presenting to each guest operating system a virtual machine (VM), which is a set of virtual platform interfaces .

- The advent of several innovative technologies—multi-core chips, paravir- tualization, hardware-assisted virtualization, and live migration of VMs—has contributed to an increasing adoption of virtualization on server systems. Traditionally, perceived benefits were improvements on sharing and utilization, better manageability, and higher reliability.

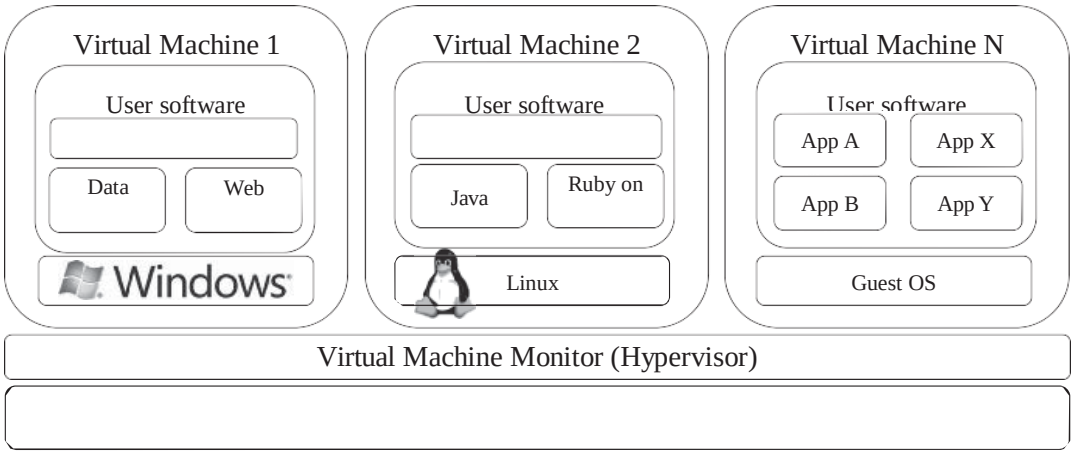


FIGURE 1.2. A hardware virtualized server hosting three virtual machines, each one running distinct operating system and user level software stack.

- Management of workload in a virtualized system, namely isolation, consolida-tion, and migration. Workload isolation is achieved since all program instructions are fully confined inside a VM, which leads to improvements in security. Better reliability is also achieved because software failures inside one VM do not affect others.

- Workload migration, also referred to as application

mobility, targets at facilitating hardware maintenance, load balancing, and disaster recovery. It is done by encapsulating a guest OS state within a VM and allowing it to be suspended, fully serialized, migrated to a different platform, and resumed immediately or preserved to be restored at a later date. A VM's state includes a full disk or partition image, configuration files, and an image of its RAM.

- A number of VMM platforms exist that are the basis of many utility or cloud computing environments. The most notable ones, VMWare, Xen, and KVM.

Virtual Appliances and the Open Virtualization Format

- An application combined with the environment needed to run it (operating system, libraries, compilers, databases, application containers, and so forth) is referred to as a "virtual appliance." Packaging application environments in the shape of virtual appliances eases software customization, configuration, and patching and improves portability. Most commonly, an appliance is shaped as a VM disk image associated with hardware requirements, and it can be readily deployed in a hypervisor.

- On-line marketplaces have been set up to allow the exchange of ready-made appliances containing popular operating systems and useful software combina- tions, both commercial and open-source.

- Most notably, the VMWare virtual appliance marketplace allows users to deploy appliances on VMWare hypervi- sors or on partners public clouds, and Amazon allows developers to share specialized Amazon Machine Images (AMI) and monetize their usage on Amazon EC2.

- In a multitude of hypervisors, where each one supports a different VM image format and the formats are incompatible with one another, a great deal of interoperability issues arises. For instance, Amazon has its Amazon machine image (AMI) format, made popular on the Amazon EC2 public cloud.

- Other formats are used by Citrix XenServer, several Linux distributions that ship with KVM, Microsoft Hyper-V, and VMware ESX.

AUTONOMIC COMPUTING

- The increasing complexity of computing systems has motivated research on autonomic computing, which seeks to improve systems by decreasing human involvement in their operation. In other words, systems should manage

themselves, with high-level guidance from humans.

- Autonomic, or self-managing, systems rely on monitoring probes and gauges (sensors), on an adaptation engine (autonomic manager) for computing optimizations based on monitoring data, and on effectors to carry out changes on the system. IBM's Autonomic Computing Initiative has contributed to define the four properties of autonomic systems: self-configuration, self- optimization, self-healing, and self-protection.

LAYERS AND TYPES OF CLOUDS

Cloud computing services are divided into three classes, according to the abstraction level of the capability provided and the service model of providers, namely:

1. Infrastructure as a Service
2. Platform as a Service and
3. Software as a Service.

Figure 1.3 depicts the layered organization of the cloud stack from physical infrastructure to applications.

- These abstraction levels can also be viewed as a layered architecture where services of a higher layer can be composed from services of the underlying layer. The reference model explains the role of each layer in an integrated architecture. A core middleware manages physical resources and the VMs deployed on top of them; in addition, it provides the required features (e.g., accounting and billing) to offer multi-tenant pay-as-you-go services.

- Cloud development environments are built on top of infrastructure services to offer application development and deployment capabilities; in this level, various programming models, libraries, APIs, and mashup editors enable the creation of a range of business, Web, and scientific applications. Once deployed in the cloud, these applications

can be consumed by end users.

Infrastructure as a Service

- Offering virtualized resources (computation, storage, and communication) on demand is known as Infrastructure as a Service (IaaS).

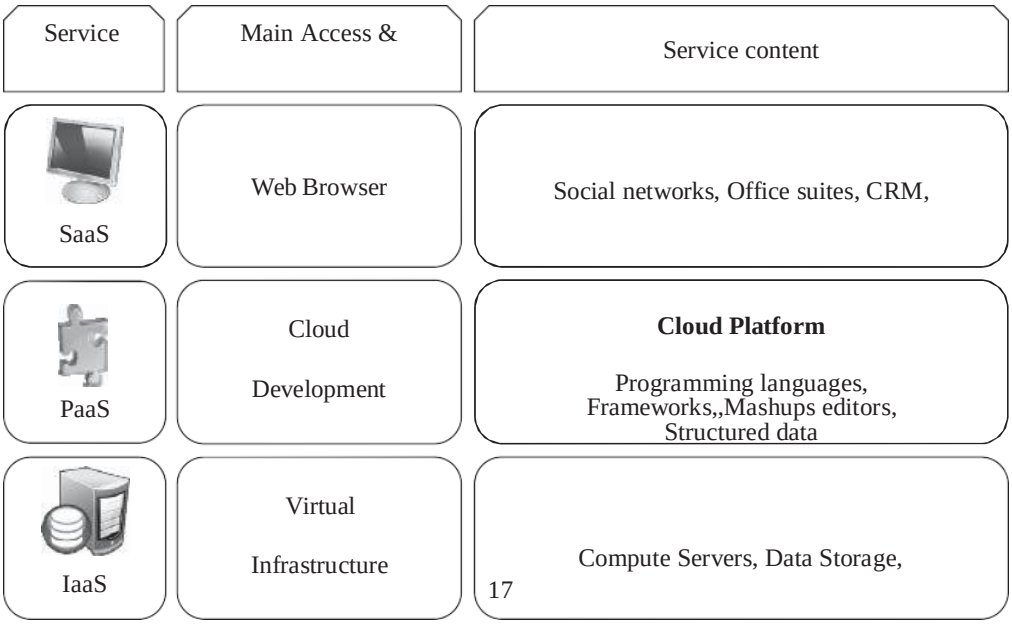


FIGURE 1.3. The cloud computing stack.

- A *cloud infrastructure* enables on-demand provisioning of servers running several choices of operating systems and a customized software stack. Infrastructure services are considered to be the bottom layer of cloud computing systems.

- Amazon Web Services mainly offers IaaS, which in the case of its EC2 service means offering VMs with a software stack that can be customized similar to how an ordinary physical server would be customized.

- Users are given privileges to perform numerous activities to the server, such as: starting and stopping it, customizing it

by installing software packages, attaching virtual disks to it, and configuring access permissions and firewalls rules.

Platform as a Service

- In addition to infrastructure-oriented clouds that provide raw computing and storage services, another approach is to offer a higher level of abstraction to make a cloud easily programmable, known as Platform as a Service (PaaS).

- A *cloud platform* offers an environment on which developers create and deploy applications and do not necessarily need to know how many processors or how much memory that applications will be using. In addition, multiple programming models and specialized services (e.g., data access, authentication, and payments) are offered as building blocks to new applications.

- Google App Engine, an example of Platform as a Service, offers a scalable environment for developing and hosting Web applications, which should be written in specific programming languages such as Python or Java, and use the services' own proprietary structured object data store.

- Building blocks include an in-memory object cache (memcache), mail service, instant messaging service (XMPP), an image manipulation service, and integration with Google Accounts authentication service.

Software as a Service

- Applications reside on the top of the cloud stack. Services provided by this layer can be accessed by end users through Web portals. Therefore, consumers are increasingly shifting from locally installed computer programs to on-line software services that offer the same functionally.
- Traditional desktop applica-tions such as word processing

and spreadsheet can now be accessed as a service in the Web. This model of delivering applications, known as Software as a Service (SaaS), alleviates the burden of software maintenance for customers and simplifies development and testing for providers.

- Salesforce.com, which relies on the SaaS model, offers business productivity applications (CRM) that reside completely on their servers, allowing costumers to customize and access applications on demand.

Deployment Models

Although cloud computing has emerged mainly from the appearance of public computing utilities, other deployment models, with variations in physical location and distribution, have been adopted. In this sense, regardless of its service class, a cloud can be classified as public, private, community, or hybrid based on model of deployment as shown in Figure 1.4.

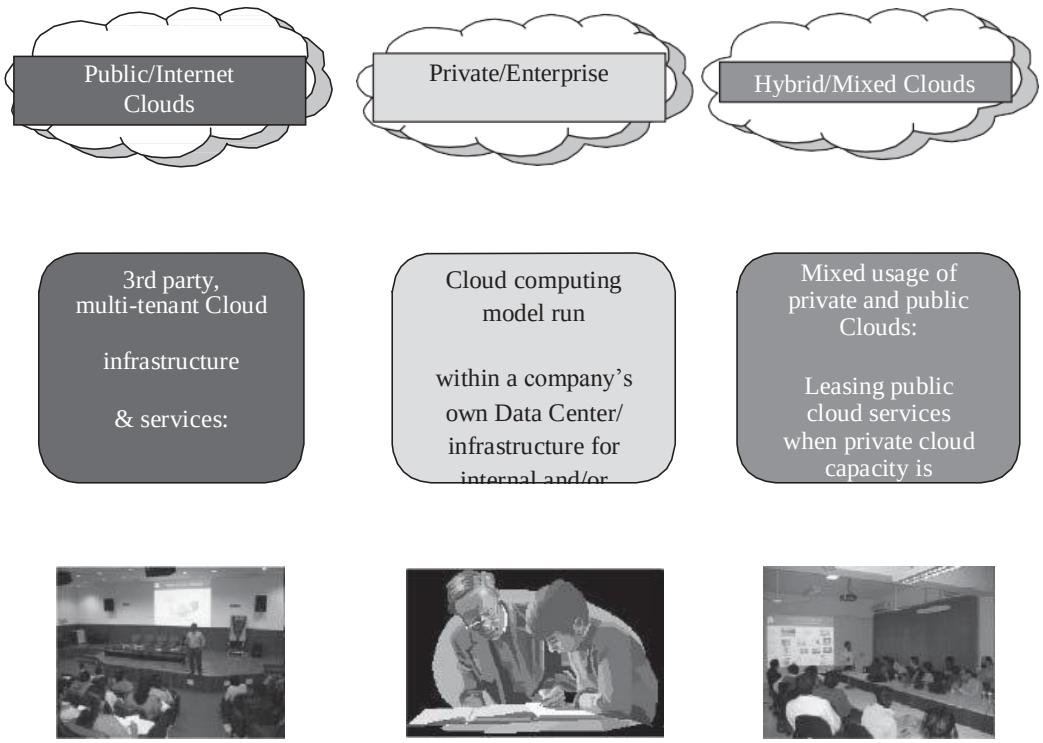


FIGURE 1.4. Types of clouds based on deployment models.

- Public cloud as a "cloud made available in a pay-as-you-go manner to the general public" and private cloud as "internal data center of a business or other organization, not made available to the general public."

- Establishing a private cloud means restructuring an existing infrastructure by adding virtualization and cloud-like interfaces. This allows users to interact with the local data center while experiencing the same advantages of public clouds, most notably self-service interface, privileged access to virtual servers, and per-usage metering and billing.

- A community cloud is "shared by several organizations and a specific community that has shared concerns (e.g., mission, security require-ments, policy, and compliance considerations) .

- A hybrid cloud takes shape when a private cloud is supplemented with computing capacity from public clouds. The approach of temporarily renting capacity to handle spikes in load is known as cloud-bursting.

FEATURES OF A CLOUD

i. Self-service
ii. Per-usage metered and billed
iii. Elastic
iv. Customizable

SELF-SERVICE

- Consumers of cloud computing services expect on-demand, nearly instant access to resources. To support this

expectation, clouds must allow self-service access so that customers can request, customize, pay, and use services without intervention of human operators.

PER-USAGE METERING AND BILLING

- Cloud computing eliminates up-front commitment by users, allowing them to request and use only the necessary amount. Services must be priced on a short- term basis (e.g., by the hour), allowing users to release (and not pay for) resources as soon as they are not needed. For these reasons, clouds must implement features to allow efficient trading of service such as pricing, accounting, and billing.
- Metering should be done accordingly for different types of service (e.g., storage, processing, and bandwidth) and usage promptly reported, thus providing greater transparency.

ELASTICITY

Cloud computing gives the illusion of infinite computing resources available on demand. Therefore users expect clouds to rapidly provide resources in any Quantity at any time.

In particular, it is expected that the additional resources can be

i. Provisioned, possibly automatically, when an application load increases and
ii. Released when load decreases (scale up and down).

CUSTOMIZATION

- In a multi-tenant cloud a great disparity between user needs is often the case. Thus, resources rented from the cloud must be highly customizable. In the case of infrastructure services, customization means allowing users to deploy specialized virtual appliances and to be given privileged (root) access to the virtual servers.

CLOUD INFRASTRUCTURE MANAGEMENT

A key challenge IaaS providers face when building a cloud infrastructure is managing physical and virtual resources, namely servers, storage, and net- works.

- The software toolkit responsible for this orchestration is called a virtual infrastructure manager (VIM). This type of software resembles a traditional operating system but instead of dealing with a single computer, it aggregates resources from multiple computers, presenting a uniform view to user and applications. The term "cloud operating system" is also used to refer to it.

- The availability of a remote cloud-like interface and the ability of managing many users and their permissions are the primary features that would distinguish cloud toolkits from VIMs.

Virtually all VIMs we investigated present a set of basic features related to managing the life cycle of VMs, including networking groups of VMs together and setting up virtual disks for VMs.

FEATURES AVAILABLE IN VIMS

VIRTUALIZATION SUPPORT:

- The multi-tenancy aspect of clouds requires multiple customers with disparate requirements to be served by a single hardware infrastructure. Virtualized resources (CPUs, memory, etc.) can be sized and resized with certain flexibility. These features make hardware virtualization, the ideal technology to create a virtual infrastructure that partitions a data center among multiple tenants.

SELF-SERVICE, ON-DEMAND RESOURCE, PROVISIONING:

- Self-service access to resources has been perceived as one the most attractive features of clouds. This feature enables users to directly obtain services from clouds, such as spawning the creation of a server and tailoring its software, configurations, and security policies, without interacting with a human system administrator.

- This cap- ability "eliminates the need for more time-consuming, labor-intensive, human- driven procurement processes familiar to many in IT". Therefore, exposing a self-service interface, through which users can easily interact with the system, is a highly desirable feature of a VI manager.

MULTIPLE BACKEND HYPERVISORS:

- Different virtualization models and tools offer different benefits, drawbacks, and limitations. Thus, some VI managers provide a uniform management layer regardless of the virtualization technology used. This characteristic is more visible in open-source VI managers, which usually provide pluggable drivers to interact with multiple hypervisors.

STORAGE VIRTUALIZATION:

- Virtualizing storage means abstracting logical storage from physical storage. By consolidating all available storage devices in a data center, it allows creating virtual disks independent from device and location. Storage devices are commonly organized in a storage area network (SAN) and attached to servers via protocols such as Fibre Channel, iSCSI, and NFS; a storage controller provides the layer of abstraction between virtual and physical storage.

- In the VI management sphere, storage virtualization support is often restricted to commercial products of

companies such as VMWare and Citrix. Other products feature ways of pooling and managing storage devices, but administrators are still aware of each individual device.

- *Interface to Public Clouds.* Researchers have perceived that extending the capacity of a local in-house computing infrastructure by borrowing resources from public clouds is advantageous. In this fashion, institutions can make good use of their available resources and, in case of spikes in demand, extra load can be offloaded to rented resources .

- A VI manager can be used in a hybrid cloud setup if it offers a driver to manage the life cycle of virtualized resources obtained from external cloud providers. To the applications, the use of leased resources must ideally be transparent.

- Virtual Networking. Virtual networks allow creating an isolated network on top of a physical infrastructure independently from physical topology and locations . A virtual LAN (VLAN) allows isolating traffic that shares a switched network, allowing VMs to be grouped into the same broadcast domain. Additionally, a VLAN can be configured to block traffic originated from VMs from other networks. Similarly, the VPN (virtual private network) concept is used to describe a secure and private overlay network on top of a public network (most commonly the public Internet).

DYNAMIC RESOURCE ALLOCATION:

- In cloud infrastructures, where applications have variable and dynamic needs, capacity management and demand predic-tion are especially complicated. This fact triggers the need for dynamic resource allocation aiming at obtaining a timely match of supply and demand.

- A number of VI managers include a dynamic resource allocation feature that continuously monitors utilization

across resource pools and reallocates available resources among VMs according to application needs.

VIRTUAL CLUSTERS:

- Several VI managers can holistically manage groups of VMs. This feature is useful for provisioning computing *virtual clusters on demand*, and interconnected VMs for multi-tier Internet applications.

RESERVATION AND NEGOTIATION MECHANISM:

- When users request computational resources to available at a specific time, requests are termed advance reservations (AR), in contrast to best-effort requests, when users request resources whenever available.

HIGH AVAILABILITY AND DATA RECOVERY:

- The high availability (HA) feature of VI managers aims at minimizing application downtime and preventing business disruption.

INFRASTRUCTURE AS A SERVICE PROVIDERS

Public Infrastructure as a Service providers commonly offer virtual servers containing one or more CPUs, running several choices of operating systems and a customized software stack.

FEATURES

The most relevant features are:

i. Geographic distribution of data centers;

ii. Variety of user interfaces and APIs to access the system;

iii. Specialized components and services that aid particular applications (e.g., load- balancers, firewalls);

iv. Choice of virtualization platform and operating systems; and

v. Different billing methods and period (e.g., prepaid vs. post-paid, hourly vs. monthly).

GEOGRAPHIC PRESENCE:

- Availability zones are "distinct locations that are engineered to be insulated from failures in other availability zones and provide inexpensive, low-latency network connectivity to other availability zones in the same region." Regions, in turn, "are geographi- cally dispersed and will be in separate geographic areas or countries."

USER INTERFACES AND ACCESS TO SERVERS:

- A public IaaS provider must provide multiple access means to its cloud, thus catering for various users and their preferences. Different types of user interfaces (UI) provide different levels of abstraction, the most common being graphical user interfaces (GUI), command-line tools (CLI), and Web service (WS) APIs.

- GUIs are preferred by end users who need to launch, customize, and monitor a few virtual servers and do not necessary need to repeat the process several times.

ADVANCE RESERVATION OF CAPACITY*:*

- Advance reservations allow users to request for an IaaS provider to reserve resources for a specific time frame in the future, thus ensuring that cloud resources will be available at that time.

- Amazon Reserved Instances is a form of advance reservation of capacity, allowing users to pay a fixed amount of money in advance to guarantee resource availability at anytime during an agreed period and then paying a discounted hourly rate when resources are in use.

AUTOMATIC SCALING AND LOAD BALANCING:

- It allow users to set conditions for when they want their applications to scale up and down, based on application-specific metrics such as transactions per second, number of simultaneous users, request latency, and so forth.

- When the number of virtual servers is increased by automatic scaling, incoming traffic must be automatically distributed among the available servers. This activity enables applications to promptly respond to traffic increase while also achieving greater fault tolerance.

SERVICE-LEVEL AGREEMENT:

- Service-level agreements (SLAs) are offered by IaaS providers to express their commitment to delivery of a certain QoS. To customers it serves as a warranty. An SLA usually include availability and performance guarantees.

HYPERVISOR AND OPERATING SYSTEM CHOICE:

- IaaS offerings have been based on heavily customized open-source Xen deployments. IaaS providers needed expertise in Linux, networking, virtualization, metering, resource management, and many other low-level aspects to successfully deploy and maintain their cloud offerings.

PLATFORM AS A SERVICE PROVIDERS

- Public Platform as a Service providers commonly offer a development and deployment environment that allow users to create and run their applications with little or no concern to low-level details of the platform.

FEATURES

- *Programming Models, Languages, and Frameworks.* Programming mod-els made available by IaaS providers define how users can express their applications using higher levels of abstraction and efficiently run them on the cloud platform.

- *Persistence Options.* A persistence layer is essential to allow applications to record their state and recover it in case of crashes, as well as to store user data.

SECURITY, PRIVACY, AND TRUST

- Security and privacy affect the entire cloud computing stack, since there is a massive use of third-party services and infrastructures that are used to host important data or to perform critical operations. In this scenario, the trust toward providers is fundamental to ensure the desired level of privacy for applications hosted in the cloud.

- When data are moved into the Cloud, providers may choose to locate them anywhere on the planet. The physical location of data centers determines the set of laws that can be applied to the management of data.

DATA LOCK-IN AND STANDARDIZATION

- The Cloud Computing Interoperability Forum (CCIF) was formed by organizations such as Intel, Sun, and Cisco in order to "enable a global cloud computing ecosystem whereby organizations are able to seamlessly work together for the purposes for wider industry adoption of cloud computing technology." The development of the Unified Cloud Interface (UCI) by CCIF aims at creating a standard programmatic point of access to an entire cloud infrastructure
  .
- In the hardware virtualization sphere, the Open Virtual Format (OVF) aims at facilitating packing and distribution of software to be run on VMs so that virtual appliances can be made portable

AVAILABILITY, FAULT-TOLERANCE, AND DISASTER RECOVERY

- It is expected that users will have certain expectations about the service level to be provided once their applications are moved to the cloud. These expectations include availability of the service, its overall performance, and what measures are to be taken when something goes wrong in the system or its components. In summary, users seek for a warranty before they can comfortably move their business to the cloud.

- SLAs, which include QoS requirements, must be ideally set up between customers and cloud computing providers to act as warranty. An SLA specifies the details of the service to be provided, including availability and performance guarantees. Additionally, metrics must be agreed upon by all parties, and penalties for violating the expectations must also be approved.

RESOURCE MANAGEMENT AND ENERGY-EFFICIENCY

- The multi-dimensional nature of virtual machines complicates the activity of finding a good mapping of VMs onto available physical hosts while maximizing user utility.

- Dimensions to be considered include: number of CPUs, amount of memory, size of virtual disks, and network bandwidth. Dynamic VM mapping policies may leverage the ability to suspend, migrate, and resume VMs as an easy way of preempting low-priority allocations in favor of higher-priority ones.

- Migration of VMs also brings additional challenges such as detecting when to initiate a migration, which VM to migrate, and where to migrate. In addition, policies may take advantage of live migration of virtual machines to relocate data center load without significantly disrupting running services.

MIGRATING INTO A CLOUD

- The promise of cloud computing has raised the IT expectations of small and medium enterprises beyond measure. Large companies are deeply debating it. Cloud computing is a disruptive model of IT whose innovation is part technology and part business model in short a "disruptive techno-commercial model" of IT.

- We propose the following definition of cloud computing: "It is a techno-business disruptive model of using distributed large-scale data centers either private or public or hybrid offering customers a scalable virtualized infrastructure or an abstracted set of services qualified by service-level agreements (SLAs) and charged only by the abstracted IT resources consumed."

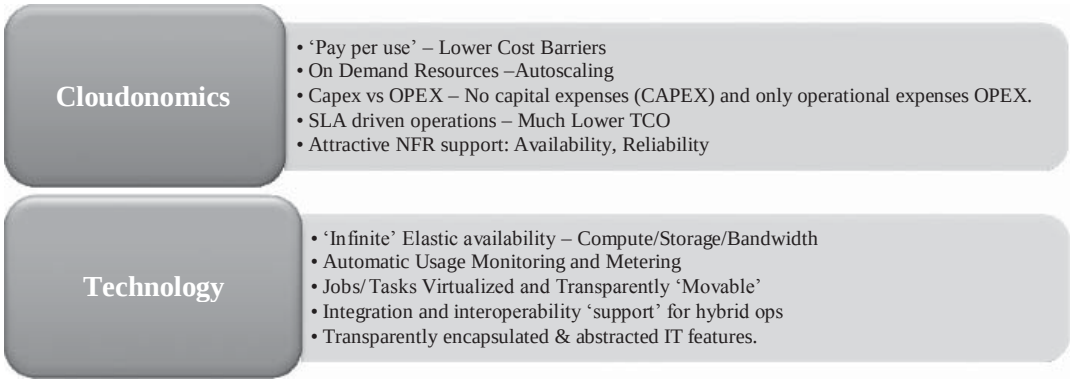


FIGURE 2.1. The promise of the cloud computing services.

- In Figure 2.1, the promise of the cloud both on the business front (the attractive cloudonomics) and the technology front widely aided the CxOs to spawn out several non-mission critical IT needs from the ambit of their captive traditional data centers to the appropriate cloud service.

- Several small and medium business enterprises, however, leveraged the cloud much beyond the cautious user. Many startups opened their IT departments exclusively using cloud services very successfully and with high ROI. Having observed these successes, several large enterprises have started successfully running pilots for leveraging the cloud.

- Many large enterprises run SAP to manage their operations. SAP itself is experimenting with running its suite of products: SAP Business One as well as SAP Netweaver on Amazon cloud offerings.

THE CLOUD SERVICE OFFERINGS AND DEPLOYMENT MODELS

- Cloud computing has been an attractive proposition both for the CFO and the CTO of an enterprise primarily due its ease of usage. This has been achieved by large data center service vendors or now better known as cloud service vendors again primarily due to their scale of operations.

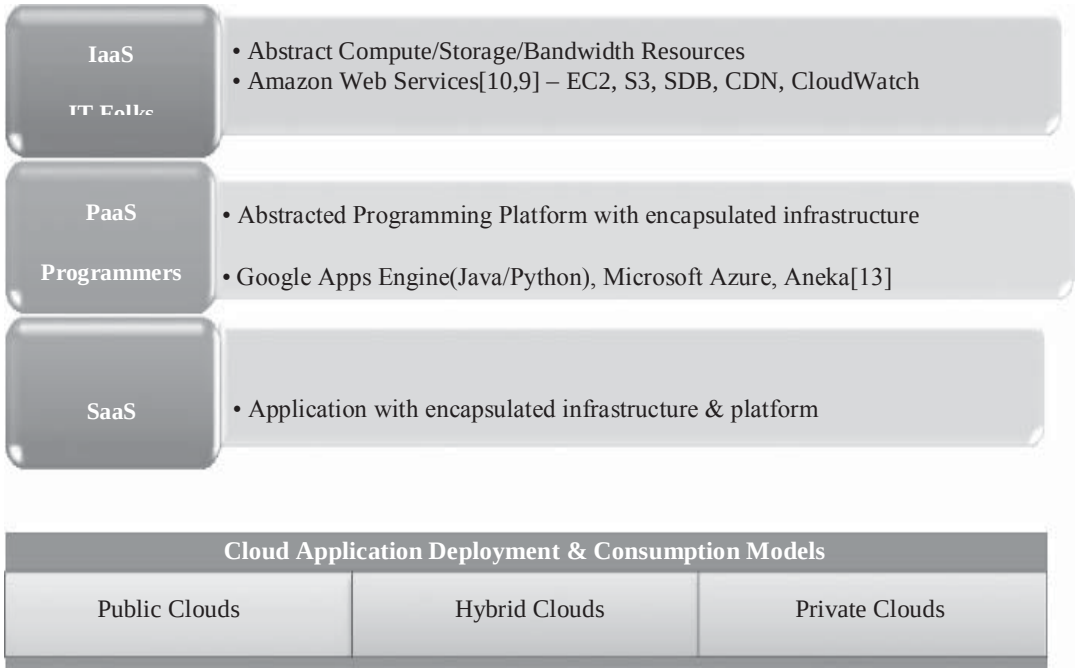


FIGURE 2.2. The cloud computing service offering and deployment models.

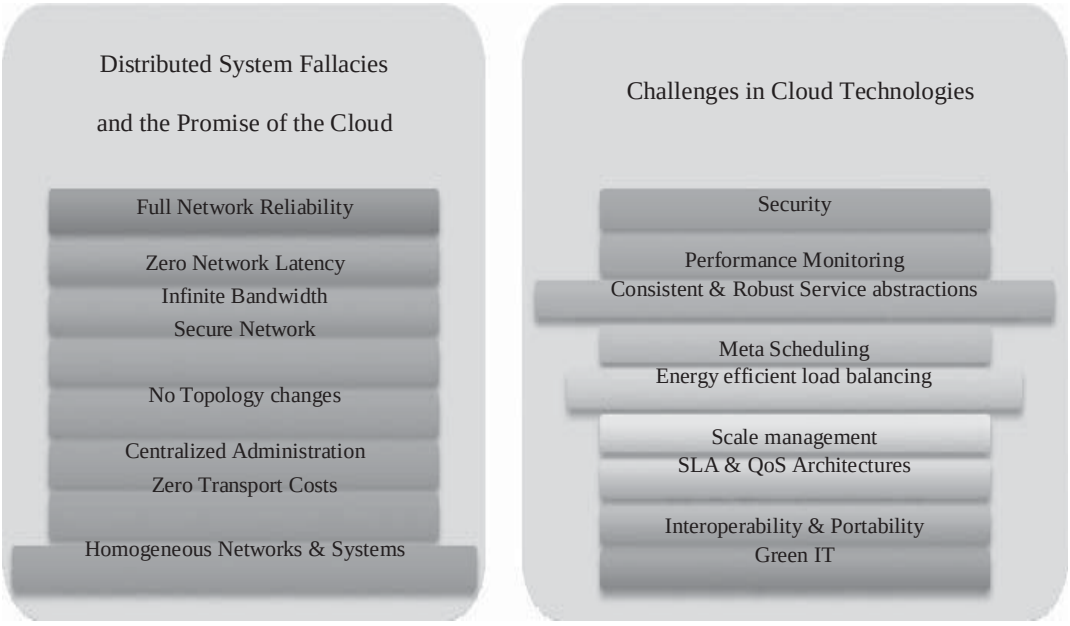


FIGURE 2.3. 'Under the hood' challenges of the cloud computing services implementations.

BROAD APPROACHES TO MIGRATING INTO THE CLOUD

- Cloud Economics deals with the economic rationale for leveraging the cloud and is central to the success of cloud-based enterprise usage. Decision-makers, IT managers, and software architects are faced with several dilemmas when planning for new Enterprise IT initiatives.

THE SEVEN-STEP MODEL OF MIGRATION INTO A CLOUD

- Typically migration initiatives into the cloud are implemented in phases or in stages. A structured and process-oriented approach to migration into a cloud has several advantages of capturing within itself the best practices of many migration projects.

1. Conduct Cloud Migration Assessments
2. Isolate the Dependencies

3. Map the Messaging & Environment

4. Re-architect & Implement the lost Functionalities

5. Leverage Cloud Functionalities & Features

6. Test the Migration

7. Iterate and Optimize

The Seven-Step Model of Migration into the Cloud. (*Source:* Infosys Research.)

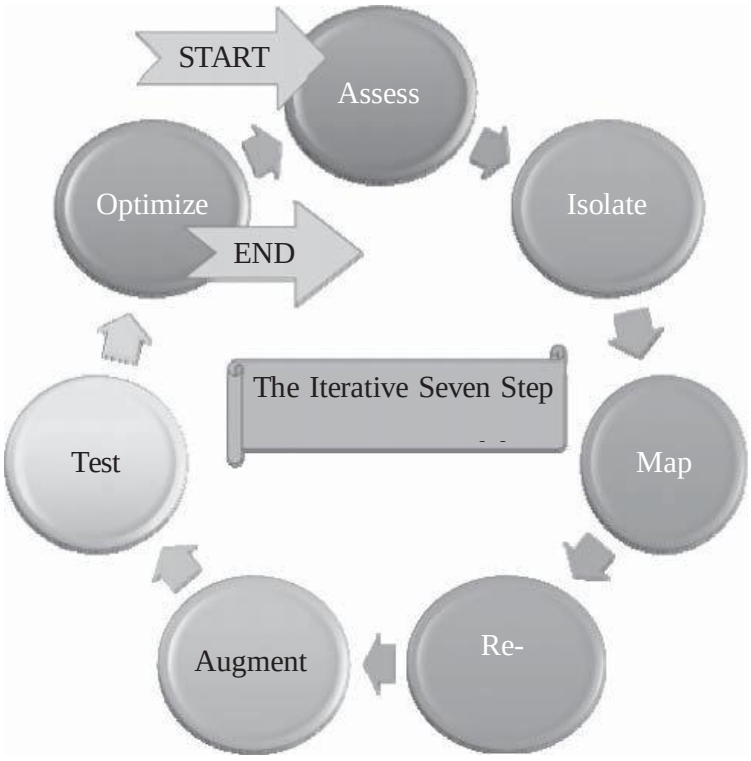


FIGURE 2.5. The iterative Seven-step Model of Migration into the Cloud. (*Source:* Infosys Research.)

Migration Risks and Mitigation

- The biggest challenge to any cloud migration project is how effectively the migration risks are identified and mitigated. In the Seven-Step Model of Migration into the Cloud, the process step of testing and validating includes efforts to identify the key migration risks. In the optimization step, we address various approaches to mitigate the identified migration risks.

- Migration risks for migrating into the cloud fall under two broad categories: the general migration risks and the security-related migration risks. In the former we address several issues including performance monitoring and tuning—essentially identifying all possible production level deviants; the business continuity and disaster recovery in the world of cloud computing service; the compliance with standards and governance issues; the IP and licensing issues; the quality of service (QoS) parameters as well as the corresponding SLAs committed to; the ownership, transfer, and storage of data in the application; the portability and interoperability issues which could help mitigate potential vendor lock-ins; the issues that result in trivializing and non comprehencing the complexities of migration that results in migration failure and loss of senior management's business confidence in these efforts.

ENRICHING THE 'INTEGRATION AS A SERVICE' PARADIGM FOR THE CLOUD ERA

THE EVOLUTION OF SaaS

- SaaS paradigm is on fast track due to its innate powers and potentials. Executives, entrepreneurs, and end-users are ecstatic about the tactic as well as strategic success of the emerging and evolving SaaS paradigm. A number of positive and progressive developments started to grip this model. Newer

resources and activities are being consistently readied to be delivered as a IT as a Service (ITaaS) is the most recent and efficient delivery method in the decisive IT landscape. With the meteoric and mesmerizing rise of the service orientation principles, every single IT resource, activity and infrastructure is being viewed and visualized as a service that sets the tone for the grand unfolding of the dreamt service era. This is accentuated due to the pervasive Internet.

- Integration as a service (IaaS) is the budding and distinctive capability of clouds in fulfilling the business integration requirements. Increasingly business applications are deployed in clouds to reap the business and technical benefits. On the other hand, there are still innumerable applications and data sources locally stationed and sustained primarily due to the security reason. The question here is how to create a seamless connectivity between those hosted and on-premise applications to empower them to work together.

- IaaS over- comes these challenges by smartly utilizing the time-tested business-to-business (B2B) integration technology as the value-added bridge between SaaS solutions and in-house business applications.

1. The Web is the largest digital information superhighway
2. The Web is the largest repository of all kinds of resources such as web pages, applications comprising enterprise components, business services, beans, POJOs, blogs, corporate data, etc.

3. The Web is turning out to be the open, cost-effective and generic business execution platform (E-commerce, business, auction, etc. happen in the web for global users) comprising a wider variety of containers, adaptors, drivers, connectors, etc.

4. The Web is the global-scale communication infrastructure (VoIP, Video conferencing, IP TV etc,)

5. The Web is the next-generation discovery, Connectivity, and integration middleware

Thus the unprecedented absorption and adoption of the Internet is the key driver for the continued success of the cloud computing.

THE CHALLENGES OF SaaS PARADIGM

As with any new technology, SaaS and cloud concepts too suffer a number of limitations. These technologies are being diligently examined for specific situations and scenarios. The prickling and tricky issues in different layers and levels are being looked into. The overall views are listed out below. Loss or lack of the following features deters the massive adoption of clouds

1. Controllability
2. Visibility & flexibility
3. Security and Privacy
4. High Performance and Availability
5. Integration and Composition
6. Standards

A number of approaches are being investigated for resolving the identified issues and flaws. Private cloud, hybrid and the latest community cloud are being prescribed as the solution for most of these inefficiencies and deficiencies. As rightly pointed out by someone in his weblogs, still there are miles to go. There are several companies focusing on this issue.

Integration Conundrum. While SaaS applications offer outstanding value in terms of features and functionalities relative to cost, they have introduced several challenges specific to integration. The first issue is that the majority of SaaS applications are point solutions and service one line of business.

APIs are Insufficient: Many SaaS providers have responded to the integration challenge by developing application programming interfaces (APIs). Unfortunately, accessing and managing data via an API requires a significant amount of coding as well as maintenance due to frequent API modifications and updates.

Data Transmission Security: SaaS providers go to great length to ensure that customer data is secure within the hosted environment. However, the need to transfer data from on-premise systems or applications behind the firewall with SaaS applications hosted outside of the client's data center poses new challenges that need to be addressed by the integration solution of choice.

***The Impacts of Cloud:*.** On the infrastructural front, in the recent past, the clouds have arrived onto the scene powerfully and have extended the horizon and the boundary of business applications, events and data. That is, business applications, development platforms etc. are getting moved to elastic, online and on-demand cloud infrastructures. Precisely speaking, increasingly for business, technical, financial and green reasons, applications and services are being readied and relocated to highly scalable and available clouds.

THE INTEGRATION METHODOLOGIES

Excluding the custom integration through hand-coding, there are three types for cloud integration

1. Traditional Enterprise Integration Tools can be empowered with special connectors to access Cloud-located Applications—This is the most likely approach for IT

organizations, which have already invested a lot in integration suite for their application integration needs.

2. Traditional Enterprise Integration Tools are hosted in the Cloud—This approach is similar to the first option except that the integration software suite is now hosted in any third-party cloud infrastructures so that the enterprise does not worry about procuring and managing the hardware or installing the integration software. This is a good fit for IT organizations that outsource the integration projects to IT service organizations and systems integrators, who have the skills and resources to create and deliver integrated systems.

Integration-as-a-Service (IaaS) or On-Demand Integration Offerings— These are SaaS applications that are designed to deliver the integration service securely over the Internet and are able to integrate cloud applications with the on-premise systems, cloud-to-cloud applications.

SaaS administrator or business analyst as the primary resource for managing and maintaining their integration work. A good example is Informatica On-Demand Integration Services.

In the integration requirements can be realised using any one of the following methods and middleware products.

1. Hosted and extended ESB (Internet service bus / cloud integration bus)
2. Online Message Queues, Brokers and Hubs
3. Wizard and configuration-based integration platforms (Niche integration solutions)
4. Integration Service Portfolio Approach
5. Appliance-based Integration (Standalone or Hosted)

CHARACTERISTICS OF INTEGRATION SOLUTIONS AND PRODUCTS.

The key attri-butes of integration platforms and backbones gleaned and gained from integration projects experience are connectivity, semantic mediation, Data mediation, integrity, security, governance etc

- Connectivity refers to the ability of the integration engine to engage with both the source and target systems using available native interfaces. This means leveraging the interface that each provides, which could vary from standards-based interfaces, such as Web services, to older and proprietary interfaces. Systems that are getting connected are very much responsible for the externalization of the correct information and the internalization of information once processed by the integration engine.

- Semantic Mediation refers to the ability to account for the differences between application semantics between two or more systems. Semantics means how information gets understood, interpreted and represented within information systems. When two different and distributed systems are linked, the differences between their own yet distinct semantics have to be covered.

- Data Mediation converts data from a source data format into destination data format. Coupled with semantic mediation, data mediation or data transformation is the process of converting data from one native format on the source system, to another data format for the target system.

- Data Migration is the process of transferring data between storage types, formats, or systems. Data migration means that the data in the old system is mapped to the new systems, typically leveraging data extraction and data loading technologies.

- Data Security means the ability to insure that information extracted from the source systems has to securely be placed into target systems. The integration method must leverage the native security systems of the source and target systems, mediate the differences, and provide the ability to transport the information safely between the connected systems.

- Data Integrity means data is complete and consistent. Thus, integrity has to be guaranteed when data is getting mapped and maintained during integration operations, such as data synchronization between on-premise and SaaS-based systems.

- Governance refers to the processes and technologies that surround a system or systems, which control how those systems are accessed and leveraged. Within the integration perspective, governance is about mana- ging changes to core information resources, including data semantics, structure, and interfaces.

THE ENTERPRISE CLOUD COMPUTING PARADIGM:

Relevant Deployment Models for Enterprise Cloud Computing

There are some general cloud deployment models that are accepted by the majority of cloud stakeholders today:

1)Public clouds are provided by a designated service provider for general public under a utility based pay-per-use consumption model. The cloud resources are hosted generally on the service provider's premises. Popular examples of public clouds are Amazon's AWS (EC2, S3 etc.), Rackspace Cloud Suite, and Microsoft's Azure Service Platform.

2)Private clouds are built, operated, and managed by an organization for its internal use only to support its business operations exclusively. Public,private, and government organizations worldwide are adopting

this model to exploit the cloud benefits like flexibility, cost reduction, agility and so on.

3) Virtual private clouds are a derivative of the private cloud deployment model but are further characterized by an isolated and secure segment of resources, created as an overlay on top of public cloud infrastructure using advanced network virtualization capabilities. Some of the public cloud vendors that offer this capability include Amazon Virtual Private Cloud , OpSource Cloud , and Skytap Virtual Lab.

4) Community clouds are shared by several organizations and support a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). They may be managed by the organizations or a third party and may exist on premise or off premise. One example of this is OpenCirrus formed by HP,Intel, Yahoo, and others.

5) Managed clouds arise when the physical infrastructure is owned by and/or physically located in the organization's data centers with an extension of management and security control plane controlled by the managed service provider. This deployment model isn't widely agreed upon, however, some vendors like ENKI and NaviSite's NaviCloud offers claim to be managed cloud offerings.

6) Hybrid clouds are a composition of two or more clouds (private, community,or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application.

UNIT-3

VIRTUAL MACHINES PROVISIONING AND MIGRATION SERVICES

ANALOGY FOR VIRTUAL MACHINE PROVISIONING:

- Historically, when there is a need to install a new server for a certain workload to provide a particular service for a client, lots of effort was exerted by the IT administrator, and much time was spent to install and provision a new server. 1) Check the inventory for a new machine, 2) get one, 3) format, install OS required, 4) and install services; a server is needed along with lots of security batches and appliances.

- Now, with the emergence of virtualization technology and the cloud computing IaaS model:

- It is just a matter of minutes to achieve the same task. All you need is to provision a virtual server through a self-service interface with small steps to get what you desire with the required specifications. 1) provisioning this machine in a public cloud like Amazon Elastic Compute Cloud (EC2), or 2) using a virtualization management software package or a private cloud management solution installed at your data center in order to provision the virtual machine inside the organization and within the private cloud setup.

Analogy for Migration Services:

- Previously, whenever there was a need for performing a server's upgrade or performing maintenance tasks, you would exert a lot of time and effort, because it is an expensive operation to maintain or upgrade a main server that has lots of applications and users.

- Now, with the advance of the revolutionized virtualization technology and migration services associated with

hypervisors' capabilities, these tasks (maintenance, upgrades, patches, etc.) are very easy and need no time to accomplish.

- Provisioning a new virtual machine is a matter of minutes, saving lots of time and effort, Migrations of a virtual machine is a matter of millisecondsVirtual Machine Provisioning and Manageability

VIRTUAL MACHINE LIFE CYCLE

- The cycle starts by a request delivered to the IT department, stating the requirement for creating a new server for a particular service.
- This request is being processed by the IT administration to start seeing the servers' resource pool, matching these resources with requirements
- Starting the provision of the needed virtual machine.
- Once it provisioned and started, it is ready to provide the required service according to an SLA.
- Virtual is being released; and free resources.

FIG.3.1 VMS LIFE CYCLE

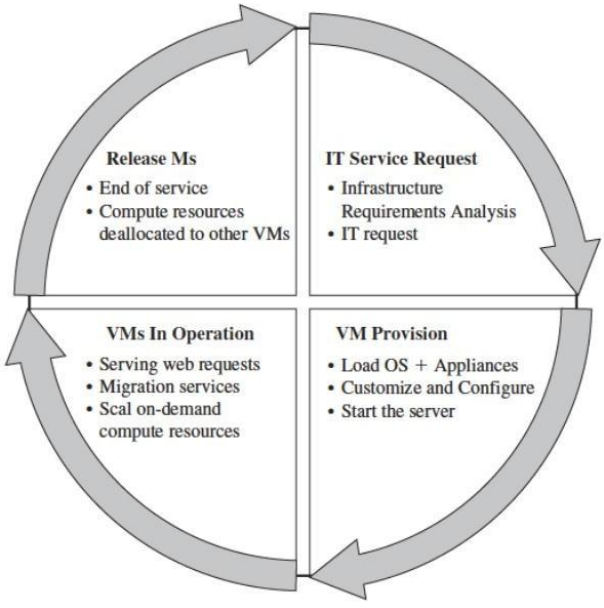


FIGURE 5.3. Virtual machine life cycle.

VM PROVISIONING PROCESS

- The common and normal steps of provisioning a virtual server are as follows:
- Firstly, you need to select a server from a pool of available servers (physical servers with enough capacity) along with the appropriate OS template you need to provision the virtual machine.
- Secondly, you need to load the appropriate software (operating System you selected in the previous step, device drivers, middleware, and the needed applications for the service required).
- Thirdly, you need to customize and configure the machine (e.g., IP address, Gateway) to configure an associated network and storage resources.
- Finally, the virtual server is ready to start with its newly loaded software.

- To summarize, server provisioning is defining server's configuration based on the organization requirements, a hardware, and software component (processor, RAM, storage, networking, operating system, applications, etc.).
- Normally, virtual machines can be provisioned by manually installing an operating system, by using a preconfigured VM template, by cloning an existing VM, or by importing a physical server or a virtual server from another hosting platform. Physical servers can also be virtualized and provisioned using P2V (Physical to Virtual) tools and techniques (e.g., virt- p2v).
- After creating a virtual machine by virtualizing a physical server, or by building a new virtual server in the virtual environment, a template can be created out of it.
- Most virtualization management vendors (VMware, XenServer, etc.) provide the data center's administration with the ability to do such tasks in an easy way.

LIVE MIGRATION AND HIGH AVAILABILITY

- **Live migration** (which is also called **hot or real-time migration**) can be defined as the movement of a virtual machine from one physical host to another while being powered on.

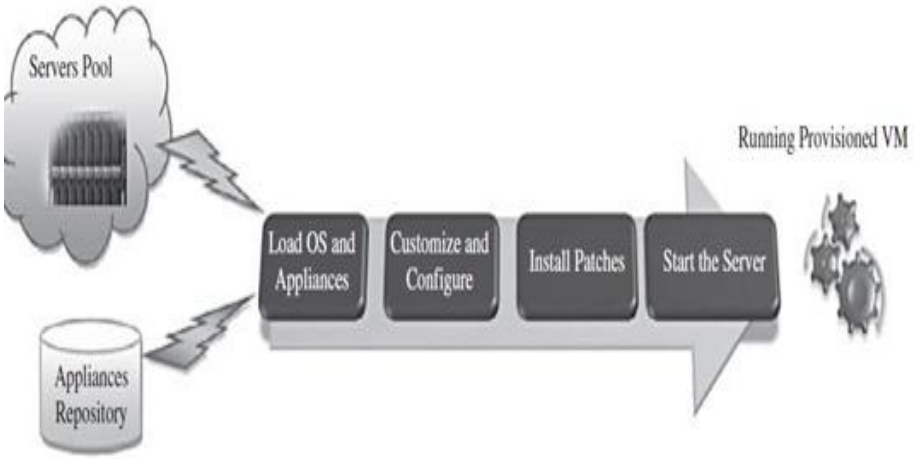


FIG.3.2 VIRTUAL MACHINE PROVISIONING PROCESS

ON THE MANAGEMENT OF VIRTUAL MACHINES FOR CLOUD INFRASTRUCTURES

THE ANATOMY OF CLOUD INFRASTRUCTURES

Here we focuses on the subject of IaaS clouds and, more specifically, on the efficient management of virtual machines in this type of cloud. There are many commercial IaaS cloud providers in the market, such as those cited earlier, and all of them share five characteristics:

(i) They provide on-demand provisioning of computational resources;

(ii) They use virtualization technologies to lease these resources;

(iii) They provide public and simple remote interfaces to manage those resources;

(iv)They use a pay-as-you-go cost model, typically charging by the hour;

(v) They operate data centers large enough to provide a seemingly unlimited amount of resources to their clients (usually touted as "infinite capacity" or "unlimited elasticity"). Private and hybrid clouds share these same characteristics, but instead of selling capacity over publicly accessible interfaces, focus on providing capacity to an organization's internal users.

DISTRIBUTED MANAGEMENT OF VIRTUAL INFRASTRUCTURES

VM Model and Life Cycle in OpenNebula The life cycle of a VM within OpenNebula follows several stages:

- Resource Selection. Once a VM is requested to OpenNebula, a feasible placement plan for the VM must be made. OpenNebula's default scheduler provides an implementation of a rank scheduling policy, allowing site administrators to configure the scheduler to prioritize the resources that are more suitable for the VM, using information from the VMs and the physical hosts. In addition, OpenNebula can also use the Haizea lease manager to support more complex scheduling policies.

- Resource Preparation. The disk images of the VM are transferred to the target physical resource. During the boot process, the VM is contextualized, a process where the disk images are specialized to work in a given environment. For example, if the VM is part of a group of VMs offering a service (a compute cluster, a DB-based application, etc.), contextualization could involve setting up the network and the machine hostname, or registering the new VM with a service (e.g., the head node in a compute cluster). Different techniques are available to contextualize a worker node, including use of an automatic installation system (for instance, Puppet or Quattor), a context server, or access to a disk image with the context data for the worker node (OVF recommendation).

- VM Termination. When the VM is going to shut down, OpenNebula can transfer back its disk images to a known location. This way, changes in the VM can be kept for a future use.

ENHANCING CLOUD COMPUTING ENVIRONMENTS USING A CLUSTER AS A SERVICE

RVWS DESIGN

Dynamic Attribute Exposure

- There are two categories of dynamic attributes addressed in the RVWS framework: state and characteristic. State attributes cover the current activity of the service and its resources, thus indicating readiness. For example, a Web service that exposes a cluster (itself a complex resource) would most likely have a dynamic state attribute that indicates how many nodes in the cluster are busy and how many are idle.

- Characteristic attributes cover the operational features of the service, the resources behind it, the quality of service (QoS), price and provider informa- tion. Again with the cluster Web service example, a possible characteristic is an array of support software within the cluster. This is important information as cluster clients need to know what software libraries exist on the cluster.

- To keep the stateful Web service current, a Connector [2] is used to detect changes in resources and then inform the Web service. The Connector has three logical modules: Detection, Decision, and Notification. The Detection module routinely queries the resource for attribute information. Any changes in the attributes are passed to the Decision module (3) that decides if the attribute change is large enough to warrant a notification. This prevents excessive communication with the Web service. Updated attributes are passed on to the Notification module (4), which informs the stateful Web service (5) that updates its internal state. When clients requests the stateful WSDL document (6), the Web service returns the WSDL document with the values of all attributes (7) at the request time.

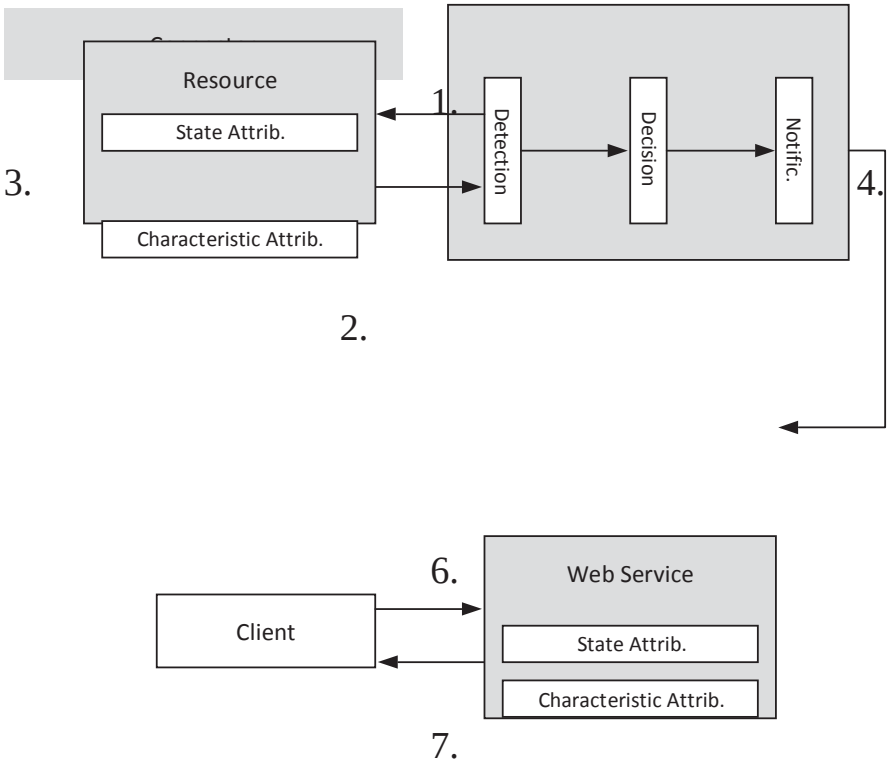


FIGURE 3.3. Exposing resource attributes.

CLUSTER AS A SERVICE: THE LOGICAL DESIGN

Simplification of the use of clusters could only be achieved through higher layer abstraction that is proposed here to be implemented using the service-based Cluster as a Service (CaaS) Technology. The purpose of the CaaS Technology is to ease the publication, discovery, selection, and use of existing computa- tional clusters.

CaaS Overview

- The exposure of a cluster via a Web service is intricate and comprises several services running on top of a physical cluster. Figure 7.6 shows the complete CaaS technology.

A typical cluster is comprised of three elements:

nodes, data storage, and middleware. The middleware virtualizes the cluster into a single system image; thus resources such as the CPU can be used without knowing the organization of the cluster. Of interest to this chapter are the components that manage the allocation of jobs to nodes (scheduler) and that monitor the activity of the cluster (monitor). As time progresses, the amount of free memory, disk space, and CPU usage of each cluster node changes. Information about how quickly the scheduler can take a job and start it on the cluster also is vital in choosing a cluster.

- To make information about the cluster publishable, a Publisher Web service and Connector were created using the RVWS framework. The purpose of the publisher Web service was to expose the dynamic attributes of the cluster via the stateful WSDL document. Furthermore, the Publisher service is published to the Dynamic Broker so clients can easily discover the cluster.

- To find clusters, the CaaS Service makes use of the Dynamic Broker. While the Broker is detailed in returning dynamic attributes of matching services, the results from the Dynamic Broker are too detailed for the CaaS Service. Thus another role of the CaaS Service is to "summarize" the result data so that they convey fewer details.

- Ordinarily, clients could find required clusters but they still had to manually transfer their files, invoke the scheduler, and get the results back. All three tasks require knowledge of the cluster and are conducted using complex tools. The role of the CaaS Service is to
  (i) provide easy and intuitive file transfer tools so clients can upload jobs and download results and
  (ii) offer an easy to use interface for clients to monitor their jobs. The CaaS Service does this by allowing clients to upload files as they would any Web page while carrying out the required data transfer to the cluster transparently.

Because clients to the cluster cannot knowhow the data storage is managed, the CaaS Service offers a simple transfer interface to clients while addressing the transfer specifics. Finally, the CaaS Service communicates with the cluster's scheduler, thus freeing the client from needing to know how the scheduler is invoked when submitting and monitoring jobs.

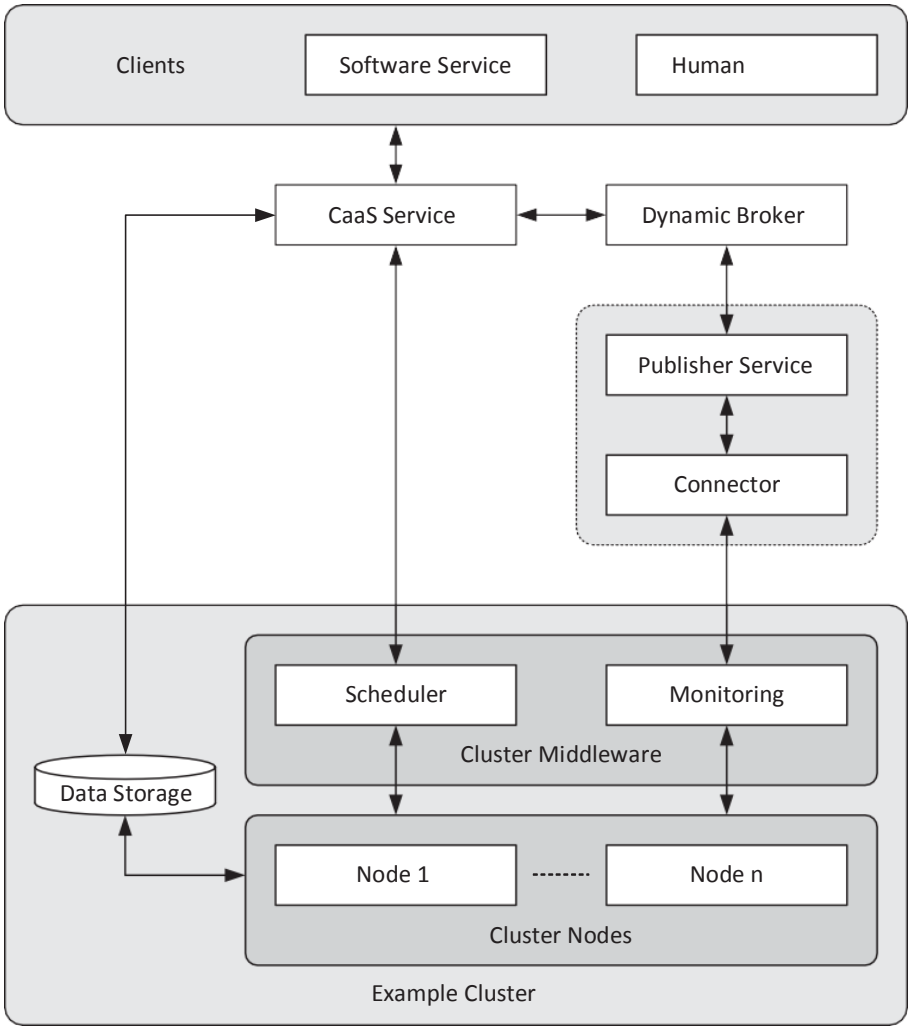


FIGURE 3.4 Complete CaaS system.

- It allowing clients to upload files as they would any Web

page while carrying out the required data transfer to the cluster transparently.

Because clients to the cluster cannot know how the data storage is managed, the CaaS Service offers a simple transfer interface to clients while addressing the transfer specifics. Finally, the CaaS Service communicates with the cluster's scheduler, thus freeing the client from needing to know how the scheduler is invoked when submitting and monitoring jobs.

Cluster Discovery

Before a client uses a cluster, a cluster must be discovered and selected first. Figure 3.5 shows the workflow on finding a required cluster. To start, clients submit cluster requirements in the form of attribute values to the CaaS Service Interface (1). The requirements range from the number of nodes in the cluster to the installed software (both operating systems and software APIs). The CaaS Service Interface invokes the Cluster Finder module

(2) that communicates with the Dynamic Broker

(3) and returns service matches (if any).

To address the detailed results from the Broker, the Cluster Finder module invokes the Results Organizer module (4) that takes the Broker results and returns an organized version that is returned to the client (5—6). The organized

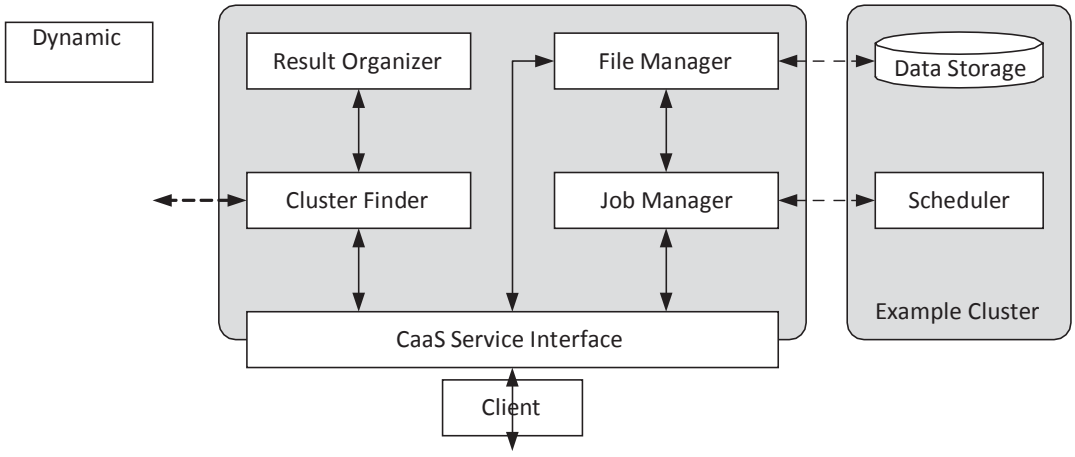


FIGURE 3.5 CaaS Service design.

results instruct the client what clusters satisfy the specified requirements. After reviewing the results, the client chooses a cluster.

Job Submission. After selecting a required cluster, all executables and data files have to be transferred to the cluster and the job submitted to the scheduler for execution. As clusters vary significantly in the software middleware used to create them, it can be difficult to place jobs on the cluster. To do so requires knowing how jobs are stored and how they are queued for execution on the cluster

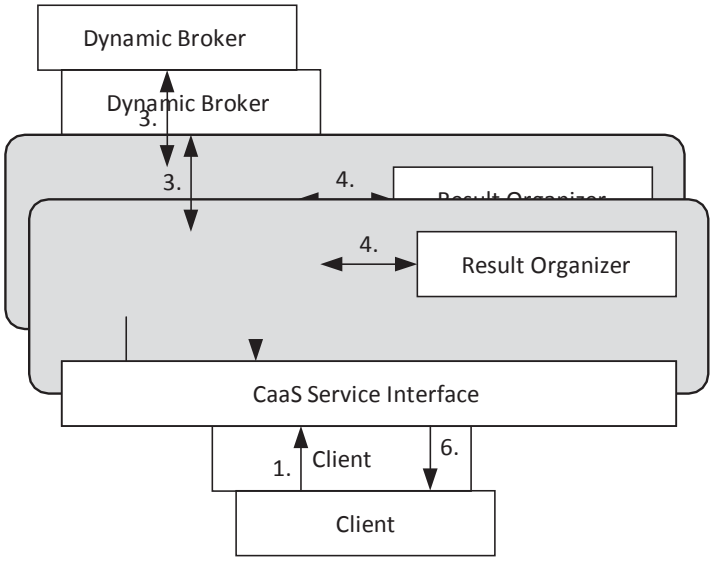


FIGURE 3.6. Cluster discovery.

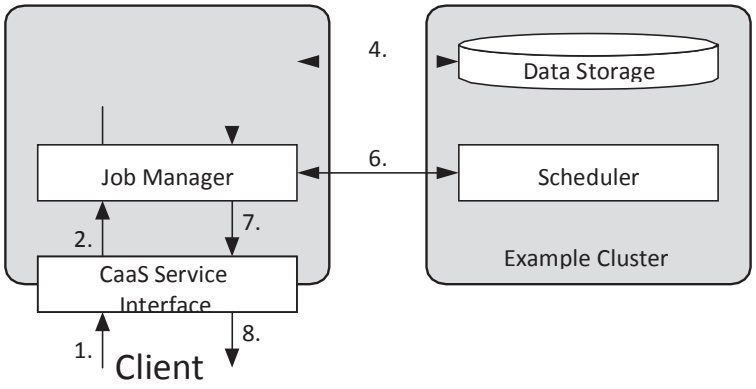


FIGURE 3.7. Job submission.

Job Monitoring. During execution, clients should be able to view the execution progress of their jobs. Even though the cluster is not the owned by the client, the job is. Thus, it is the right of the client to see how the job is progressing and (if the client decides) terminate the job and remove it from the cluster

Result Collection. The final role of the CaaS Service is addressing jobs that have terminated or completed their execution successfully. In both

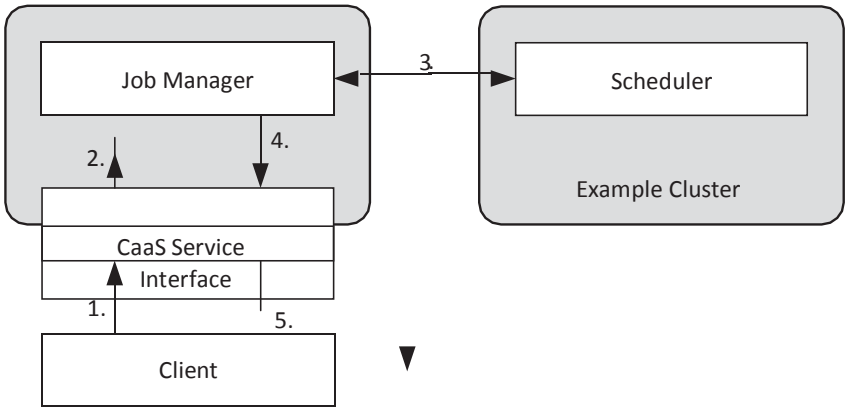


FIGURE 3.8. Job result collection

cases, error or data files need to be transferred to the client. Figure 3.8 presents the workflow and CaaS Service modules used to retrieve error or result files from the cluster.

Clients start the error or result file transfer by contacting the CaaS Service Interface (1) that then invokes the File Manager (2) to retrieve the files from the cluster's data storage (3). If there is a transfer error, the File Manager attempts to resolve the issue first before informing the client. If the transfer of files (3) is successful, the files are returned to the CaaS Service Interface (4) and then the client (5). When returning the files, URL link or a FTP address is provided so the client can retrieve the files.

SECURE DISTRIBUTED DATA STORAGE IN CLOUD COMPUTING

CLOUD STORAGE: FROM LANs TO WANs

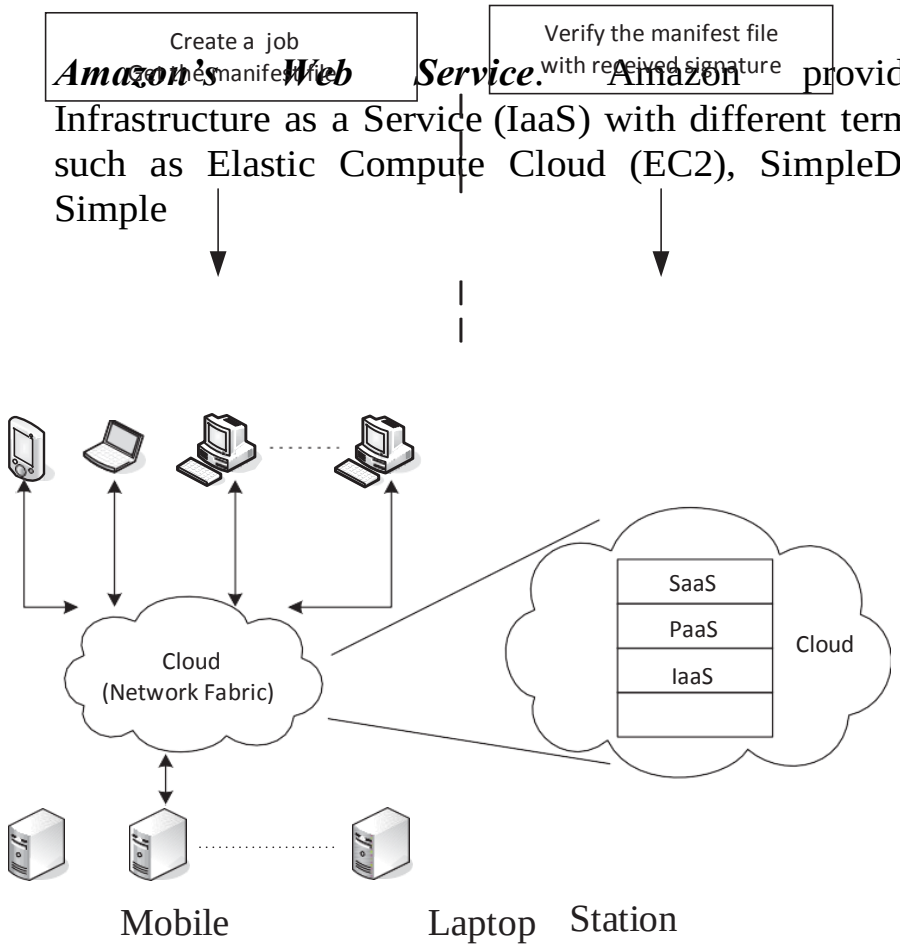
- Cloud computing has been viewed as the future of the IT industry. It will be a revolutionary change in computing services. Users will be allowed to purchase CPU cycles, memory utilities, and information storage services conveniently just like how we pay our monthly water and electricity bills. However, this image will not become realistic until some challenges have been addressed. In this section, we will briefly introduce the major difference brought by distributed data storage in cloud computing environment. Then, vulnerabilities in today's cloud computing platforms are analyzed and illustrated.

Most designs of distributed storage take the form of either storage area networks (SANs) or network-attached storage (NAS) on the LAN level

- Such as the networks of an enterprise, a campus, or an organization. SANs are constructed on top of block-addressed storage units connected through dedicated high-speed networks. In contrast, NAS is implemented by attaching specialized file servers to a TCP/IP network and providing a file-based interface to client machine . For SANs and NAS, the distributed storage nodes are managed by the same authority. The system administrator has control over each node, and essentially the security level of data is under control. The reliability of such systems is often achieved by redundancy, and the storage security is highly dependent on the security of the system against the attacks and intrusion from outsiders. The confidentiality and integrity of data are mostly achieved using robust cryptographic schemes.

Existing Commercial Cloud Services

- In normal network-based applications, user authentication, data confidenti- ality, and data integrity can be solved through IPSec proxy using encryption and digital signature. The key exchanging issues can be solved by SSL proxy. These methods have been applied to today's cloud computing to secure the data on the cloud and also secure the communication of data to and from the cloud. The service providers claim that their services are secure. This section describes three secure methods used in three commercial cloud services and discusses their vulnerabilities.



Amazon's Web Service. Amazon provides Infrastructure as a Service (IaaS) with different terms, such as Elastic Compute Cloud (EC2), SimpleDB, Simple

FIGURE 3.9 Storage Server Farm

Storage Service (S3), and so on. They are supposed to ensure the confidenti- ality, integrity, and availability of the customers' applications and data. Figure 3.9 presents one of the data processing methods adopted in Amazon's AWS , which is used to transfer large amounts of data between the AWS cloud and portable storage devices.

The downloading process is similar to the uploading process. The user creates a manifest and signature file, e-mails the manifest file, and ships the storage device attached with signature file. When Amazon receives these two files, it will validate the two files, copy the data into the storage device, ship it back, and e-mail to the user with the status including the MD5 checksum of the data. Amazon claims that the maximum security is obtained via SSL endpoints.

Microsoft Windows Azure. The Windows Azure Platform (Azure) is an Internet-scale cloud services platform hosted in Microsoft data centers, which provides an operating system and a set of developer services that can be used individually or together [8]. The platform also provides scalable storage service. There are three basic data items: blobs (up to 50 GB), tables, and queues (,8k). In the Azure Storage, based on the blob, table, and queue structures, Microsoft promises to achieve confidentiality of the users' data.

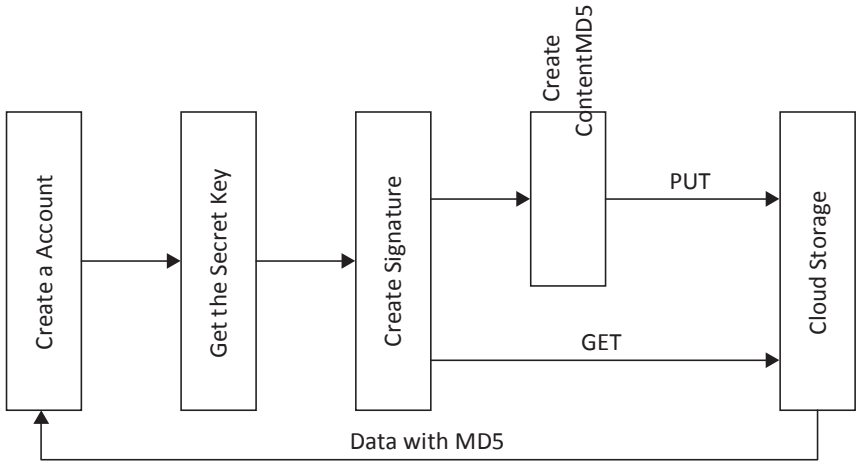


FIGURE 3.10. Security data access procedure.

TECHNOLOGIES FOR DATA SECURITY IN CLOUD COMPUTING

This section presents several technologies for data security and privacy in cloud computing. Focusing on the unique issues of the cloud data storage platform, this section does not repeat the normal approaches that provide confidentiality, integrity, and availability in distributed data storage applications. Instead, we select to illustrate the unique requirements for cloud computing data security from a few different perspectives:

- *Database Outsourcing and Query Integrity Assurance.* Researchers have pointed out that storing data into and fetching data from devices and machines behind a cloud are essentially a novel form of database outsourcing.
- *Data Integrity in Untrustworthy Storage*. One of the main challenges that prevent end users from adopting cloud storage services is the fear of losing data or data corruption. It is critical to relieve the users' fear by providing technologies that enable users to check the integrity of their data.
- *Web-Application-Based Security.* Once the dataset is stored remotely, a Web browser is one of the most convenient approaches that end users can use to access their data on remote services. In the era of cloud computing, Web security plays a more important role than ever.
- *Multimedia Data Security.* With the development of high-speed network technologies and large bandwidth connections, more and more multi- media data are being stored and shared in cyber space. The security requirements for video, audio, pictures, or images are different from other applications.

DATABASE OUTSOURCING AND QUERY INTEGRITY ASSURANCE

- In recent years, database outsourcing has become an important component of cloud computing. Due to the rapid advancements in network technology, the cost of transmitting a terabyte of data over long distances has decreased significantly in the past decade.
- In addition, the total cost of data management is five to ten times higher than the initial acquisition costs. As a result, there is a growing interest in outsourcing database management tasks to third parties that can provide these tasks for a much lower cost due to the economy of scale.
- This new outsourcing model has the benefits of reducing the costs for running Database Management Systems demonstrates the general architecture of a database outsourcing environment with clients.
- The database owner outsources its data management tasks, and clients send queries to the untrusted service provider. Let *T* denote the data to be outsourced. The data *T* are is preprocessed, encrypted, and stored at the service provider. For evaluating queries, a user rewrites a set of queries *Q* against *T* to queries against the encrypted database.
- The outsourcing of databases to a third-party service provider was first introduced by Hacigu¨ mu¨s et al. Generally, there are two security concems

Query Integrity Assurance.

- In addition to data privacy, an important security concern in the database outsourcing paradigm is query integrity. Query integrity examines the trustworthiness of the hosting environment.

- When a client receives a query result from the service provider, it wants to be assured that the result is both correct and complete, where correct means that the result must originate in the owner's data and not has been tampered with, and complete means that the result includes all records satisfying the query.

Data Integrity in Untrustworthy Storage

- While the transparent cloud provides flexible utility of network-based resources, the fear of loss of control on their data is one of the major concerns that prevent end users from migrating to cloud storage services.
- Actually it is a potential risk that the storage infrastructure providers become self-interested, untrustworthy, or even malicious.
- There are different motivations whereby a storage service provider could become untrustworthy—for instance, to cover the consequence of a mistake in operation, or deny the vulnerability in the system after the data have been stolen by an adversary. This section introduces two technologies to enable data owners to verify the data integrity while the files are stored in the remote untrustworthy storage services.

- Note that the *verifier* could be either the data owner or a trusted third party, and the *prover* could be the storage service provider or storage medium owner or system administrator.

    - *Requirement #1. It should not be a pre-requirement that the verifier has to possess a complete copy of the data to be checked. And in practice, it does not make sense for a verifier to keep a duplicated copy of the content to be verified. As long as it serves the purpose well, storing a more concise contents digest of the data at the verifier should be enough.*

- *Requirement #2*. The protocol has to be very robust considering the untrustworthy prover. A malicious prover is motivated to hide the violation of data integrity. The protocol should be robust enough that such a prover ought to fail in convincing the verifier.

- *Requirement #3*. The amount of information exchanged during the verification operation should not lead to high communication overhead.

- *Requirement #4*. The protocol should be computationally efficient.

- *Requirement #5*. It ought to be possible to run the verification an unlimited number of times.

A PDP-Based Integrity Checking Protocol.

- Ateniese et al proposed a protocol based on the provable data procession (PDP) technology, which allows users to obtain a probabilistic proof from the storage service providers. Such a proof will be used as evidence that their data have been stored there. One of the advantages of this protocol is that the proof could be generated by the storage service provider by accessing only a small portion of the whole dataset. At the same time, the amount of the metadata that end users are required to store is also small—that is, O(1). Additionally, such a small amount data exchanging procedure lowers the overhead in the communication channels too.

- As part of pre-processing procedure, the data owner (client) may conduct operations on the data such as expanding the data or generating additional metadata to be stored at the cloud server side. The data owner could execute the PDP protocol before the local copy is deleted to ensure that the uploaded copy has been stored at the server machines successfully. Actually, the data owner may encrypt a dataset before transferring them to the storage machines.

Web-Application-Based Security

- In cloud computing environments, resources are provided as a service over the Internet in a dynamic, virtualized, and scalable way . Through cloud computing services, users access business applications on-line from a Web browser, while the software and data are stored on the servers. Therefore, in the era of cloud computing, Web security plays a more important role than ever. The Web site server is the first gate that guards the vast cloud resources. Since the cloud may operate continuously to process millions of dollars' worth of daily on-line transactions, the impact of any Web security vulnerability will be amplified at the level of the whole cloud.

- Web attack techniques are often referred as the class of attack. When any Web security vulnerability is identified, attacker will employ those techniques to take advantage of the security vulnerability. The types of attack can be categorized in Authentication, Authorization, Client-Side Attacks, Comm- and Execution, Information Disclosure, and Logical Attacks . Due to the limited space, this section introduces each of them briefly. Interested read- ers are encouraged to explore for more detailed information from the materials cited.

- **Authentication.** Authentication is the process of verifying a claim that a subject made to act on behalf of a given principal. Authentication attacks target a Web site's method of validating the identity of a user, service, or application, including Brute Force, Insufficient Authentication, and Weak Password Recovery Validation. Brute Force attack employs an automated process to guess a person's username and password by trial and error.

- In the Insufficient Authentication case, some sensitive content or functionality are protected by "hiding" the specific location in obscure string but still remains

accessible directly through a specific URL. The attacker could discover those URLs through a Brute Force probing of files and directories. Many Web sites provide password recovery service. This service will automatically recover the user name or password to the user if she or he can answer some questions defined as part of the user registration process. If the recovery questions are either easily guessed or can be skipped, this Web site is considered to be Weak Password Recovery Validation.

- **Authorization.** Authorization is used to verify if an authenticated subject can perform a certain operation. Authentication must precede authorization. For example, only certain users are allowed to access specific content or functionality.

  Authorization attacks use various techniques to gain access to protected areas beyond their privileges. One typical authorization attack is caused by Insufficient Authorization. When a user is authenticated to a Web site, it does not necessarily mean that she should have access to certain content that has been granted arbitrarily. Insufficient authorization occurs when a Web site does not protect sensitive content or functionality with proper access control restrictions. Other authorization attacks are involved with session. Those attacks include Credential/Session Prediction, Insufficient Session Expiration, and Session Fixation.

- In many Web sites, after a user successfully authenticates with the Web site for the first time, the Web site creates a session and generate a unique "session ID" to identify this session. This session ID is attached to subsequent requests to the Web site as "Proof" of the authenticated session.

- Credential/Session Prediction attack deduces or guesses the unique value of a session to hijack or impersonate a user.

- Insufficient Session Expiration occurs when an attacker is allowed to reuse old session credentials or session IDs

for authorization. For example, in a shared computer, after a user accesses a Web site and then leaves, with Insufficient Session Expiration, an attacker can use the browser's back button to access Web pages previously accessed by the victim.

- Session Fixation forces a user's session ID to an arbitrary value via Cross- Site Scripting or peppering the Web site with previously made HTTP requests. Once the victim logs in, the attacker uses the predefined session ID value to impersonate the victim's identity.

- **Client-Side Attacks.** The Client-Side Attacks lure victims to click a link in a malicious Web page and then leverage the trust relationship expectations of the victim for the real Web site. In Content Spoofing, the malicious Web page can trick a user into typing user name and password and will then use this information to impersonate the user.

    - Cross-Site Scripting (XSS) launches attacker-supplied executable code in the victim's browser. The code is usually written in browser-supported scripting languages.
    - Languages such as JavaScript, VBScript, ActiveX, Java, or Flash. Since the code will run within the security context of the hosting Web site, the code has the ability to read, modify, and transmit any sensitive data, such as cookies, accessible by the browser.
    - Cross-Site Request Forgery (CSRF) is a serve security attack to a vulnerable site that does not take the checking of CSRF for the HTTP/HTTPS request. Assuming that the attacker knows the URLs of the vulnerable site which are not protected by CSRF checking and the victim's browser stores credentials such as cookies of the vulnerable site, after luring the victim to click a link in a malicious Web page, the attacker can forge the victim's identity and access the vulnerable Web site on victim's behalf.

- **Command Execution.** The Command Execution attacks exploit server-side vulnerabilities to execute remote commands on the Web site. Usually, users supply inputs to the Web-site to request services.

- If a Web application does not properly sanitize user-supplied input before using it within application code, an attacker could alter command execution on the server.

- For example, if the length of input is not checked before use, buffer overflow could happen and result in denial of service. Or if the Web application uses user input to construct statements such as SQL, XPath, C/C11 Format String, OS system command, LDAP, or dynamic HTML, an attacker may inject arbitrary executable code into the server if the user input is not properly filtered.

- **Information Disclosure**. The Information Disclosure attacks acquire sensi- tive information about a web site revealed by developer comments, error messages, or well-know file name conventions. For example, a Web server may return a list of files within a requested directory if the default file is not present. This will supply an attacker with necessary information to launch further attacks against the system. Other types of Information Disclosure includes using special paths such as "." and ".." for Path Traversal, or uncovering hidden URLs via Predictable Resource Location.

- ***Logical Attacks.*** Logical Attacks involve the exploitation of a Web applica- tion's logic flow. Usually, a user's action is completed in a multi-step process. The procedural workflow of the process is called application logic. A common Logical Attack is Denial of Service (DoS). DoS attacks will attempt to consume all available resources in the Web server such as CPU, memory, disk space, and so on, by abusing the functionality provided by the Web site. When any one of any system resource

reaches some utilization threshold, the Web site will no long be responsive to normal users. DoS attacks are often caused by Insufficient Anti-automation where an attacker is permitted to automate a process repeatedly. An automated script could be executed thousands of times a minute, causing potential loss of performance or service.

Multimedia Data Security Storage

- With the rapid developments of multimedia technologies, more and more multimedia contents are being stored and delivered over many kinds of devices, databases, and networks. Multimedia Data Security plays an important role in the data storage to protect multimedia data. Recently, how storage multimedia contents are delivered by both different providers and users has attracted much attentions and many applications. This section briefly goes through the most critical topics in this area.

Protection from Unauthorized Replication.

- Contents replication is requi- red to generate and keep multiple copies of certain multimedia contents. For example, content distribution networks (CDNs) have been used to manage content distribution to large numbers of users, by keeping the replicas of the same contents on a group of geographically distributed surrogates .

- Although the replication can improve the system performance, the unauthor- ized replication causes some problems such as contents copyright, waste of replication cost, and extra control overheads.

Protection from Unauthorized Replacement.

- As the storage capacity is limited, a replacement process must be carried out when the capacity exceeds its limit. It means the situation that a currently stored content must be removed from the storage space in order to make space for the new coming content. However, how to decide which content should be removed is very important. If an unauthorized replacement happens, the content which the user doesn't want to delete will be removed resulting in an accident of the data loss. Furthermore, if the important content such as system data is removed by unauthorized replacement, the result will be more serious.

Protection from Unauthorized Pre-fetching.

- The Pre-fetching is widely deployed in Multimedia Storage Network Systems between server databases and end users' storage disks . That is to say, If a content can be predicted to be requested by the user in future requests, this content will be fetched from the server database to the end user before this user requests it, in order to decrease user response time. Although the Pre-fetching shows its efficiency, the un- authorized pre-fetching should be avoided to make the system to fetch the necessary content.

UNIT-4

MONITORING AND MANAGEMENT:

AN ARCHITECTURE FOR FEDERATED CLOUD COMPUTING

THE BASIC PRINCIPLES OF CLOUD COMPUTING

In this section we unravel a set of principles that enable Internet scale cloud computing services. We seek to highlight the fundamental requirement from the providers of cloud computing to allow virtual applications to freely migrate, grow, and shrink.

Federation

- All cloud computing providers, regardless of how big they are, have a finite capacity. To grow beyond this capacity, cloud computing providers should be able to form federations of providers such that they can collaborate and share their resources. The need for federation-capable cloud computing offerings is also derived from the industry trend of adopting the cloud computing paradigm internally within companies to create *private clouds* and then being able to extend these clouds with resources leased on-demand from *public clouds*.

Independence

- Just as in other utilities, where we get service without knowing the internals of the utility provider and with standard equipment not specific to any provider (e.g., telephones), for cloud computing services to really fulfill the computing as a utility vision, we need to offer cloud computing users full independence. Users should be able to use the services of the cloud without relying on any

provider- specific tool, and cloud computing providers should be able to manage their infrastructure without exposing internal details to their customers or partners. As a consequence of the independence principle, all cloud services need to be encapsulated and generalized such that users will be able to acquire equivalent

- virtual resources at different providers.

Isolation

- Cloud computing services are, by definition, hosted by a provider that will simultaneously host applications from many different users. For these users to move their computing into the cloud, they need warranties from the cloud computing provider that their stuff is completely isolated from others. Users must be ensured that their resources cannot be accessed by others sharing the same cloud and that adequate performance isolation is in place to ensure that no other user may possess the power to directly effect the service granted to their application.

Elasticity

- One of the main advantages of cloud computing is the capability to provide, or release, resources on-demand. These "elasticity" capabilities should be enacted automatically by cloud computing providers to meet demand variations, just as electrical companies are able (under normal operational circumstances) to automatically deal with variances in electricity consumption levels. Clearly the behavior and limits of automatic growth and shrinking should be driven by contracts and rules agreed on between cloud computing providers and consumers.

Trust

- Probably the most critical issue to address before cloud computing can become the preferred computing paradigm

is that of establishing trust. Mechanisms to build and maintain trust between cloud computing consumers and cloud computing providers, as well as between cloud computing providers among themselves, are essential for the success of any cloud computing offering.

SLA MANAGEMENT IN CLOUD COMPUTING: A SERVICE PROVIDER'S PERSPECTIVE

TRADITIONAL APPROACHES TO SLO MANAGEMENT

Traditionally, load balancing techniques and admission control mechanisms have been used to provide guaranteed quality of service (QoS) for hosted web applications. These mechanisms can be viewed as the first attempt towards managing the SLOs. In the following subsections we discuss the existing approaches for load balancing and admission control for ensuring QoS.

Load Balancing

The objective of a load balancing is to distribute the incoming requests onto a set of physical machines, each hosting a replica of an application.

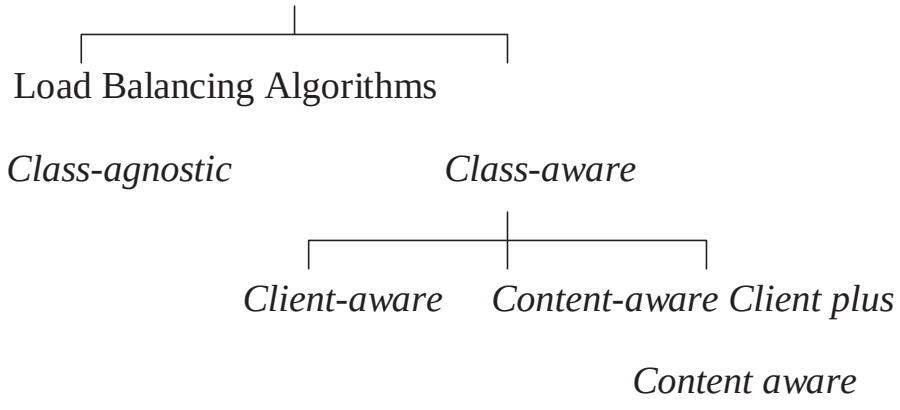
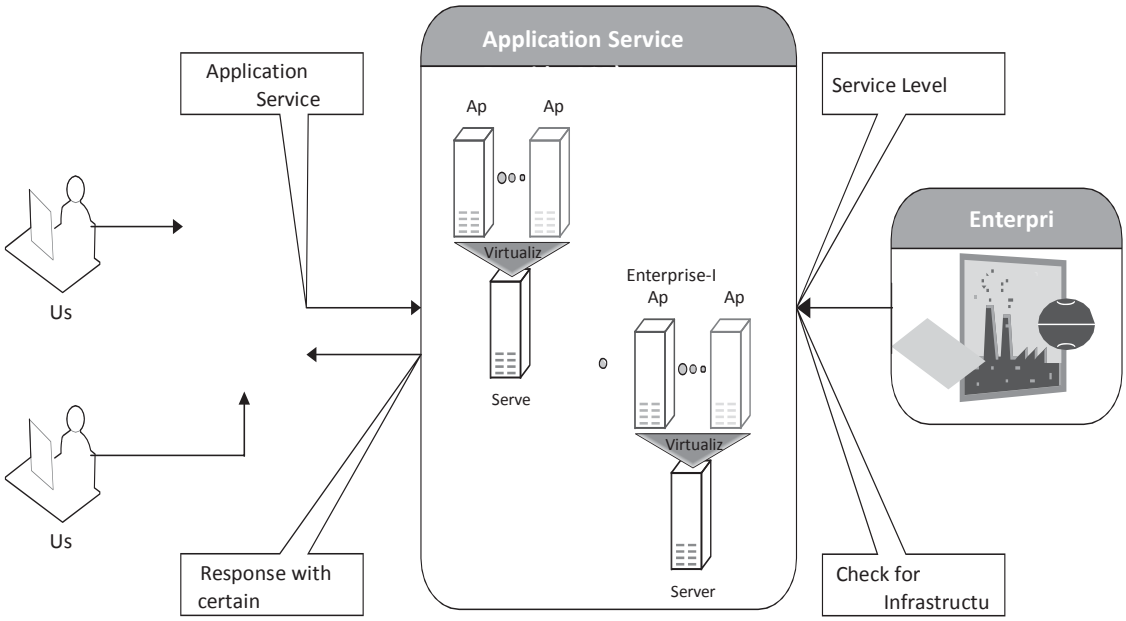


FIGURE 4.1. General taxonomy of load-balancing algorithms.

The load balancing algorithm executes on a physical machine that interfaces with the clients. This physical machine, also called the front-end node, receives the incoming requests and distributes these requests to different physical machines for further execution. This set of physical machines is responsible for serving the incoming requests and are known as the back-end nodes.

FIGURE 4.2. Shared hosting of applications on virtualized servers within ASP's data centers.



- Typically, the algorithm executing on the front-end node is agnostic to the nature of the request. This means that the front-end node is neither aware of the type of client from which the request originates nor aware of the category (e.g., browsing, selling, payment, etc.) to which the request belongs to. This category of load balancing algorithms is known as class-agnostic.

- There is a second category of load balancing algorithms that is known as class-aware. With class-aware load balancing and requests distribution, the front-end node must additionally inspect the type of client making the request and/or the type of service requested before deciding which back-end node should service the request. Inspecting a request to find out the class or category of a request is difficult because the client must first establish a connection with a node (front-end node) that is not responsible for servicing the request.

Admission Control

- Admission control algorithms play an important role in deciding the set of requests that should be admitted into the application server when the server experiences "very" heavy loads During overload situations, since the response time for all the requests would invariably degrade if all the arriving requests are admitted into the server, it would be preferable to be selective in identifying a subset of requests that should be admitted into the system so that the overall pay-off is high. The objective of admission control mechanisms, therefore, is to police the incoming requests and identify a subset of incoming requests that can be admitted into the system when the system faces overload situations.

TYPES OF SLA

- Service-level agreement provides a framework within which both seller and buyer of a service can pursue a profitable service business relationship. It outlines the broad understanding between the service provider and the service

consumer for conducting business and forms the basis for maintaining a mutually beneficial relationship. From a legal perspective, the necessary terms and conditions that bind the service provider to provide services continually to the service consumer are formally defined in SLA.

- SLA can be modeled using web service-level agreement (WSLA) language specification .Although WSLA is intended for web-service-based applica- tions, it is equally applicable for hosting of applications. Service-level parameter, metric, function, measurement directive, service-level objective, and penalty are some of the important components of WSLA and are described in Table 4.2.1.

TABLE 4.1. Key Components of a Service-Level Agreement

Service-Level Parameter	Describes an observable property of a service whose value is measurable.
Metrics	These are definitions of values of service roperties that are measured from a service-providing system or computed from other metrics and constants. Metrics are the key instrument to describe exactly what SLA parameters mean by specifying how to measure or compute the parameter values.
Function	A function specifies how to compute a metric's value from the values of other metrics and constants. Functions are central to describing exactly how SLA parameters are computed from resource metrics.
Measurement directives	These specify how to measure a metric

There are two types of SLAs from the perspective of application hosting. These are described in detail here.

Infrastructure SLA.

- The infrastructure provider manages and offers guaran- tees on availability of the infrastructure, namely, server machine, power, network connectivity, and so on. Enterprises manage themselves, their applica- tions that are deployed on these server machines. The machines are leased to the customers and are isolated from machines of other customers. In such dedicated hosting environments, a practical example of service-level guarantees offered by infrastructure providers.

Application SLA.

- In the application co-location hosting model, the server capacity is available to the applications based solely on their resource demands. Hence, the service providers are flexible in allocating and de-allocating computing resources among the co-located applications. Therefore, the service

LIFE CYCLE OF SLA

Each SLA goes through a sequence of steps starting from identification of terms and conditions, activation and monitoring of the stated terms and conditions, and eventual termination of contract once the hosting relationship ceases to exist. Such a sequence of steps is called SLA life cycle and consists of the following five phases:

1. Contract definition
2. Publishing and discovery
3. Negotiation
4. Operationalization
5. De-commissioning

Here, we explain in detail each of these phases of SLA life cycle.

Contract Definition.

- Generally, service providers define a set of service offerings and corresponding SLAs using standard templates. These service offerings form a catalog. Individual SLAs for enterprises can be derived by customizing these base SLA templates.

Publication and Discovery.

- Service provider advertises these base service offerings through standard publication media, and the customers should be able to locate the service provider by searching the catalog. The customers can search different competitive offerings and shortlist a few that fulfill their requirements for further negotiation.

Negotiation.

- Once the customer has discovered a service provider who can meet their application hosting need, the SLA terms and conditions needs to be mutually agreed upon before signing the agreement for hosting the application. For a standard packaged application which is offered as service, this phase could be automated. For customized applications that are hosted on cloud platforms, this phase is manual. The service provider needs to analyze the application's behavior with respect to scalability and performance before agreeing on the specification of SLA. At the end of this phase, the SLA is mutually agreed by both customer and provider and is eventually signed off. SLA negotiation can utilize the WS-negotiation specification .

Operationalization.

- SLA operation consists of SLA monitoring, SLA ac-counting, and SLA enforcement. SLA monitoring involves measuring parameter values and calculating the metrics defined as a part of SLA and determining the deviations. On

identifying the deviations, the concerned parties are notified. SLA accounting involves capturing and archiving the SLA adherence for compliance. As part of accounting, the application's actual performance and the performance guaranteed as a part of SLA is reported.

De-commissioning.

- SLA decommissioning involves termination of all activities performed under a particular SLA when the hosting relationship between the service provider and the service consumer has ended. SLA specifies the terms and conditions of contract termination and specifies situations under which the relationship between a service provider and a service consumer can be considered to be legally ended.

SLA MANAGEMENT IN CLOUD

SLA management of applications hosted on cloud platforms involves five phases.

1. Feasibility
2. On-boarding
3. Pre-production
4. Production
5. Termination

These activities are explained in detail in the following subsections.

Feasibility Analysis

MSP conducts the feasibility study of hosting an application on their cloud platforms. This study involves three kinds of feasibility: (1) technical feasibility,
(2) Infra structure feasibility, and (3) financial feasibility. The technical feasi-bility of an application implies determining the following:

1. Ability of an application to scale out.
2. Compatibility of the application with the cloud platform

being used within the MSP's data center.

3. The need and availability of a specific hardware and software required for hosting and running of the application.
4. Preliminary information about the application performance and whether be met by the MSP.

PERFORMANCE PREDICTION FOR HPC ON CLOUDS

GRID AND CLOUD

- *"Grid vs Cloud"* is the title of an incredible number of recent Web blogs and articles in on-line forums and magazines, where many HPC users express their own opinion on the relationship between the two paradigms .

- Cloud is simply presented, by its supporters, as an evolution of the grid. Some consider grids and clouds as alternative options to do the same thing in a different way. However, there are very few clouds on which one can build, test, or run compute-intensive applications. In fact it still necessary to deal with some open issues. One is when, in term of performance, a cloud is better than a grid to run a specific application. Another problem to be addressed concerns the effort to port a grid application to a cloud.

Grid and Cloud Integration

- To understand why grids and clouds should be integrated, we have to start by considering what the users want and what these two technologies can provide. Then we can try to understand how cloud and grid can complement each other and why their integration is the goal of intensive research.

- The integration of cloud and grid, or at least their integrated utilization, has been proposed since there is a trade-off between application turnaround and system utilization, and sometimes it is useful to choose the right compromise between them.

Some issues to be investigated have been pointed out:

- Integration of virtualization into existing e-infrastructures
- Deployment of grid services on top of virtual infrastructures
- Integration of cloud-base services in e-infrastructures
- Promotion of open-source components to build clouds
- Grid technology for cloud federation

In light of the above, the integration of the two environments is a debated issue . At the state of the art, two main approaches have been proposed:

- *Grid on Cloud*. A cloud IaaS (Infrastructure as a Service) approach is

adopted to build up and to manage a flexible grid system .Doing so, the grid middleware runs on a virtual machine. Hence the main drawback of this approach is performance. Virtualization inevitably entails performance losses as compared to the direct use of physical resources.

- *Cloud on Grid*: The stable grid infrastructure is exploited to build up a cloud environment. This solution is usually preferred because the cloud approach mitigates the inherent complexity of the grid. In this case, a set of grid services is offered to manage (create, migrate, etc.) virtual machines. The use of *Globus workspaces* [16], along with a set of grid services for the Globus Toolkit 4, is the prominent solution, as in the Nimbus project .

The integration could simplify the task of the HPC user to select, to configure, and to manage resources according to the application requirements. It adds flexibility to exploit available resources, but both of the above-presented approaches have serious problems for overall system management, due to the complexity of the resulting architectures. Performance prediction, application tuning, and benchmarking are some of the relevant activities that become critical and that cannot be performed in the absence of performance evaluation of clouds.

HPC IN THE CLOUD: PERFORMANCE-RELATED ISSUES

This section will discuss the issues linked to the adoption of the cloud paradigm in the HPC context. In particular, we will focus on three different issues:

1. The difference between typical HPC paradigms and those of current cloud environments, especially in terms of performance evaluation.
2. A comparison of the two approaches in order to point out their advantages and drawbacks, as far as performance is concerned.
3. New performance evaluation techniques and tools to support HPC in cloud systems.

TABLE 4.1. Example of Cost Criteria

Cloud Provider	Index	Description
Amazon	$/hour	Cost (in $) per hour of activity of the virtual machines.
Amazon	$/GB	Cost (in $) per Gigabyte transferred outside the cloud zone (transfers inside the same zone have no price)
Go Grid	$*RAM/hour	Cost (in $) by RAM memory allocated per hour

APPLICATIONS

BEST PRACTICES IN ARCHITECTING CLOUD APPLICATIONS IN THE AWS CLOUD

- **Business Benefits of Cloud Computing**

There are some clear business benefits to building applications in the cloud. A few of these are listed here:

Almost Zero Upfront Infrastructure Investment. If you have to build a large- scale system, it may cost a fortune to invest in real estate, physical security, hardware (racks, servers, routers, backup power supplies), hardware management (power management, cooling), and operations personnel. Because of the high upfront costs, the project would typically require several rounds of management approvals before the project could even get started. Now, with utility-style cloud computing, there is no fixed cost or startup cost.

Just-in-Time Infrastructure. In the past, if your application became popular and your systems or your infrastructure did not scale, you became a victim of your own success. Conversely, if you invested heavily and did not get popular, you became a victim of your failure. By deploying applications in-the-cloud with just-in-time self-provisioning, you do not have to worry about pre-procuring capacity for large-scale systems. This increases agility, lowers risk, and lowers operational cost because you scale only as you grow and only pay for what you use.

More Efficient Resource Utilization. System administrators usually worry about procuring hardware (when they run out of capacity) and higher infrastructure utilization (when they have excess and idle capacity). With the cloud, they can manage resources more effectively

and efficiently by having the applications request and relinquish resources on-demand.

Usage-Based Costing. With utility-style pricing, you are billed only for the infrastructure that has been used. You are not paying for allocated infrastructure but instead for unused infrastructure. This adds a new dimension to cost savings. You can see immediate cost savings (sometimes as early as your next month's bill) when you deploy an optimization patch to update your cloud application. For example, if a caching layer can reduce your data requests by 70%, the savings begin to accrue immediately and you see the reward right in the next bill. Moreover, if you are building platforms on the top of the cloud, you can pass on the same flexible, variable usage-based cost structure to your own customers.

Reduced Time to Market. Parallelization is one of the great ways to speed up processing. If one compute-intensive or data-intensive job that can be run in parallel takes 500 hours to process on one machine, with cloud architectures , it would be possible to spawn and launch 500 instances and process the same job in 1 hour. Having available an elastic infrastructure provides the application with the ability to exploit paralle- lization in a cost-effective manner reducing time to market.

Technical Benefits of Cloud Computing

Some of the technical benefits of cloud computing includes:

Automation—"Scriptable Infrastructure": You can create repeatable build and deployment systems by leveraging programmable (API-driven) infrastructure.

Auto-scaling: You can scale your applications up and down to match your unexpected demand without any human intervention. Auto-scaling encourages automation and drives more efficiency.

Proactive Scaling: Scale your application up and down to meet your anticipated demand with proper planning understanding of your traffic patterns so that you keep your costs low while scaling.

More Efficient Development Life Cycle: Production systems may be easily cloned for use as development and test environments. Staging environ- ments may be easily promoted to production.

Improved Testability: Never run out of hardware for testing. Inject and automate testing at every stage during the development process. You can spawn up an "instant test lab" with preconfigured environments

only for the duration of testing phase.

Disaster Recovery and Business Continuity: The cloud provides a lower cost option for maintaining a fleet of DR servers and data storage. With the cloud, you can take advantage of geo-distribution and replicate the environment in other location within minutes.

"Overflow" the Traffic to the Cloud: With a few clicks and effective load balancing tactics, you can create a complete overflow-proof application by routing excess traffic to the cloud.

UNIT-5

GOVERNANCE AND CASE STUDIES

ORGANIZATIONAL READINESS AND CHANGE MANAGEMENT IN THE CLOUD AGE

INTRODUCTION

- Studies for Organization for Economic Co-operation and Development (OECD) economies in 2002 demonstrated that there is a strong correlation between changes in organization and workplace practices and investment in information technologies .

- This finding is also further confirmed in Canadian government studies, which indicate that the frequency and intensity of organizational changes is positively correlated with the amount and extent of information technologies investment. It means that the incidence of organizational change is much higher in the firms that invest in information technologies (IT) than is the case in the firms that do not invest in IT, or those that invest less than the competitors in the respective industry .

- In another study, Bresnahan, Brynjolfsson, and Hitt found that there is positive correlation between information technology change (investment), organizational change (e.g., process re-engineering, organizational structure), cultural change (e.g., employee empowerment), and the value of the firm as a measure of the stock market share price. This is mostly due to the productivity and profitability gain through technology investment and organizational changes. The research and analysis firm Gartner has released the Hype Cycle report for 2009, which evaluates the maturity of 1650 technologies and trends in 79 technologies. The report, which covers new areas this year, defines cloud computing as the latest growing trend in the IT industry, stating it as "super- hyped." The other new areas include data center power, cooling technologies, and mobile device technologies.

- In order to effectively enable and support enterprise business goals and strategies, information technology (IT) must adapt and continually change. IT must adopt emerging technologies to facilitate business to leverage the new technologies to create new opportunities, or to gain productivity and reduce cost. Sometimes emerging technology (e.g., cloud computing: IaaS, PaaS, SaaS) is quite disruptive to the existing business process, including core IT services— for example, IT service strategy, service design, service transition, service operation, and continual service improvement—and requires fundamental re-thinking of how to minimize the negative impact to the business, particularly the potential impact on morale and productivity of the organization.

The Context:

- The adaptation of cloud computing has forced many companies to recognize that clarity of ownership of the data is of paramount importance. The protection of intellectual property (IP) and other copyright issues is of big concern and needs to be addressed carefully.

- This will help the student to assess the organization readiness to adopt the new/emerging technology. What is the best way to implement and manage change? While this chapter attempts to explain why change is important and why change is complex, it also raises the question of (a) managing emerging technologies and (b) the framework and approaches to assess the readiness of the organization to adopt. Managing emerging technologies is always a complex issue, and managers must balance the desire to create competiveness through innovation with the need to manage the complex challenges presented by these emerging technologies. Managers need to feel comfortable dealing with the paradox of increasing complexity and uncertainty, and they need balance it with desirable level of commitment and built-in flexibility.

The Take Away:

- Transition the organization to a desirable level of change management maturity level by enhancing the following key domain of knowledge and competencies:

Domain 1. Managing the Environment: Understand the organization (people, process, and culture).

Domain 2. Recognizing and Analyzing the Trends (Business and Technology): Observe the key driver for changes.

Domain 3. Leading for Results: Assess organizational readiness and architect solution that delivers definite business values.

Basic Concept Of Organizational Readiness:

- Change can be challenging; it brings out the fear of having to deal with uncertainties. This is the FUD syndrome: Fear, Uncertainty, and Doubt.

- Employees understand and get used to their roles and responsibility and are able to leverage their strength. They are familiar with management's expectation of them and don't always see a compelling reason to change. Whenever there are major changes being introduced to the organization, changes that require redesign or re-engineering the

business process, change is usually required to the organizational structure and to specific jobs. Corporate leader- ship must articulate the reasons that change is critical and must help the workers to visualize and buy into the new vision. Corporate leadership also needs to communicate and cultivate the new value and beliefs of the organization that align and support the corporate goals and objectives. The human resources department also needs to communicate the new reward and compensation system that corresponds to the new job description and identify new training and skills requirements that support the new corporate goal and objectives.

It is a common, observable human behavior that people tend to become comfortable in an unchanging and stable environment, and will become uncomfortable and excited when any change occurs, regardless the level and intensity of the change.

- Protect Existing Investment: By building a private cloud to leverage existing infrastructure.
- Manage Security Risk: Placing private cloud computing inside the company reduces some of the fear (e.g., data integrity and privacy issues) usually associated with public cloud.

A Case Study: Waiting in Line for a Special Concert Ticket

- It is a Saturday morning in the winter, the temperature is 212°C outside, and you have been waiting in line outside the arena since 5:00 AM this morning for concert tickets to see a performance by Super tramp. You have been planning for this with your family for the past 10 months since they announced that Super tramp is coming into town next December. When it is your turn at the counter to order tickets, the sales clerk announces that the concert is all sold out. What is your reaction? What should you do you need to change the plan? Your reaction would most likely be something like this:

- Denial. You are in total disbelief, and the first thing you do is to reject the fact that the concert has been sold out.
- Anger. You probably want to blame the weather; you could have come here 10 minutes earlier.
- Bargaining. You try to convince the clerk to check again for any available seats.
- Depression. You are very disappointed and do not know what to do next.
- Acceptance. Finally accepting the inevitable fate, you go to plan B if you have one.

- The five-stage process illustrated above was originally proposed by Dr. Elizabeth Ku¨ bler-Ross to deal with catastrophic news. There are times

in which people receive news that can seem catastrophic; for example; company merger, right- sizing, and so on. In her book *On Death and Dying*, Elizabeth Ku¨bler-Ross describes what is known as the "Ku¨bler-Ross model" or the "Five Stages of Grief"; this model relates to change management, specifically the emotions felt by those affected by change. The first stage of major change is often the announcement; there are situations when an understanding of the five-stage process will help you move more quickly to deal with the issue.

Drivers for Changes: A Framework To Comprehend The Competitive Environment

The Framework: The five driving factors for change encapsulated by the framework are:

- Economic (global and local, external and internal)
- Legal, political, and regulatory compliance
- Environmental (industry structure and trends)
- Technology developments and innovation
- Socio cultural (markets and customers)

The five driving factors for change is an approach to investigate, analyze, and forecast the emerging trends of a plausible future, by studying and understanding the five categories of drivers for change. The results will help the business to make better decisions, and it will also help shape the short- and long-term strategies of that business. It is this process that helps reveal the important factors for the organization's desirable future state, and it helps the organization to comprehend which driving forces will change the competitive landscape in the industry the business is in, identify critical uncertainties, and recognize what part of the future is predetermined such that it will happen regardless how the future will play out. This approach also helps seek out those facts and perceptions that challenge one's underlying assumptions, and thus it helps the company make a better decision.

- Every organization's decisions are influenced by particular key factors, some of them are within the organization's control, such as (a) internal financial weakness and strength and (b) technology development and innovation, and therefore the organization has more control. The others, such as legal compliance issues, competitor capabilities, and strategies, are all external factors over which the organization has little or no control. There are also many other less obvious external factors that will impact the organization; identifying and assessing these fundamental factors and formulating a course of action proactively is paramount to any business success.

  A driving force or factor is a conceptual tool; it guides us to think deeply about the underlying issues that impact our well-being and

success. In a

business setting, it helps us to visualize and familiarize ourselves with future possibilities (opportunities and threats).

Economic (Global and Local, External and Internal)

- Economic factors are usually dealing with the state of economy, both local and global in scale. To be successful, companies have to live with the paradox of having new market and business opportunities globally, and yet no one can be isolated from the 2008 global financial crisis, because we are all interdependent. Managers are often asked to do more with less, and this phenomenon is especially true during economic downturn. Managers and groups are expected to deal with the unpleasant facts of shrinking market share, declining profit margins, unsatisfactory earnings, new and increasing competition, and decreasing competitiveness.

Following are sample questions that could help to provoke further discussion:

- What is the current economic situation?
- What will the economy looks like in 1 year, 2 years, 3 years, 5 years, and so on?
- What are some of the factors that will influence the future economic outlook?
- Is capital easy to access?
- How does this technology transcend the existing business model?
- Buy vs. build? Which is the right way?
- What is the total cost of ownership (TCO)?

TECHNOLOGY DEVELOPMENTS AND INNOVATION

Scientific discoveries are seen to be key drivers of economic growth; leading economists have identified technological innovations as the single most important contributing factor in sustained economic growth. There are many fronts of new and emerging technologies that could potentially transform our world. For example, new research and development in important fields such as bioscience, nanotechnology, and information technology could potentially change our lives.

The following are sample questions that could help to provoke further discussion:

- When will the IT industry standards be finalized? By who? Institute of Electrical and Electronics Engineers (IEEE)?
- Who is involved in the standardization process?
- Who is the leader in cloud computing technology?

- What about virtualization of application operating system (platform) pair (i.e., write once, run anywhere)?
- How does this emerging technology (cloud computing) open up new areas for innovation?
- How can an application be built once so it can configure dynamically in real time to operate most effectively, based on the situational constraint (e.g., out in the cloud somewhere, you might have bandwidth constraint to transfer needed data)?
- What is the guarantee from X Service Providers (XSP) that the existing applications will still be compatible with the future infrastructure (IaaS)? Will the data still be executed correctly?

SOCIO CULTURAL (MARKETS AND CUSTOMERS)

- Societal factors usually deal with the intimate understanding of the human side of changes and with the quality of life in general. A case in point: The companies that make up the U.S. defense industry have seen more than 50% of their market disappear. When the Berlin Wall tumbled, the U.S. government began chopping major portions out of the defense budget. Few would disagree that the post Cold War United States could safely shrink its defense industry. Survival of the industry, and therefore of the companies, demands that companies combine with former competitors and transform into new species.

CREATING A WINNING ENVIRONMENT

- At the cultural level of an organization, change too often requires a lot of planning and resource. This usually stems from one common theme: Senior management and employees have different perspectives and interpretations of what change means, what change is necessary, and even if changes are necessary at all. In order to overcome this, executives must articulate a new vision and must communicate aggressively and extensively to make sure that every employee understands.

COMMON CHANGE MANAGEMENT MODELS

There are many different change management approaches and models, and we will discuss two of the more common models and one proposed working model (CROPS) here; the Lewin's Change Management Model, the Deming Cycle(Plan, Do, Study, Act) and the proposed CROPS Change Management Framework.

Lewin's Change Management Model

- Kurt Lewin, a psychologist by training, created this change model in the 1950s. Lewin observed that there are three stages of change,

which are: *Unfreeze, Transition*, and *Refreeze*. It is recognized that people tend to become compla- cent or comfortable in this "freeze" or "unchanging/stable" environment, and they wish to remain in this "safe/comfort" zone. Any disturbance/disruption to this unchanging state will cause pain and become uncomfortable.

- In order to encourage change, it's necessary to unfreeze the environment by motivating people to accept the change. The motivational value has to be greater than the pain in order to entice people to accept the change. Maintain- ing a high level of motivation is important in all three phases of the change management life cycle, even during the transition period. As Lewin put it, "Motivation for change must be generated before change can occur. One must be helped to reexamine many cherished assumptions about oneself and one's relations to others." This is the unfreezing stage from which change begins.

- Since these "activities" take time to be completed, the process and organizational structure may also need to change, specific jobs may also change. The most resistance to change may be experienced during this transition period. This is when leadership is critical for the change process to succeed, and motivational factors are paramount to project success.The last phase is Refreeze; this is the stage when the organization once again becomes unchanging/frozen until the next time a change is initiated.
- The Deming cycle is also known as the PDCA cycle; it is a continuous improvement (CI) model comprised of four sequential sub processes; Plan,Do, Check, and Act. This framework of process and system improvement was originally conceived by Walter She whart in the 1930s and was later adopted by Edward Deming. The PDCA cycle is usually implemented as an evergreen process, which means that the end of one complete pass (cycle) flows into the beginning of the next pass and thus supports the concept of continuous quality improvement.

- Edward Deming proposed in the 1950s that business processes and systems should be monitored, measured, and analyzed continuously to identify variations and substandard products and services, so that corrective actions can be taken to improve on the quality of the products or services delivered to the customers.

  - PLAN: *R*ecognize an opportunity and plan a change.
  - DO: *E*xecute the plan in a small scale to prove the concept.
  - CHECK: *E*valuate the performance of the change and report the results to sponsor.
  - ACT: *D*ecide on accepting the change and standardizing it as part of the process.

Incorporate what has been learned from the previous steps to plan new improvements, and begin a new cycle.

- For many organizations, change management focuses on the project management aspects of change. There are a good number of vendors offering products that are intended to help organizations manage projects and project changes, including the Project Portfolio Management Systems (PPMS). PPMS groups projects so they can be managed as a portfolio, much as an investor would manage his/her stock investment portfolio to reduce risks.

- In the IT world, a project portfolio management system gives management timely critical information about projects so they can make better decisions; re-deploy resources due to changing priorities, and keep close tabs on progress.

- However, as the modern economy moves from product and manufacturing centric to a more information and knowledge base focus, the change manage- ment process needs to reflect that people are truly the most valuable asset of the organization. Usually, an organization experiences strong resistance to change. Employees are afraid of the uncertainty, they feel comfortable with the stable state and do not want to change, and are afraid to lose their power if things change. To them, there is no compelling reason to change, unless the company can articulate a compelling reason and communicate it effectively to convince them and influentially engage them to change.

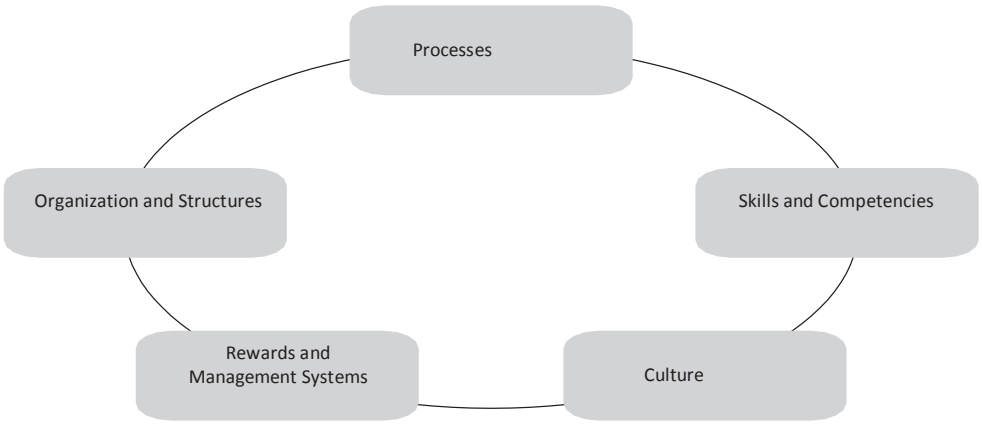
- The best approaches to address resistance are through increased and sustained communications and education. The champion of change, usually the leader— for example, the Chief Information Officer (CIO) of the organization—should communicate the *Why* aggressively and provide a *Vision* of *Where* he wants to go today. There are many writings and models on organization development (i.e., how). A summary of this working model follows: Culture, Rewards, Organization and Structures, Process, Skills and Competencies (CROPS) framework.

Culture*:* Corporate culture is a reflection of organizational (management and employees) values and belief. Edgar Schein, one of the most prominent theorists of organizational culture, gave the following very general definition

The culture of a group can now be defined as: A pattern of shared basic assumptions that the group learned as it solved its problems of external adaptation and internal integration, that has worked well enough to be considered valid and, therefore, to be taught to new members as the correct way to perceive, think, and feel in relation to those problems.

Elements of organizational culture may include:

- Stated values and belief
- Expectations for member behavior
- Customs and rituals
- Stories and myths about the history of the organization



- Norms—the feelings evoked by the way members interact with each other, with outsiders, and with their environment
- Metaphors and symbols—found embodied in other cultural elements

- Rewards and Management System: This management system focuses on how employees are trained to ensure that they have the right skills and tools to do the job right. It identifies how to measure employee job performance and how the company compensates them based on their performance. Reward is the most important ingredient that shapes employees' value and beliefs.
- Organization and Structures: How the organization is structured is largely influenced by what the jobs are and how the jobs are performed. The design of the business processes govern what the jobs are, and when and where they get done. Business processes need to align with organizational vision, mission, and strategies in order to create customer and shareholder values. Therefore, all the components of the CROPS framework are interrelated.
- Process: Thomas Davenport defined a business process or business method as a collection of related, structured activities or tasks that produce a specific service or product (serve a particular goal) for a particular customer or customers.
- hammer and Champy's definition can be considered as a subset of Davenport's. They define a process as "a collection of activities that takes one or more kinds of input and creates an output that is of value to the customer."

CHANGE MANAGEMENT MATURITY MODEL (CMMM):

- A Change Management Maturity Model (CMMM) helps organizations to (a) analyze, understand, and visualize the strength and weakness of the firm's change management process and (b) identify opportunities for improvement and building competitiveness. The model should be simple enough to use and flexible to adapt to different situations. The working model in Table 22.1 is based on CMM (Capability Maturity Model), originally developed by American Software Engineering Institute (SEI) in cooperation with Mitre Corpora- tion. CMM is a model of process maturity

  How does CMMM help organizations to adopt new technology, including cloud computing, successfully? The business value of CMMM can be expressed in terms of improvements in business efficiency and effectiveness. All organizational investments are business investments, including IT investments. The resulting benefits should be measured in terms of business returns. Therefore, CMMM value can be articulated as the ratio of business performance to CMMM investment.

DATA SECURITY IN CLOUD:

Introduction To The Idea Of Data Security

- Taking information and making it secure, so that only yourself or certain others can see it, is obviously not a new concept. However, it is one that we have struggled with in both the real world and the digital world. In the real world, even information under lock and key, is subject to theft and is certainly open to accidental or malicious misuse. In the digital world, this analogy of lock-and-key protection of information has persisted, most often in the form of container-based encryption. But even our digital attempt at protecting information has proved less than robust, because of the limitations inherent in protecting a container rather than in the content of that container. This limitation has become more evident as we move into the era of cloud computing: Information in a cloud environment has much more dynamism and fluidity than information that is static on a desktop or in a network folder, so we now need to start to think of a new way to protect information.

- Before we embark on how to move our data protection methodologies into the era of the cloud, perhaps we should stop, think, and consider the true applicability of information security and its value and scope. Perhaps we should be viewing the application of data security as less of a walled and impassable fortress and more of a sliding series of

options that are more appropriately termed “risk mitigation.”

- The reason that I broach this subject so early on is that I want the reader to start to view data security as a lexicon of choices, as opposed to an on/off technology. In a typical organization, the need for data security has a very wide scope, varying from information that is set as public domain, through to information that needs some protection (perhaps access control), through data that are highly sensitive, which, if leaked, could cause catastrophic damage, but nevertheless need to be accessed and used by selected users.

- One other aspect of data security that I want to draw into this debate is the human variable within the equation. Computer technology is the most modern form of the toolkit that we have developed since human prehistory to help us improve our lifestyle. From a human need perspective, arguably, computing is no better or worse than a simple stone tool, and similarly, it must be built to fit the hand of its user. Technology built without considering the human impact is bound to fail. This is particularly true for security technology, which is renowned for failing at the point of human error.

- If we can start off our view of data security as more of a risk mitigation exercise and build systems that will work with humans (i.e., human-centric), then perhaps the software we design for securing data in the cloud will be successful.

THE CURRENT STATE OF DATA SECURITY IN THE CLOUD

- At the time of writing, cloud computing is at a tipping point: It has many arguing for its use because of the improved interoperability and cost savings it offers. On the other side of the argument are those who are saying that cloud computing cannot be used in any type of pervasive manner until we resolve the security issues inherent when we allow a third party to control our information.

- These security issues began life by focusing on the securing of access to the datacenters that cloud-based information resides in. However, it is quickly becoming apparent in the industry that this does not cover the vast majority of instances of data that are outside of the confines of the data center, bringing us full circle to the problems of having a container-based view of securing data. This is not to say that data-center security is obsolete. Security, after all, must be viewed as a series of concentric circles emanating from a resource and touching the various places that the data go to and reside.

- However, the very nature of cloud computing dictates that data are fluid objects, accessible from a multitude of nodes and geographic

locations and, as such, must have a data security methodology that takes this into account while ensuring that this fluidity is not compromised. This apparent dichotomy data security with open movement of data—is not as juxtaposed as it first seems. Going back to my previous statement that security is better described as "risk mitigation," we can then begin to look at securing data as a continuum of choice in terms of levels of accessibility and content restrictions: This continuum allows us to choose to apply the right level of protection, ensuring that the flexibility bestowed by cloud computing onto the whole area of data communication is retained.

- As I write, the IT industry is beginning to wake up to the idea of content- centric or information-centric protection, being an inherent part of a data object. This new view of data security has not developed out of cloud computing, but instead is a development out of the idea of the "deperimerization" of the enterprise. This idea was put forward by a group of Chief Information Officers (CIOs) who formed an organization called the Jericho Forum [1]. The Jericho Forum was founded in 2004 because of the increasing need for data exchange between companies and external parties for example: employees using remote computers; partner companies; customers; and so on.

- The old way of securing information behind an organization's perimeter wall prevented this type of data exchange in a secure manner. However, the ideas forwarded by the Jericho Forum are also applicable to cloud computing. The idea of creating, essentially, de-centralized perimeters, where the perimeters are created by the data object itself, allows the security to move with the data, as opposed to retaining the data within a secured and static wall. This simple but revolutionary change in mindset of how to secure data is the ground stone of securing information within a cloud and will be the basis of this discussion on securing data in the cloud.

HOMO SAPIENS AND DIGITAL INFORMATION

- Cloud computing offers individuals and organizations a much more fluid and open way of communicating information. This is a very positive move forward in communication technology, because it provides a more accurate mimic of the natural way that information is communicated between individuals and groups of human beings.

- Human discourse, including the written word, is, by nature, an open transaction: I have this snippet of information and I will tell you, verbally or in written form, what that information is. If the information is sensitive, it may be whispered, or, if written on paper, passed only to those allowed to read it. The result is that human-to-human

- Cloud computing is a platform for creating the digital equivalent of this fluid, human-to-human information flow, which is something that internal computing networks have never quite achieved. In this respect, cloud computing should be seen as a revolutionary move forward in the use of technology to enhance human communications.

- Although outside of the remit of this chapter, it is worthwhile for any person looking into developing systems for digital communications to attempt to understand the underlying social evolutionary and anthropological reasons behind the way that human beings communicate This can give some insight into digital versions of communication models, because most fit with the natural way that humans communicate information. Security system design, in particular, can benefit from this underlying knowledge, because this type of system is built both to thwart deceptive attempts to intercept communication and to enhance and enable safe and trusted communications: Bear in mind that both deception and trust are intrinsic evolutionary traits, which human beings have developed to help them to successfully communicate.

Cloud Computing and Data Security Risk

- The cloud computing model opens up old and new data security risks. By its very definition, Cloud computing is a development that is meant to allow more open accessibility and easier and improved data sharing.

- Data are uploaded into a cloud and stored in a data center, for access by users from that data center; or in a more fully cloud-based model, the data themselves are created in the cloud and stored and accessed from the cloud (again via a data center). The most obvious risk in this scenario is that associated with the storage of that data.

- A user uploading or creating cloud-based data include those data that are stored and maintained by a third-party cloud provider such as Google, Amazon, Microsoft, and so on. This action has several risks associated with it: Firstly, it is necessary to protect the data during upload into the data center to ensure that the data do not get hijacked on the way into the database.

- Secondly, it is necessary to the stores the data in the data center to ensure that they are encrypted at all times. Thirdly, and perhaps less obvious, the access to those data need to be controlled; this control should also be applied to the hosting company, including the administrators of the data center.

- In addition, an area often forgotten in the application of security to a data resource is the protection of that resource during its use—that is, during a collaboration step as part of a document workflow process.

- Other issues that complicate the area of hosted data include ensuring that the various data security acts and rules are adhered to; this becomes particularly complicated when you consider the cross border implications of cloud computing and the hosting of data in a country other than that originating the data.

CLOUD COMPUTING AND IDENTITY

- Digital identity holds the key to flexible data security within a cloud environment. This is a bold statement, but nonetheless appears to be the method of choice by a number of industry leaders.

- However, as well as being a perceived panacea for the ills of data security, it is also one of the most difficult technological methods to get right. Identity, of all the components of information technology, is perhaps the most closest to the heart of the individual.

- After all, our identity is our most personal possession and a digital identity represents who we are and how we interact with others on-line. The current state of the art in digital identity, in particular with reference to cloud identities, is a work in progress, which by the time you are reading this should hopefully be entering more maturity.

- However, going back to my opening statement, digital identity can be used to form the basis of data security, not only in the cloud but also at the local network level too. To expand on this somewhat, we need to look at the link between access, identity, and risk. These three variables can become inherently connected when applied to the security of data, because access and risk are directly proportional:

- As access increases, so then risk to the security ofthe data increases. Access controlled by identifying the actor attempting the access is the most logical manner of performing this operation. Ultimately, digital identity holds the key to securing data, if that digital identity can be programmatically linked to security policies controlling the post-access usage of data.

- The developments seen in the area of a cloud-based digital identity layer have been focused on creating a "user-centric" identity mechanism. User- centric identity, as opposed to enterprise-centric identity, is a laudable design goal for something that is ultimately owned by the user. However, the Internet tenet of "I am who I say I am" cannot support the security requirements of a data protection methodology based on digital identity, therefore digital identity, in the

context of a security system backbone, must be a verified identity by some trusted third party: It is worth noting that even if your identity is verified by a trusted host, it can still be under an individual's management and control.

- With this proposed use of identity, on the type of scale and openness as expected in a cloud computing context, we must also consider the privacy implications of that individual's identity. A digital identity can carry with it many identifiers about an individual that make identity theft a problem, but identity should also be kept private for the simple reason of respect. However, privacy is a very personal choice and, as such, the ability to remain private within a cloud, should be, at the very least, an option.

THE CLOUD, DIGITAL IDENTITY, AND DATA SECURITY

- When we look at protecting data, irrespective of whether that protection is achieved on a desktop, on a network drive, on a remote laptop, or in a cloud, we need to remember certain things about data and human beings. Data are most often information that needs to be used; it may be unfinished and require to be passed through several hands for collaboration for completion, or it could be a finished document needing to be sent onto many organizations and then passed through multiple users to inform.

- It may also be part of an elaborate workflow, across multiple document management systems, working on plat- forms that cross the desktop and cloud domain. Ultimately, that information may end up in storage in a data center on a third-party server within the cloud, but even then it is likely to be re-used from time to time.

- This means that the idea of "static" data is not entirely true and it is much better (certainly in terms of securing that data) to think of it as highly fluid, but intermittently static.

- What are the implications of this? If we think of data as being an "entity" that is not restricted by network barriers and that is opened by multiple users in a distributed manner, then we should start to envision that a successful protection model will be based on that protection policy being an intrinsic part of that entity. If the protection becomes inherent in the data object, in much the same way that perhaps a font type is inherent in a document (although in the case of security in a much more persistent manner), then it is much less important where that data resides. However, how this is achieved programmatically is a little trickier, particularly in terms of interoperability across hybrid cloud systems.

- One of the other aspects of data security we need to assess before embarking on creating a security model for data in the cloud is the *levels of need;* that is, how secure do you want that data to be? The levels of security of any data object should be thought of as concentric layers of increasingly pervasive security, which I have broken down here into their component parts to show the increasing granularity of this pervasiveness:

  Level 1: Transmission of the file using encryption protocols
  Level 2: Access control to the file itself, but without encryption of the content Level 3: Access control (including encryption of the *content* of a data object)
  Level 4: Access control (including encryption of the *content* of a data object) also including rights management options (for example, no copying content, no printing content, date restrictions, etc.)

- Other options that can be included in securing data could also include watermarking or red-acting of content, but these would come under level 4 above as additional options.

- You can see from the increasing granularity laid out here that security, especially within highly distributed environments like cloud computing, is not an on/off scenario. This way of thinking about security is crucial to the successful creation of cloud security models. Content level application of data security gives you the opportunity to ensure that all four levels can be met by a single architecture, instead of multiple models of operation which can cause interoperability issues and, as previously mentioned, can add additional elements of human error, leading to loss of security.

- The current state of cloud computing provides us with a number of cloud deployment models, namely, public (cloud infrastructure that is open for public use, for example, Google App engine is deployed in a public cloud), private (privately available clouds on a private network used by an individual company; for example,

- IBM provides private clouds to customers, particularly concerned by the security issues surrounding public cloud deployments), managed (clouds offered by a third-party hosting company who look after the implementation and operational aspects of cloud computing for an organization), and hybrid (a mix of both public and private cloud implementations).

- It is highly likely, especially in the early years of cloud computing, that organizations will use a mixture of several, if not all, of these different

models. With this in mind, to allow an organization to deal with securing data within any of these types of systems means that the issues of interoperability, cross-cloud support, minimization of human error, and persistence of security are crucial. The fluid movement of data through and between these clouds is an integral part of the cloud philosophy, and any data security added into this mix must not adversely encumber this movement.

- This requires that you look at that data as a separate entity with respect to the underlying system that it moves through and resides within. If you do not view the data as a free-moving object, you will build a data security model that is not built to suit the data, but instead is built for the specific system surrounding that data.

- In a cloud-type system, the end result is likely to be only suitable for static data (something that we have already described as not truly existing) which will not be able to transcend that original system without potentially having to be re-engineered to do so, or at the very least having additional features and functions tagged onto the original specification.

- This type of software engineering results in interoperability issues and an increased chance of bugs occurring, because of feature adjuncts being added as an afterthought, as opposed to being built into the original working architecture of the software.

- In addition, what can occur with security software development, which uses a non-extensible approach to software design, is that security holes end up being inadvertently built into the software, which may be very difficult to test for as the software feature bloat increases. With this in mind, the way forward in creating data security software models for a cloud computing environment must be done from scratch.

- We must leave the previous world of encrypted containers behind us and open up a new paradigm of fluidic protection mechanisms based on content-centric ideologies. Only through this approach will we hope to achieve transcendence of security across the varying types of cloud architectures.

EGAL ISSUES IN CLOUD COMPUTING

Definition of Cloud Computing

This chapter assumes that the reader is familiar with the manner in which cloud computing is defined as set forth by the National Institute of Standards and Technology, a federal agency of the United States Government.

- In brief, cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released. This cloud model is composed of five essential characteristics, three service models, and four deployment models.

OVERVIEW OF LEGAL ISSUES

- The legal issues that arise in cloud computing are wide ranging. Significant issues regarding privacy of data and data security exist, specifically as they relate to protecting personally identifiable information of individuals, but also as they relate to protection of sensitive and potentially confidential business information either directly accessible through or gleaned from the cloud systems (e.g., identification of a company's customer by evaluating traffic across the network).

- Additionally, there are multiple contracting models under which cloud services may be offered to customers (e.g., licensing, service agreements, on-line agreements, etc.).

- The appropriate model depends on the nature of the services as well as the potential sensitivity of the systems being implemented or data being released into the cloud. In this regard, the risk profile (i.e., which party bears the risk of harm in certain foreseeable and other not-so-foreseeable situations) of the agreement and the cloud provider's limits on its liability also require a careful look when reviewing contracting models.

- Additionally, complex jurisdictional issues may arise due to the potential for data to reside in disparate or multiple geographies. This geographical diversity is inherent in cloud service offerings. This means that both virtualization of and physical locations of servers storing and processing data may potentially impact what country's law might govern in the event of a data breach or intrusion into cloud systems.

DISTINGUISHING CLOUD COMPUTING FROM OUTSOURCING AND PROVISION OF APPLICATION SERVICES

Cloud computing is different from traditional outsourcing and the application service provider (ASP) model in the following ways:

- In general, outsourcers tend to take an entire business or IT process of a customer organization and completely run the business for the benefit of the customer. Though the outsourcer may provide services similar to those by multiple customers, each outsourcing arrangement is highly negotiated, and the contract is typically lengthy and complex.

- Depending on the nature of the outsourcing, the software belongs to the customer, and software sublicense rights were transferred to the outsourcer as part of the arrangement. The customer's systems are run on the customer's equip- ment, though it is usually at an offsite location managed by the out- sourcer.

- Pricing is typically negotiated for each outsourced relationship. The outsourcer's ability to scale to meet customer demand is a slow, and also negotiated, process. The location of the data and processing is known, predetermined, and agreed to contractually.

- In the ASP model, the service provided is a software service. The software application may have been used previously in-house by the customer, or it may be a new value-added offering. The ASP offering is a precursor to what is now called "software as a service." The transaction is negotiated, though typically it is not as complex and highly negotiated as a traditional outsourcing arrangement.

- The provider owns the software and hardware, and the software is accessed over the Internet. The software tends to reside in one physical location or a group of known locations with redundant and disaster recovery backups, if any, being housed with third-party providers.

- Pricing models vary by service, but tend to be negotiated. The more sophisticated ASPs have realized that the provision of software over the Internet is not the same as licensing of software, and the contracting vehicles for ASP relationships have slowly morphed from typical licensing models into services arrangements. There is no inherent ability to scale the use or availability of ASP services on demand, nor is it required.

Cloud Service Life Cycle

The input to the production of a cloud services are all the resources and assets that will compose the cloud service (i.e., in the form of hardware,

software, man power required from developer to the management level and cost). The outcome of the cloud services production is an acceptable and marketable cloud service, which will provide a measurable value to the business objectives and outcomes. The sets of inputs are transformed to derive the outcome by using the cloud service life cycle.

At the core of the cloud service life cycle is service strategy, which is the fundamental phase in defining the service principles. The main core of the cloud

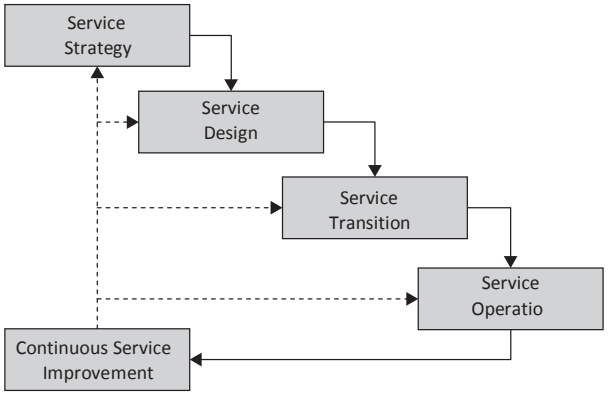


FIGURE 5.1 CLOUD SERVICE LIFE CYCLE

Less emphasis is placed on location of data and processing than in outsourcing, though this information was a generally ascertainable.

- Cloud computing covers multiple service models (i.e, software, infrastruc- ture, and platform as a service). As of this writing, access to cloud computing services are (at least in the public cloud computing frame- work), for the most part, one-size-fits-all 'click here to accept' agreements, not negotiated arrangements. Similarly, pricing tended to be unit-based (hence its comparison to utility computing).

- In the cloud environment, performance economies are important for the profitability of the cloud provider. Therefore a cloud provider may have multiple data centers geographically dispersed to take advantage of geographic cost differentials.

- Additionally, the ability of cloud providers to quickly scale up and down to meet customer requirements dictate that secondary and tertiary data centers be available either directly from the cloud provider or through its subcontracted arrangements. The location of data and processing at any given instant in time tends to be less well known to the customer in a cloud environment.