

CLOUD COMPUTING FUNDAMENTALS

(R20A0554)

LECTURE NOTES

B. TECH

III YEAR – II SEM

2023-2024

Department of Electrical and Electronics Engineering



MALLA REDDY COLLEGE OF ENGINEERING & TECHNOLOGY

(Autonomous Institution – UGC, Govt. of India)

Recognized under 2(f) and 12 (B) of UGC ACT 1956

(Affiliated to JNTUH, Hyderabad, Approved by AICTE - Accredited by NBA & NAAC – 'A' Grade - ISO 9001:2015 Certified)

Maisammaguda, Dhulapally (Post Via. Hakimpet), Secunderabad – 500100, Telangana State, India



MALLA REDDY COLLEGE OF ENGINEERING & TECHNOLOGY
Department of Electrical and Electronics Engineering

III Year B.Tech EEE – II Sem

L/T/P/C
3/-/-/3

(R20A0554) CLOUD COMPUTING

Objectives:

- To learn various system models for Distributed and Cloud Computing.
- To understand about Virtual machines, Its Structure and mechanisms.
- To learn Cloud Computing Paradigm.
- To introduce the various levels of services that can be achieved by cloud.
- To describe the security aspects in cloud.

UNIT- I

Systems Modeling: Distributed System Models and Enabling Technologies- Scalable Computing over the Internet- System Models for Distributed and Cloud Computing- Software Environments for Distributed Systems and Clouds-- Performance, Security, and Energy Efficiency

UNIT- II

Virtualization: Virtual Machines and Virtualization of Clusters and Data Centers- Implementation Levels of Virtualization -Virtualization Structures/Tools and Mechanisms-Virtualization of CPU, Memory, and I/O Devices-Virtual Clusters and Data-Centers

UNIT- III

Foundations: Introduction to Cloud Computing- Migrating into a Cloud-The Enterprise Cloud Computing Paradigm.

UNIT- IV

Infrastructure as a Service (IAAS)& Platform (PAAS): Virtual machines provisioning and Migration services-On the Management of Virtual machines for Cloud Infrastructures-Aneka—Integration of Private and Public Clouds

UNIT- V

Software as a Service (SAAS) &Data Security in the Cloud: Google App Engine – An Introduction to the idea of Data Security- The Current State of Data Security in the Cloud- Cloud Computing and Data Security Risk- Cloud Computing and Identity

TEXT BOOKS:

1. Distributed and Cloud Computing, Kaittwang Geoffrey C.Fox and Jack J Dongrra, Elsevier India 2012.

2. Mastering Cloud Computing- Raj Kumar Buyya, Christian Vecchiola and S.TanuraiSelvi, TMH, 2012.
3. Michael Miller, Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online, Que Publishing, August 2008.

Reference Books:

1. Cloud Computing : A Practical Approach, Anthony T.Velte, Toby J.Velte, Robert Elsenpeter, Tata McGraw Hill, rp2011.
2. Enterprise Cloud Computing, Gautam Shroff, Cambridge University Press, 2010.
3. Cloud Computing: Implementation, Management and Security, John W.Rittinghouse, James F.Ransome, CRC Press, rp2012.
4. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, George Reese, O'reilly, SPD, rp2011.
5. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly, SPD, rp2011.

Outcomes:

- To distinguish the different models and computing paradigms.
- To explain the levels of virtualization and resources virtualization
- To analyze the reasons for migrating into cloud
- To effectively use the cloud services in terms of infrastructure and operating platforms.
- To apply the services in the cloud for real world scenarios

INDEX		
UNIT	Topic	Page No
I	Scalable computing over the Internet	1
	System Models for Distributed and Cloud Computing	6
	Software Environments for distributed Systems and Clouds	11
	Performance, Security & Energy Efficiency	13
II	Implementation Levels of Virtualization	18
	Virtualization Structures/Tools and Mechanisms	21
	Virtualization of CPU, Memory, and I/O Devices	23
	Virtual Clusters and Data Centers	26
III	Introduction to Cloud Computing	27
	Introduction – Migration into cloud	31
	The Enterprise Cloud Computing Paradigm	34
IV	Introduction to IAAS & PAAS	38
	Virtual machines provisioning services	41
	Virtual machines Migration services	42
	Aneka— Integration of Private and Public Clouds	45
	On the Management of Virtual machines for Cloud Infrastructures	46
V	Google App Engine	50
	An Introduction to the idea of Data Security	52
	The Current State of Data Security in the Cloud	53
	Cloud Computing and Identity	54
	Cloud Computing and Data Security Risk	56

UNIT -1

Scalable Computing over the Internet

High-Throughput Computing-HTC

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 measures high throughput or the number of tasks completed per unit of time. HTC technology needs to improve batch processing speed, and also address the acute problems of cost, energy savings, security, and reliability at many data and enterprise computing centers

Computing Paradigm Distinctions

- **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
- Inter Process 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**

- 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.
- Distributed computing system uses multiple computers to solve large-scale problems over the Internet using a centralized computer to solve computational problems.

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.
- Cloud computing can also be a form of utility computing or service computing

Degrees of Parallelism

- **Bit-level parallelism (BLP) :**
 - converts bit-serial processing toward-level processing gradually.
- **Instruction-level parallelism (ILP)**
 - the processor executes multiple instructions simultaneously rather than only one instruction at a time.
 - ILP is executed through pipelining, superscalar computing, VLIW (very long instruction word) architectures, and multithreading.
 - ILP requires branch prediction, dynamic scheduling, speculation, and compiler support to work efficiently.
- **Data-level parallelism (DLP)**
 - DLP through SIMD (single instruction, multiple data) and vector machines using vector or array types of instructions.
 - DLP requires even more hardware support and compiler assistance to work properly.
- **Task-level parallelism (TLP):**
 - Ever since the introduction of multicore processors and chip multiprocessors (CMPs), we have been exploring TLP
 - TLP is far from being very successful due to difficulty in programming and compilation of

code for efficient execution on multicore CMPs.

- **Utility Computing**

- Utility computing focuses on a business model in which customers receive computing resources from a paid service provider. All grid/cloud platforms are regarded as utility service providers.

- **The Internet of Things (IoT)**

- Traditional Internet connects machines to machines or web pages to web pages.
- IoT was introduced in 1999 at MIT
- networked interconnection of everyday objects, tools, devices, or computers
- a wireless network of sensors that interconnect all things in our daily life.
- Three communication patterns co-exist: namely H2H (human-to-human), H2T (human-to-thing), and T2T (thing-to-thing).
- connect things (including human and machine objects) at any time and any place intelligently with low cost
- IPv6 protocol, 2¹²⁸ IP addresses are available to distinguish all the objects on Earth, including all computers and pervasive devices
- IoT needs to be designed to track 100 trillion static or moving objects simultaneously.
- IoT demands universal addressability of all of the objects or things.
- The dynamic connections will grow exponentially into a new dynamic network of networks, called the Internet of Things (IoT).

Cyber-Physical Systems

- A cyber-physical system (CPS) is the result of interaction between computational processes and the physical world.
- CPS integrates “cyber” (heterogeneous, asynchronous) with “physical” (concurrent and information-dense) objects
- CPS merges the “3C” technologies of computation, communication, and control into an intelligent closed feedback system

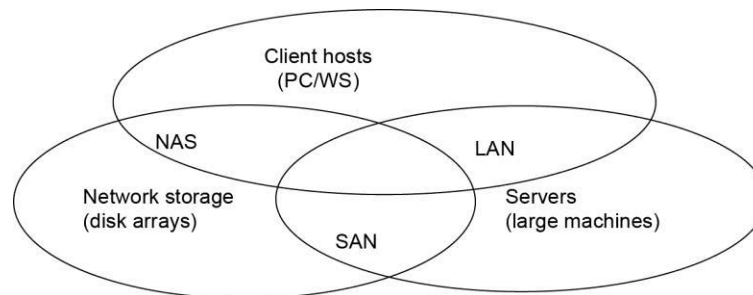
IoT emphasizes various networking connections among physical objects, while the CPS emphasizes exploration of virtual reality (VR) applications in the physical world

TECHNOLOGIES FOR NETWORK-BASED SYSTEMS

Multicore CPUs and Multithreading Technologies: Today, advanced CPUs or microprocessor chips assume a multicore architecture with dual, quad, six, or more processing cores. These processors exploit parallelism at ILP and TLP levels. Both multi-core CPU and many-core GPU processors can handle multiple instruction threads at different magnitudes today. Multiple cores are housed in the same chip with an L2 cache that is shared by all cores. In the future, multiple CMPs could be built on the same CPU chip with even the L3 cache on the chip. Multicore and multithreaded CPUs are equipped with many high-end processors, including the Intel i7, Xeon, AMD Opteron, Sun Niagara, IBM Power 6, and X cell processors. Each core could be also multithreaded.

Memory, Storage, and Wide-Area Networking: Memory chips have experienced a 4x increase in capacity every three years. For hard drives, capacity increased from 260 MB in 1981 to 250 GB in 2004. Disks or disk arrays have exceeded 3 TB in capacity. The rapid growth of flash memory and solid-state drives (SSDs) also impacts the future of HPC and HTC systems.

System-Area Interconnects: The nodes in small clusters are mostly interconnected by an Ethernet switch or a local area network (LAN).

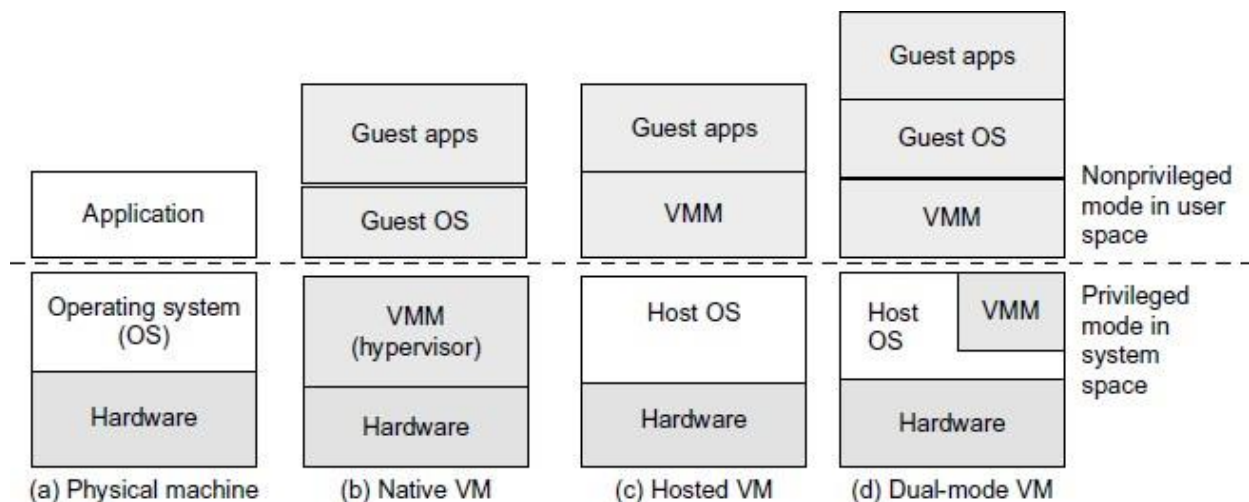


As Figure shows, a LAN typically is used to connect client hosts to big servers. A storage area network (SAN) connects servers to network storage such as disk arrays. Network attached storage (NAS) connects client hosts directly to the disk arrays. All three types of networks often appear in a large cluster built with commercial network components.

Wide-Area Networking: High-bandwidth networking increases the capability of building massively distributed systems. The rapid growth of Ethernet bandwidth from 10 Mbps in 1979 to 1 Gbps in 1999, and 40 ~ 100 GE in 2011. It has been speculated that 1 Tbps network links will become available by 2013.

Virtual Machines and Virtualization Middleware

Virtual machines (VMs) offer novel solutions to underutilized resources, application inflexibility, software manageability, and security concerns in existing physical machines. Today, to build large clusters, grids, and clouds, we need to access large amounts of computing, storage, and networking resources in a virtualized manner. We need to aggregate those resources, and hopefully, offer a single system image. In particular, a cloud of provisioned resources must rely on virtualization of processors, memory, and I/O facilities dynamically.



Three VM architectures in (b), (c), and (d), compared with the traditional physical machine shown in (a).

The host machine is equipped with the physical hardware. The VM is built with virtual resources managed by a guest OS to run a specific application. Between the VMs and the host platform, one needs to deploy a middleware layer called a virtual machine monitor (VMM).

Figure shows a native VM installed with the use of a VMM called a hypervisor in privileged Mode.

The guest OS could be a Linux system and the hypervisor is the XEN system developed at Cambridge University. This hypervisor approach is also called bare-metal VM, because the hypervisor handles the bare hardware (CPU, memory, and I/O) directly. Architecture is the host VM shown in Figure(c). Here the VMM runs in non-privileged mode. The host OS need not be modified. The VM can also be implemented with a dual mode, as shown in Figure 1.12(d). Part of the VMM runs at the user level and another part runs at the supervisor level. In this case, the host OS may have to be modified to some extent. Multiple VMs can be ported to a given hardware system to support the virtualization process. The VM approach offers hardware independence of the OS and applications.

VM Primitive Operations: The VMM provides the VM abstraction to the guest OS. With full virtualization, the VMM exports a VM abstraction identical to the physical machine so that a standard OS such as Windows 2000 or Linux can run just as it would on the physical hardware

Low-level VMM operations are

- the VMs can be multiplexed between hardware machines,
- a VM can be suspended and stored in stable storage
- a suspended VM can be resumed or provisioned to a new hardware platform
- a VM can be migrated from one hardware platform to another

These VM operations enable a VM to be provisioned to any available hardware platform. They also enable flexibility in porting distributed application executions. Furthermore, the VM approach will significantly enhance the utilization of server resources.

SYSTEM MODELS FOR DISTRIBUTED AND CLOUD COMPUTING

- Distributed and cloud computing systems are built over a large number of autonomous

Table 1.2 Classification of Parallel and Distributed Computing Systems				
Functionality, Applications	Computer Clusters [10,28,38]	Peer-to-Peer Networks [34,46]	Data/ Computational Grids [6,18,51]	Cloud Platforms [1,9,11,12,30]
Architecture, Network Connectivity, and Size	Network of compute nodes interconnected by SAN, LAN, or WAN hierarchically	Flexible network of client machines logically connected by an overlay network	Heterogeneous clusters interconnected by high-speed network links over selected resource sites	Virtualized cluster of servers over data centers via SLA
Control and Resources Management	Homogeneous nodes with distributed control, running UNIX or Linux	Autonomous client nodes, free in and out, with self-organization	Centralized control, server-oriented with authenticated security	Dynamic resource provisioning of servers, storage, and networks
Applications and Network-centric Services	High-performance computing, search engines, and web services, etc.	Most appealing to business file sharing, content delivery, and social networking	Distributed supercomputing, global problem solving, and data center services	Upgraded web search, utility computing, and outsourced computing services
Representative Operational Systems	Google search engine, SunBlade, IBM Road Runner, Cray XT4, etc.	Gnutella, eMule, BitTorrent, Napster, KaZaA, Skype, JXTA	TeraGrid, GriPhyN, UK EGEE, D-Grid, ChinaGrid, etc.	Google App Engine, IBM Bluecloud, AWS, and Microsoft Azure

computer nodes. These node machines are interconnected by SANs, LANs, or WANs

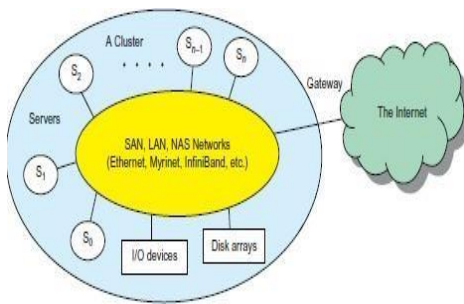
- A massive system is with millions of computers connected to edge networks.
- Massive systems are considered highly scalable
- massive systems are classified into four groups: clusters, P2P networks, computing grids, and Internet clouds

Computing cluster

- A computing cluster consists of interconnected stand-alone computers which work cooperatively as a single integrated computing resource.

Cluster Architecture

- the architecture consists of a typical server cluster built around a low-latency, high bandwidth interconnection network.
- build a larger cluster with more nodes, the interconnection network can be built with multiple levels of Gigabit Ethernet, Myrinet, or InfiniBand switches.
- Through hierarchical construction using a SAN, LAN, or WAN, one can build scalable clusters with an increasing number of nodes
- cluster is connected to the Internet via a virtual private network (VPN) gateway.
- gateway IP address locates the cluster



- Clusters have loosely coupled node computers.
- All resources of a server node are managed by their own OS.
- Most clusters have multiple system images as a result of having many autonomous nodes under different OS control

Single-System Image -Cluster

- an ideal cluster should merge multiple system images into a single-system image (SSI)
- cluster operating system or some middleware have to support SSI at various levels, including the sharing of CPUs, memory, and I/O across all cluster nodes.
- illusion created by software or hardware that presents a collection of resources as one integrated, powerful resource
- SSI makes the cluster appear like a single machine to the user.
- A cluster with multiple system images is nothing but a collection of independent ~~compts~~ **compts**
-

Hardware, Software, and Middleware Support –Cluster

- Clusters exploring massive parallelism are commonly known as MPPs –Massive Parallel Processing

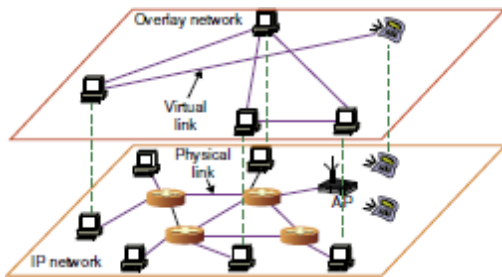
- The building blocks are computer nodes (PCs, workstations, servers, or SMP), special communication software such as PVM or MPI, and a network interface card in each computer node.
- Most clusters run under the Linux OS.
- nodes are interconnected by a high-bandwidth network
- Special cluster middleware supports are needed to create SSI or high availability (HA).
- all distributed memory to be shared by all servers by forming distributed shared memory (DSM).
- SSI features are expensive
- achieving SSI, many clusters are loosely coupled machines
- virtual clusters are created dynamically, upon user demand

Grid Computing

- A web service such as HTTP enables remote access of remote web pages
- computing grid offers an infrastructure that couples computers, software/middleware, special instruments, and people and sensors together
- Enterprises or organizations present grids as integrated computing resources. They can also be viewed as virtual platforms to support virtual organizations.
- The computers used in a grid are primarily workstations, servers, clusters, and supercomputers

Peer-to-Peer Network-P2P

- P2P architecture offers a distributed model of networked systems.
- P2P network is client-oriented instead of server-oriented
- In a P2P system, every node acts as both a client and a server
- Peer machines are simply client computers connected to the Internet.
- All client machines act autonomously to join or leave the system freely. This implies that no master-slave relationship exists among the peers.
- No central coordination or central database is needed. The system is self-organizing with distributed control.
- P2P two layer of abstractions as given in the figure

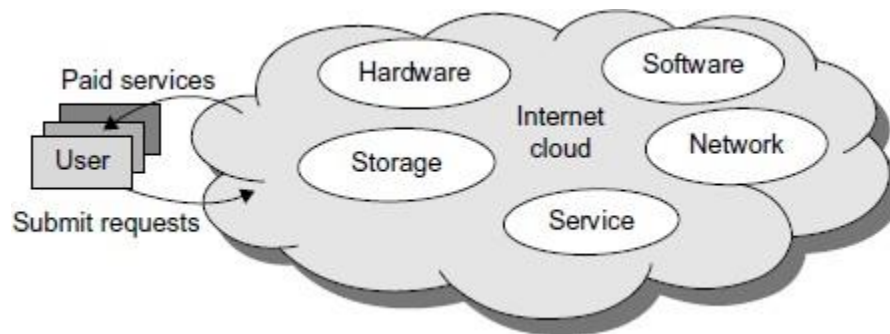


- Each peer machine joins or leaves the P2P network voluntarily
- Only the participating peers form the physical network at any time.
- Physical network is simply an ad hoc network formed at various Internet domains randomly using the TCP/IP and NAI protocols.

Peer-to-Peer Network-Overlay network

- Data items or files are distributed in the participating peers.
- Based on communication or file-sharing needs, the peer IDs form an overlay network at the logical level.
- When a new peer joins the system, its peer ID is added as a node in the overlay network.
- When an existing peer leaves the system, its peer ID is removed from the overlay network automatically.
- An unstructured overlay network is characterized by a random graph. There is no fixed route to send messages or files among the nodes. Often, flooding is applied to send a query to all nodes in an unstructured overlay, thus resulting in heavy network traffic and nondeterministic search results.
- Structured overlay networks follow certain connectivity topology and rules for inserting and removing nodes (peer IDs) from the overlay graph

Cloud Computing



Virtualized resources from data centers to form an Internet cloud

- A cloud is a pool of virtualized computer resources.
- A cloud can host a variety of different workloads, including batch-style backend jobs and interactive and user-facing applications.”
- Cloud computing applies a virtualized platform with elastic resources on demand by provisioning hardware, software, and data sets dynamically

The Cloud Landscape

Infrastructure as a Service (IaaS)

- This model puts together infrastructures demanded by users—namely servers, storage, networks, and the data center fabric.
- The user can deploy and run on multiple VMs running guest OSes on specific applications.
- The user does not manage or control the underlying cloud infrastructure, but can specify when to request and release the needed resources.

Platform as a Service (PaaS)

- This model enables the user to deploy user-built applications onto a virtualized cloud platform.
- PaaS includes middleware, databases, development tools, and some runtime support such as Web 2.0 and Java.
- The platform includes both hardware and software integrated with specific programming interfaces.
- The provider supplies the API and software tools (e.g., Java, Python, Web 2.0, .NET). The user is freed from managing the cloud infrastructure.

Software as a Service (SaaS)

- This refers to browser-initiated application software over thousands of paid cloud customers.
- The SaaS model applies to business processes, industry applications, consumer relationship management (CRM), enterprise resources planning (ERP), human resources (HR), and collaborative applications.
- On the customer side, there is no upfront investment in servers or software licensing.
- On the provider side, costs are rather low, compared with conventional hosting of user applications

Internet clouds offer four deployment modes: private, public, managed, and hybrid

SOFTWARE ENVIRONMENTS FOR DISTRIBUTED SYSTEMS AND CLOUDS

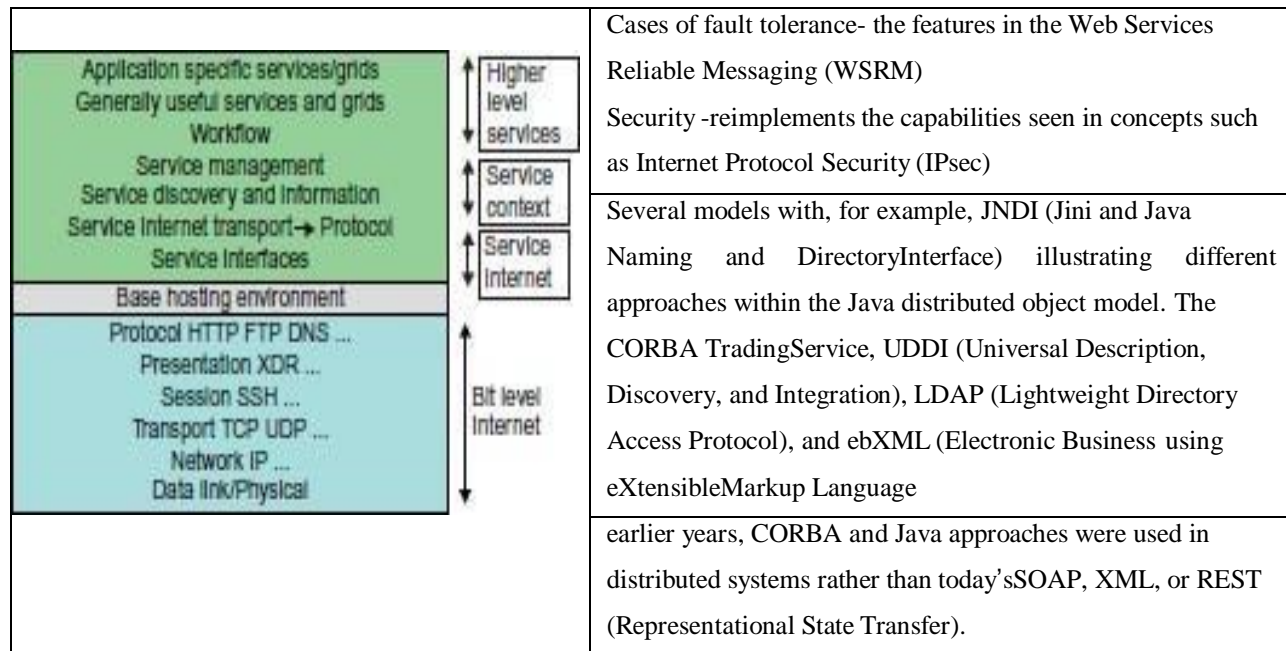
Service-Oriented Architecture (SOA)

- In grids/web services, Java, and CORBA, an entity is, respectively, a service, a Java object, and a CORBA distributed object in a variety of languages.
- These architectures build on the traditional seven Open Systems Interconnection (OSI) layers that provide the base networking abstractions.
- On top of this we have a base software environment, which would be
 - .NET or Apache Axis for web services,
 - the Java Virtual Machine for Java, and a broker network for CORBA
- On top of this base environment one would build a higher level environment reflecting the special features of the distributed computing environment.
- SOA applies to building grids, clouds, grids of clouds, clouds of grids, clouds of clouds (also known as interclouds),
- SS (sensor service) : A large number of sensors provide data-collection services (ZigBee device, a Bluetooth device, WiFi access point, a personal computer, a GPA, or a wireless phone etc)
- Filter services : to eliminate unwanted raw data, in order to respond to specific requests from the web, the grid, or web services

Layered Architecture for Web Services and Grids

- **Entity Interfaces**
- Java method interfaces correspond to the Web Services Description Language (WSDL),
- CORBA interface - definition language (IDL) specifications

- These interfaces are linked with customized, high-level communication systems: SOAP, RMI, and IIOP
- These communication systems support features including particular message patterns (such as Remote Procedure Call or RPC), fault recovery, and specialized routing.
- Communication systems are built on message-oriented middleware (enterprise bus) infrastructure such as Web-Sphere MQ or Java Message Service (JMS)



Web Services and Tools

REST approach:

- delegates most of the difficult problems to application (implementation-specific) software. In a web services language
- minimal information in the header, and the message body (that is opaque to generic message processing) carries all the needed information.
- architectures are clearly more appropriate for rapid technology environments.
- REST can use XML schemas but not those that are part of SOAP; "XML over HTTP" is a popular design choice in this regard.
- Above the communication and management layers, we have the ability to compose new entities or distributed programs by integrating several entities together.

CORBA and Java:

- the distributed entities are linked with RPCs, and the simplest way to build composite

applications is to view the entities as objects and use the traditional ways of linking them together.

- For Java, this could be as simple as writing a Java program with method calls replaced by Remote Method Invocation (RMI),
- CORBA supports a similar model with a syntax reflecting the C++ style of its entity (object) interfaces.

Parallel and Distributed Programming Models

Table 1.7 Parallel and Distributed Programming Models and Tool Sets		
Model	Description	Features
MPI	A library of subprograms that can be called from C or FORTRAN to write parallel programs running on distributed computer systems [6,28,42]	Specify synchronous or asynchronous point-to-point and collective communication commands and I/O operations in user programs for message-passing execution
MapReduce	A web programming model for scalable data processing on large clusters over large data sets, or in web search operations [16]	Map function generates a set of intermediate key/value pairs; Reduce function merges all intermediate values with the same key
Hadoop	A software library to write and run large user applications on vast data sets in business applications (http://hadoop.apache.org/core)	A scalable, economical, efficient, and reliable tool for providing users with easy access of commercial clusters

PERFORMANCE, SECURITY, AND ENERGY EFFICIENCY

Performance Metrics:

- In a distributed system, performance is attributed to a large number of factors.
- System throughput is often measured in MIPS, Flops (tera floating-point operations per second), or TPS (transactions per second).
- System overhead is often attributed to OS boot time, compile time, I/O data rate, and the runtime support system used.
- Other performance-related metrics include the QoS for Internet and web services; system availability and dependability; and security resilience for system defense against network attacks

Dimensions of Scalability

Any resource upgrade in a system should be backward compatible with existing hardware and

software resources. System scaling can increase or decrease resources depending on many practical factors

Size scalability

- This refers to achieving higher performance or more functionality by increasing the machine size.
- The word “size” refers to adding processors, cache, memory, storage, or I/O Channels. The most obvious way to determine size scalability is to simply count the number of processors installed.
- Not all parallel computer or distributed architectures are equally size scalable.
- For example, the IBM S2 was scaled up to 512 processors in 1997. But in 2008, the IBM BlueGene/L system scaled up to 65,000 processors.

• Software scalability

- This refers to upgrades in the OS or compilers, adding mathematical and engineering libraries, porting new application software, and installing more user- friendly programming environments.
- Some software upgrades may not work with large system configurations.
- Testing and fine-tuning of new software on larger systems is a nontrivial job.

• Application scalability

- This refers to matching problem size scalability with machine size scalability.
- Problem size affects the size of the data set or the workload increase. Instead of increasing machine size, users can enlarge the problem size to enhance system efficiency or cost-effectiveness.

• Technology scalability

- This refers to a system that can adapt to changes in building technologies, such as the component and networking technologies
- When scaling a system design with new technology one must consider three aspects: time, space, and heterogeneity.
- (1) Time refers to generation scalability. When changing to new-generation processors, one must consider the impact to the motherboard, power supply, packaging and cooling, and so forth. Based on past experience, most systems upgrade their commodity processors every three to five years.

- (2) Space is related to packaging and energy concerns. Technology scalability demands harmony and portability among suppliers.
- (3) Heterogeneity refers to the use of hardware components or software packages from different vendors. Heterogeneity may limit the scalability.

Amdahl's Law

- Let the program has been parallelized or partitioned for parallel execution on a cluster of many processing nodes.
- Assume that a fraction α of the code must be executed sequentially, called the sequential bottleneck.
- Therefore, $(1 - \alpha)$ of the code can be compiled for parallel execution by n processors.

The total execution time of the program is calculated by $\alpha T + (1 - \alpha)T/n$, where the first term is the sequential execution time on a single processor and the second term is the parallel execution time on n processing nodes.

- I/O time or exception handling time is also not included in the following speedup analysis.

$$\text{Speedup} = S = T / [\alpha T + (1 - \alpha)T/n] = 1 / [\alpha + (1 - \alpha)/n]$$

- Amdahl's Law states that the speedup factor of using the n -processor system over the use of a single processor is expressed by:
- the code is fully parallelizable with $\alpha = 0$. As the cluster becomes sufficiently large, that is, $n \rightarrow \infty$, S approaches $1/\alpha$, an upper bound on the speedup S .
- this upper bound is independent of the cluster size n . The sequential bottleneck is the portion of the code that cannot be parallelized.

Gustafson's Law

- To achieve higher efficiency when using a large cluster, we must consider scaling the problem size to match the cluster capability. This leads to the following speedup proposed by John Gustafson (1988), referred as scaled-workload speedup.
- Let W be the workload in a given program.

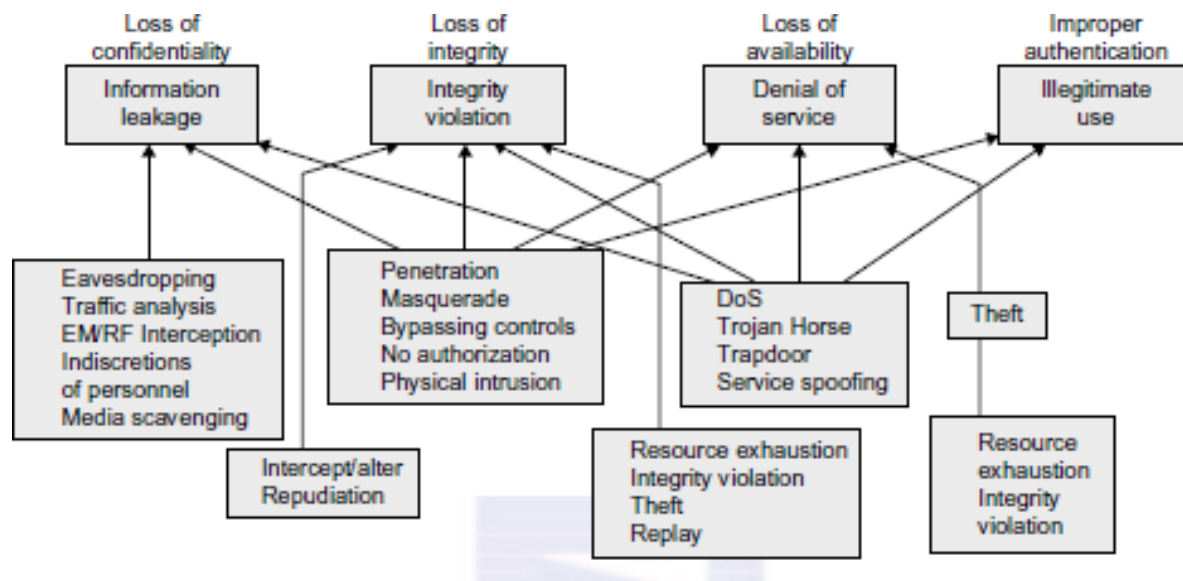
- When using an n-processor system, the user scales the workload to

$W' = \alpha W + (1 - \alpha)nW$. Scaled workload W' is essentially the sequential execution time

in a single processor. The parallel execution time of a scaled workload W' on n processors is defined by a scaled-workload speedup as follows:

$$S' = W'/W = [\alpha W + (1 - \alpha)nW]/W = \alpha + (1 - \alpha)n$$

Network Threats and Data Integrity



ENERGY EFFICIENCY IN DISTRIBUTED COMPUTING

Primary performance goals in conventional parallel and distributed computing systems are highperformance and high throughput, considering some form of performance reliability (e.g., fault toleranceand security). However, these systems recently encountered new challenging issues includingenergy efficiency, and workload and resource outsourcing

Energy Consumption of Unused Servers: To run a server farm (data center) a company has to spend a huge amount of money for hardware,software, operational support, and energy every year. Therefore, companies should thoroughlyidentify whether their installed server farm (more specifically, the volume of provisioned resources)is at an appropriate level, particularly in terms of utilization.

Reducing Energy in Active Servers: In addition to identifying unused/underutilized servers for energy savings, it is also necessary to apply appropriate techniques to decrease energy consumption in active distributed systems with negligible influence on their performance.

Application Layer: Until now, most user applications in science, business, engineering, and financial areas tend to increase a system's speed or quality. By introducing energy-aware applications, the challenge is to design sophisticated multilevel and multi-domain energy management applications without hurting performance.

Middleware Layer: The middleware layer acts as a bridge between the application layer and the resource layer. This layer provides resource broker, communication service, task analyzer, task scheduler, security access, reliability control, and information service capabilities. It is also responsible for applying energy-efficient techniques, particularly in task scheduling.

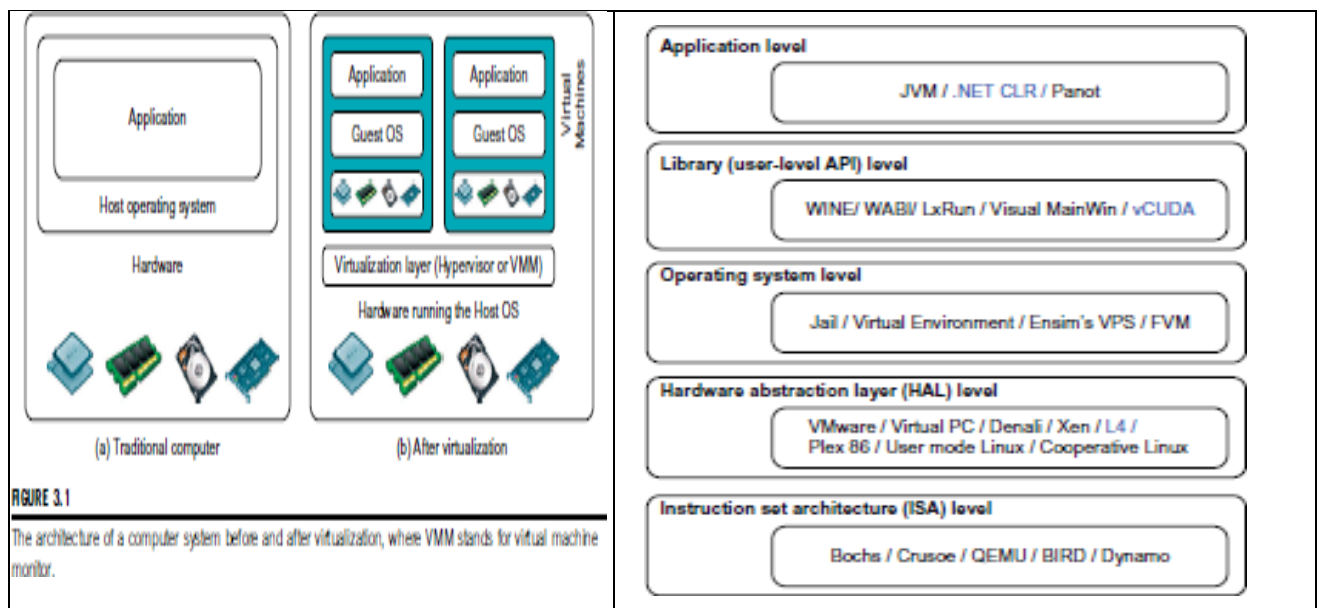
Resource Layer: The resource layer consists of a wide range of resources including computing nodes and storage units. This layer generally interacts with hardware devices and the operating system; therefore, it is responsible for controlling all distributed resources in distributed computing systems. Dynamic power management (DPM) and dynamic voltage-frequency scaling (DVFS) are two popular methods incorporated into recent computer hardware systems. In DPM, hardware devices, such as the CPU, have the capability to switch from idle mode to one or more lower power modes. In DVFS, energy savings are achieved based on the fact that the power consumption in CMOS circuits has a direct relationship with frequency and the square of the voltage supply.

Network Layer: Routing and transferring packets and enabling network services to the resource layer are the main responsibility of the network layer in distributed computing systems. The major challenge to build energy-efficient networks is, again, determining how to measure, predict, and create a balance between energy consumption and performance.

UNIT-2

Levels of Virtualization Implementation

- Virtualization is a computer architecture technology by which multiple virtual machines (VMs) are multiplexed in the same hardware machine.
- After virtualization, different user applications managed by their own operating systems (guest OS) can run on the same hardware independent of the host OS
- done by adding additional software, called a virtualization layer
- This virtualization layer is known as hypervisor or virtual machine monitor (VMM)



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

Instruction Set Architecture Level

- At the ISA level, virtualization is performed by emulating a given ISA by the ISA of the host machine. For example, MIPS binary code can run on an x86-based host machine with the help of ISA emulation. With this approach, it is possible to run a large amount of legacy binary code written for various processors on any given new hardware host machine.

- Instruction set emulation leads to virtual ISAs created on any hardware machine. The basic emulation method is through code interpretation. An interpreter program interprets the source instructions to target instructions one by one. One source instruction may require tens or hundreds of native target instructions to perform its function. Obviously, this process is relatively slow. For better performance, dynamic binary translation is desired.
- This approach translates basic blocks of dynamic source instructions to target instructions. The basic blocks can also be extended to program traces or super blocks to increase translation efficiency.
- Instruction set emulation requires binary translation and optimization. A virtual instruction set architecture (V-ISA) thus requires adding a processor-specific software translation layer to the compiler.

Hardware Abstraction Level

- Hardware-level virtualization is performed right on top of the bare hardware.
- This approach generates a virtual hardware environment for a VM.
- The process manages the underlying hardware through virtualization. The idea is to virtualize a computer's resources, such as its processors, memory, and I/O devices.
- The intention is to upgrade the hardware utilization rate by multiple users concurrently. The idea was implemented in the IBM VM/370 in the 1960s.
- More recently, the Xen hypervisor has been applied to virtualize x86-based machines to run Linux or other guest OS applications.

Operating System Level

- This refers to an abstraction layer between traditional OS and user applications.
- OS-level virtualization creates isolated containers on a single physical server and the OS instances to utilize the hardware and software in data centers.
- The containers behave like real servers.
- OS-level virtualization is commonly used in creating virtual hosting environments to allocate hardware resources among a large number of mutually distrusting users.
- It is also used, to a lesser extent, in consolidating server hardware by moving services on separate hosts into containers or VMs on one server.

Library Support Level

- Most applications use APIs exported by user-level libraries rather than using lengthy system calls by the OS.
- Since most systems provide well-documented APIs, such an interface becomes another candidate for virtualization.
- Virtualization with library interfaces is possible by controlling the communication link between applications and the rest of a system through API hooks.

User-Application Level

- Virtualization at the application level virtualizes an application as a VM.
- On a traditional OS, an application often runs as a process. Therefore, application-level virtualization is also known as process-level virtualization.
- The most popular approach is to deploy high level language (HLL) VMs. In this scenario, the virtualization layer sits as an application program on top of the operating system,
- The layer exports an abstraction of a VM that can run programs written and compiled to a particular abstract machine definition.
- Any program written in the HLL and compiled for this VM will be able to run on it. The Microsoft .NET CLR and Java Virtual Machine (JVM) are two good examples of this class of VM.

VMM Design Requirements and Providers

- layer between real hardware and traditional operating systems. This layer is commonly called the Virtual Machine Monitor (VMM)
- three requirements for a VMM
- a VMM should provide an environment for programs which is essentially identical to the original machine
- programs run in this environment should show, at worst, only minor decreases in speed
- VMM should be in complete control of the system resources.
- VMM includes the following aspects:
 - (1) The VMM is responsible for allocating hardware resources for programs;
 - (2) it is not possible for a program to access any resource not explicitly allocated to it;
 - (3) it is possible under certain circumstances for a VMM to regain control of resources already allocated.

Virtualization Support at the OS Level

- Why OS-Level Virtualization? :
 - it is slow to initialize a hardware-level VM because each VM creates its own image from scratch.
- OS virtualization inserts a virtualization layer inside an operating system to partition a machine's physical resources.
- It enables multiple isolated VMs within a single operating system kernel.
- This kind of VM is often called a virtual execution environment (VE), Virtual Private System (VPS), or simply container
- The benefits of OS extensions are twofold:
 - (1) VMs at the operating system level have minimal startup/shutdown costs, low resource requirements, and high scalability;
 - (2) for an OS-level VM, it is possible for a VM and its host environment to synchronize state changes when necessary.

Middleware Support for Virtualization

- Library-level virtualization is also known as user-level Application Binary Interface (ABI) or API emulation.
- This type of virtualization can create execution environments for running alien programs on a platform

Hypervisor and Xen Architecture

- The hypervisor software sits directly between the physical hardware and its OS.
- This virtualization layer is referred to as either the VMM or the hypervisor

Xen Architecture

- Xen is an open source hypervisor program developed by Cambridge University.
- Xen is a microkernel hypervisor
- The core components of a Xen system are the hypervisor, kernel, and applications
- The guest OS, which has control ability, is called **Domain 0**, and the others are called **Domain U**
- Domain 0 is designed to access hardware directly and manage devices

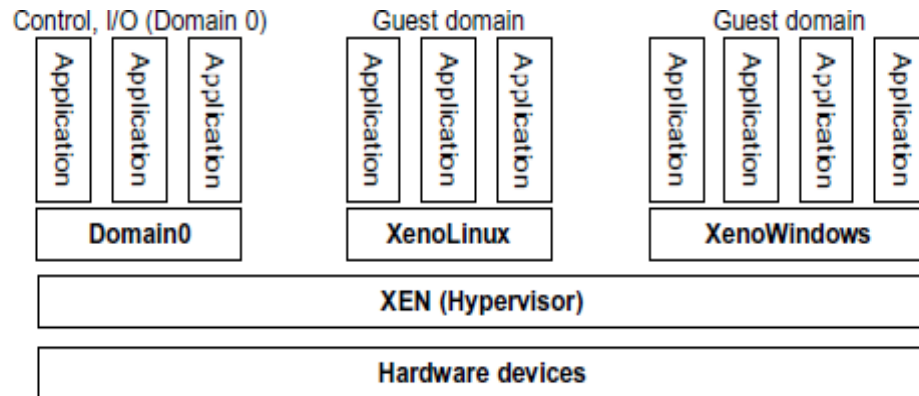


FIGURE 3.5

The Xen architecture's special domain 0 for control and I/O, and several guest domains for user applications.

(Courtesy of P. Barham, et al. [7])

- VM state is akin to a tree: the current state of the machine is a point that progresses monotonically as the software executes.
- VMs are allowed to roll back to previous states in their execution (e.g., to fix configuration errors) or rerun from the same point many times

Full virtualization

- Full virtualization, noncritical instructions run on the hardware directly while critical instructions are discovered and replaced with traps into the VMM to be emulated by software
- **VMware** puts the **VMM at Ring 0** and the **guest OS at Ring 1**.
- The VMM scans the instruction stream and identifies the privileged, control- and behavior-sensitive instructions.
- When these instructions are identified, they are trapped into the VMM, which emulates the behavior of these instructions.
- The method used in this emulation is called binary translation.
- Therefore, **full virtualization combines binary translation and direct execution**.

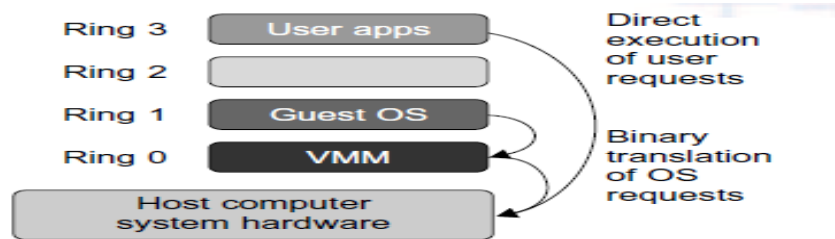


FIGURE 3.6
Indirect execution of complex instructions via binary translation of guest OS requests using the VMM plus direct execution of simple instructions on the same host.
(Courtesy of VM Ware [71])

Para-Virtualization

- Para-virtualization needs to modify the guest operating systems
- A para-virtualized VM provides special APIs requiring substantial OS modifications in user applications

CPU Virtualization

- A CPU architecture is virtualizable if it supports the ability to run the VM's privileged and unprivileged instructions in the CPU's user mode while the VMM runs in supervisor mode.
- Hardware-Assisted CPU Virtualization: This technique attempts to simplify virtualization because full or paravirtualization is complicated

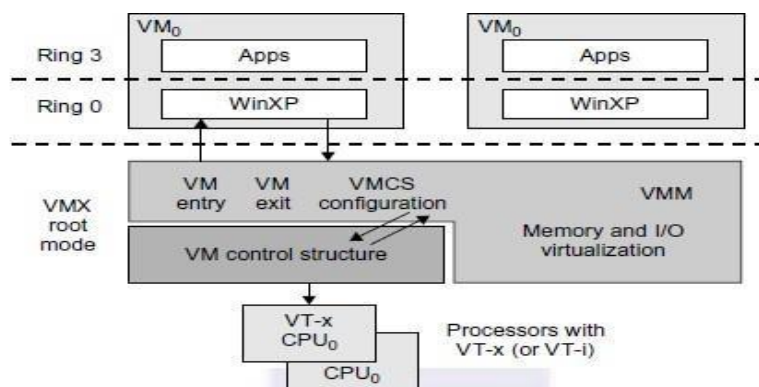


FIGURE 3.11
Intel hardware-assisted CPU virtualization.
(Modified from [68], Courtesy of Lizhong Chen, USC)

Memory Virtualization

- **Memory Virtualization** :the operating system maintains mappings of virtual memory to machine memory using page table
- All modern x86 CPUs include a memory management unit (MMU) and a translation lookaside buffer (TLB) to optimize virtual memory performance
- Two-stage mapping process should be maintained by the guest OS and the VMM,

respectively: virtual memory to physical memory and physical memory to machine memory.

- The VMM is responsible for mapping the guest physical memory to the actual machine memory.

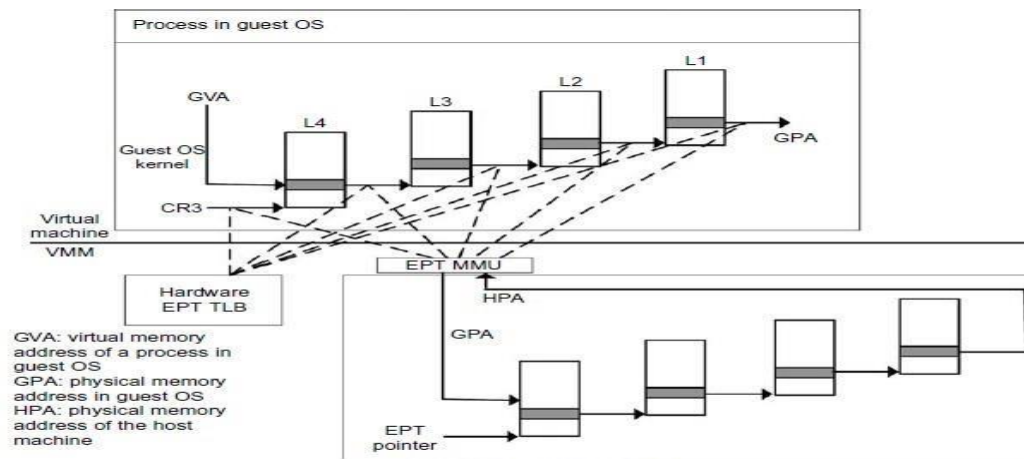


FIGURE 3.13
Memory virtualization using EPT by Intel (the EPT is also known as the shadow page table [68]).

I/O Virtualization

- I/O Virtualization managing the routing of I/O requests between virtual devices and the shared physical hardware
- managing the routing of I/O requests between virtual devices and the shared physical hardware
- Full device emulation emulates well-known, real-world devices All the functions of a device or bus infrastructure, such as device enumeration, identification, interrupts, and DMA, are replicated in software. This software is located in the VMM and acts as a virtual device
- Two-stage mapping process should be maintained by the guest OS and the VMM, respectively: virtual memory to physical memory and physical memory to machine memory.
- The VMM is responsible for mapping the guest physical memory to the actual machine memory.

Virtualization in Multi-Core Processors

- **Muti-core virtualization** has raised some new **challenges**
- **Two difficulties**: Application programs must be parallelized to use all cores fully, and software must explicitly
- Assign tasks to the cores, which is a very complex problem

- The **first challenge**, new programming models, languages, and libraries are needed to make parallel programming easier.
- The **second challenge** has spawned research involving scheduling algorithms and resource management policies
- **Dynamic heterogeneity** is emerging to mix the fat CPU core and thin GPU cores on the same chip

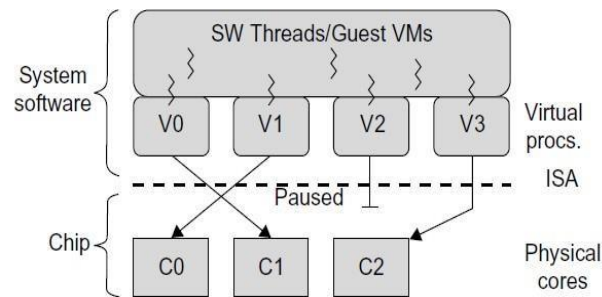


FIGURE 3.16

Multicore virtualization method that exposes four VCPUs to the software, when only three cores are actually present.

(Courtesy of Wells, et al. [74])

- In **many-core chip multiprocessors (CMPs)** →
- Instead of supporting **time-sharing jobs** on one or a few cores, use the abundant cores → **space-sharing**, where single-threaded or multithreaded jobs are simultaneously assigned to separate groups of cores

Physical versus Virtual Clusters

- Virtual clusters are built with VMs installed at distributed servers from one or more physical clusters.
- Assign tasks to the cores, which is a very complex problem
- Fast deployment
- High-Performance Virtual Storage
- reduce duplicated blocks

Virtual Clusters

- **Four ways to manage a virtual cluster.**
- First, you can use a **guest-based manager**, by which the cluster manager resides on a guest system.
- The **host-based manager** supervises the guest systems and can restart the guest system on another physical machine

- Third way to manage a virtual cluster is to use an **independent cluster manager** on both the host and guest systems.
- Finally, use an **integrated cluster** on the guest and host systems.
- This means the manager must be designed to distinguish between virtualized resources and physical resources

Virtualization for data-center automation

- **Data-center automation** means that huge volumes of hardware, software, and database resources in these data centers can be allocated dynamically to millions of Internet users simultaneously, with guaranteed QoS and cost-effectiveness
- This **automation** process is triggered by the growth of virtualization products and cloud computing services.
- The latest virtualization development highlights high availability (HA), backup services, workload balancing, and further increases in client bases.

Server Consolidation in Data Centers

- heterogeneous workloads -chatty workloads and noninteractive workloads
- **Server consolidation** is an approach to improve the low utility ratio of hardware resources by reducing the number of physical servers

Virtual Storage Management

- **storage virtualization** has a different meaning in a system virtualization environment
- **system virtualization**, virtual storage includes the storage managed by VMMs and guest OSes data stored in this environment can be classified into two categories: VM images and application data.

Cloud OS for Virtualized Data Centers

- Data centers must be virtualized to serve as cloud providers
 - **Eucalyptus for Virtual Networking of Private Cloud** : Eucalyptus is an open source software system intended mainly for supporting Infrastructure as a Service (IaaS) clouds
- The system primarily supports **virtual networking** and the management of **VMs**;
- virtual storage is not supported.
- Its purpose is to build **private clouds**
- three resource managers: Instance Manager, Group Manager, Cloud Manager

UNIT-3

Introduction to Cloud Computing

- **Cloud computing** allowing **access** to large amounts of **computing power** in a **fully virtualized** manner, by aggregating resources and offering a single system view
- **Cloud computing** has been coined as an umbrella term to describe a category of sophisticated on-demand computing services initially offered by commercial providers, such as Amazon, Google, and Microsoft.
- The main **principle** behind this model is offering computing, storage, and software “**as a service**”
- **Cloud** is a parallel and distributed computing system consisting of a collection of **inter-connected and virtualised computers** that are dynamically **provisioned**
- The National Institute of Standards and Technology (NIST) characterizes cloud computing as “. . . a pay-per-use model for enabling available, convenient, on-demand network access to a shared pool of configurable computing resources

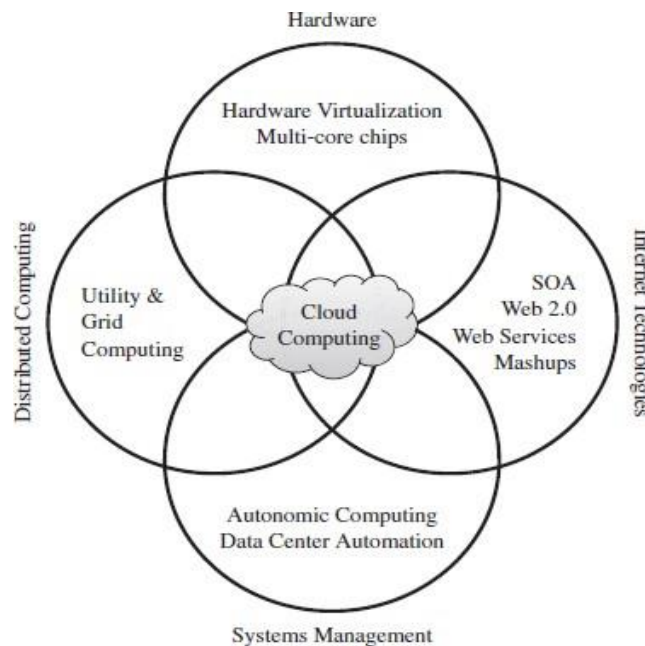
Common characteristics a cloud should have:

- pay-per-use (no ongoing commitment, utility prices);
- elastic capacity and the illusion of infinite resources;
- self-service interface; and
- resources that are abstracted or virtualised.

ROOTS OF CLOUD COMPUTING

- **Autonomic Computing**: seeks to improve systems by decreasing human involvement in their operation
- **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 - four properties of autonomic systems: self-configuration, selfoptimization, self-healing, and self-protection.
- **Reference model** for autonomic control loops of **autonomic managers**, called MAPE-K

(Monitor Analyze Plan Execute—Knowledge)



LAYERS AND TYPES OF CLOUDS

- **Cloud computing services** are divided into three classes:
- (1) Infrastructure as a Service,
- (2) Platform as a Service, and
- (3) Software as a Service.
- **Infrastructure as a Service** :A cloud infrastructure enables on-demand provisioning of servers running several choices of operating systems and a customized software stack.
 - **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.
- **Platform as a Service**: A cloud platform offers an environment on which developers create and deploy applications
 - **Google AppEngine**, an example of Platform as a Service, offers a scalable environment for developing and hosting Web applications
- **Software as a Service**: Traditional desktop applications 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)

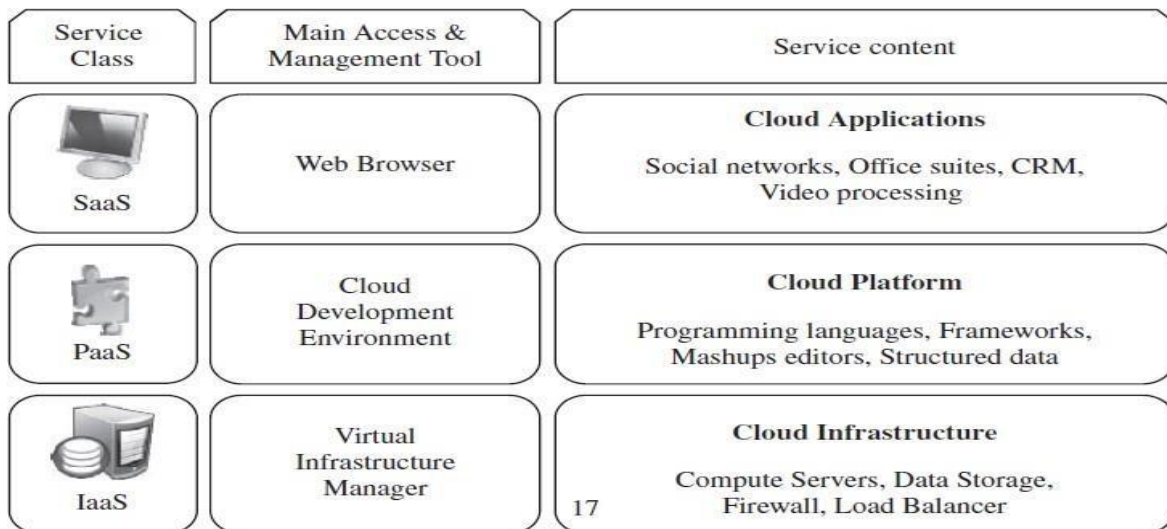


FIGURE 1.3. The cloud computing stack.

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.”
- A **community cloud** is “shared by several organizations and supports a specific community that has shared concerns
- 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”

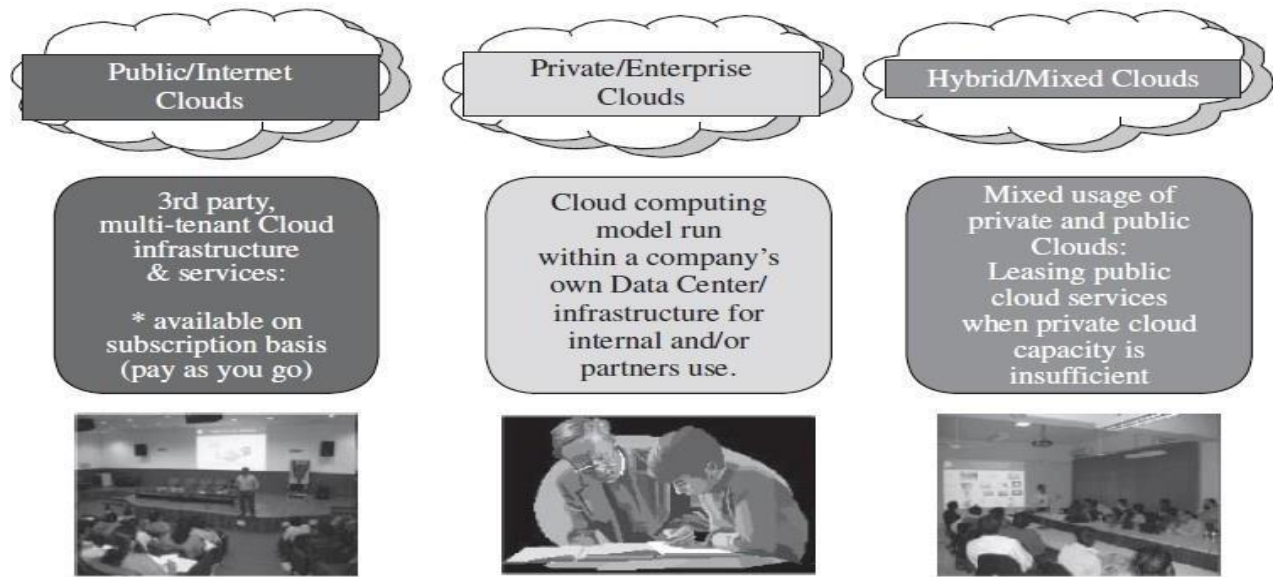


FIGURE 1.4. Types of clouds based on deployment models.

Features of a cloud

- are **essential** to enable services that truly represent the cloud computing model
- **Self-Service** : clouds must allow self-service access so that customers can request, customize, pay, and use services (expect on-demand, nearly instant access to resources) without intervention of human operators
- **Per-Usage Metering and Billing** : Services must be priced on a shortterm basis (e.g., by the hour), allowing users to release (and not pay for) resources as soon as they are not needed
- **Elasticity** : users expect clouds to rapidly provide resources in any quantity at any time. In particular, it is expected that the additional resources can be
 - (a) provisioned, possibly automatically, when an application load increases and
 - (b) released when load decreases (scale up and down)

Customization: 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 virtualservers

MIGRATING INTO THE CLOUD

Why Migrate?

- There are economic and business reasons why an enterprise application can be migrated into the cloud, and there are also a number of technological reasons.
- Initiatives in adoption of cloud technologies in the enterprise, resulting in integration of enterprise applications running off the captive data centers with the new ones that have been developed on the cloud.

Migration can happen at one of the **five levels** of

- application,
- code,
- design,
- architecture,
- usage

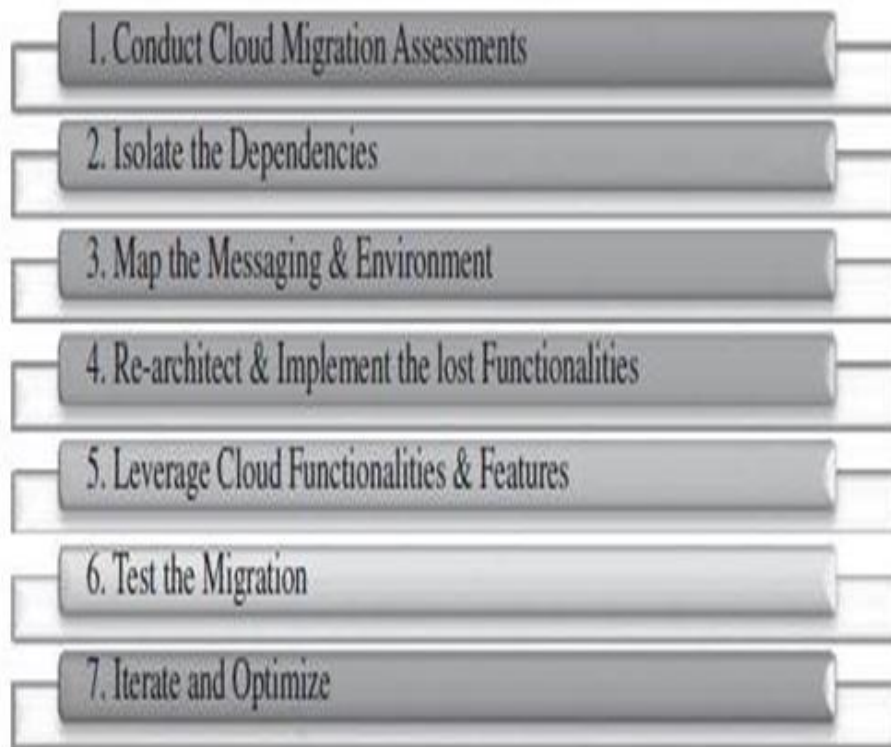
The migration of an enterprise application is best captured by the following

$$P \rightarrow P'_C + P'_I \rightarrow P'_{OFC} + P'_I$$

where

- P is the application before migration running in captive data center,
- P'_C is the application part after migration either into a (hybrid) cloud,
- P'_I is the part of application being run in the captive local data center, and
- P'_{OFC} is the application part optimized for cloud

SEVEN-STEP MODEL OF MIGRATION INTO A CLOUD



Iterative Step

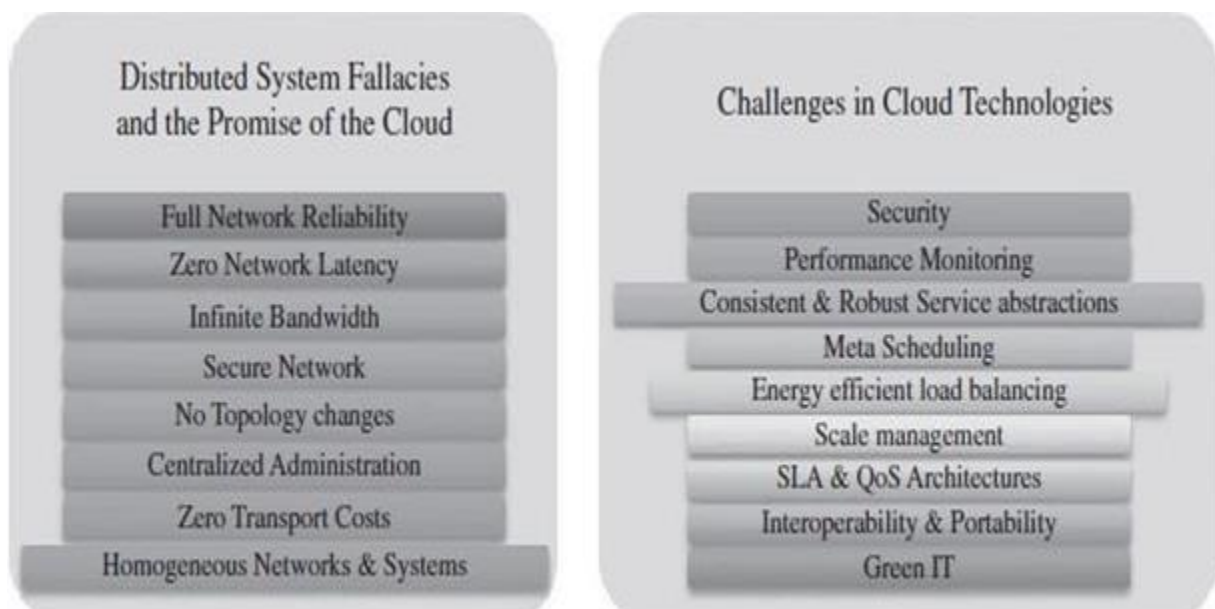


Migration Risks and Mitigation

- The biggest challenge to any cloud migration project is how effectively the migration risks are identified and mitigated.
- Migration risks for migrating into the cloud fall under two broad categories:
 - the general migration risks
 - the security-related migration risks
- several issues 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;

Challenges in the Cloud

- Security
- Costing model
- Charging model
- Service level agreement
- Cloud interoperability issue



- Use of cloud computing means dependence on others and that could possibly limit flexibility and innovation:
 - The others are likely become the bigger Internet companies like Google and IBM, who may monopolise the market.
 - Some argue that this use of supercomputers is a return to the time of mainframe computing that the PC was a reaction against.
- Security could prove to be a big issue:
 - It is still unclear how safe out-sourced data is and when using these services ownership of data is not always clear.
- There are also issues relating to policy and access:
 - If your data is stored abroad whose policy do you adhere to?
 - What happens if the remote server goes down?
 - How will you then access files?
 - There have been cases of users being locked out of accounts and losing access to data.

Enterprise cloud computing

Enterprise cloud computing is the alignment of a cloud computing model with an organization's business objectives (profit, return on investment, reduction of operations costs) and processes.

Cloud computing is composed of five essential characteristics:

- on-demand self-service
- broad network access
- resource pooling,
- rapid elasticity
- measured service

The ways in which these characteristics are manifested in an enterprise context vary according to the deployment model employed.

Deployment Models for Enterprise Cloud Computing

- **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
- **Private clouds** are built, operated, and managed by an organization for its internal use only to support its business operations exclusively
 - **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 PrivateCloud, OpSource Cloud and Skytap Virtual Lab
- **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
- **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
- **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 portability. Some examples of these offerings include Amazon Virtual Private Cloud, Skytap Virtual Lab, and CohesiveFT VPN-Cubed.

Adoption and Consumption Strategies

The selection of strategies for enterprise cloud computing is critical for IT capability as well as for the earnings and costs the organization experiences, motivating efforts toward convergence of business strategies and IT. Critical questions toward this convergence in the enterprise cloud paradigm

- Will an enterprise cloud strategy increase overall business value?

- Are the effort and risks associated with transitioning to an enterprise cloud strategy worth it?
- Which areas of business and IT capability should be considered for the enterprise cloud?
- Which cloud offerings are relevant for the purposes of an organization?
- How can the process of transitioning to an enterprise cloud strategy be piloted and systematically executed?

These questions are addressed from two strategic perspectives:

(1) Adoption (2) Consumption

Adoption strategy : an organization makes a decision to adopt a cloud computing model based on fundamental drivers for cloud computing—scalability, availability, cost and convenience

- **Scalability - Driven Strategy:** The objective is to support increasing workloads of the organization without investment and expenses exceeding returns. The conditions are that the effort, costs (CAPEX and OPEX) and time involved in accessing and installing IT capability on a CDC are less than going through a standard hardware and software procurement and licensing process
- **Availability - Driven Strategy:** Availability has close relations to scalability but is more concerned with the assurance that IT capabilities and functions are accessible, usable and acceptable by the standards of users. This is hence the objective of this basic enterprise cloud strategy.
- **Market-Driven Strategy:** This strategy is more attractive and viable for small, agile organizations that do not have (or wish to have) massive investments in their IT infrastructure. The objective here is to identify and acquire the “best deals” for IT capabilities as demand and supply change, enabling ongoing reductions in OPEX and CAPEX.
- **Convenience-Driven Strategy:** The objective is to reduce the load and need for dedicated system administrators and to make access to IT capabilities by users easier, regardless of their location and connectivity (e.g. over the Internet). The expectation is that the cost of obtaining IT capabilities from a CDC and making them accessible to users is significantly lower than the cost of having a dedicated administrator

Consumption Strategy:

The consumption strategies make a distinction between data and application logic because there are questions of programming models used, data sensitivity, software licensing and expected response times that need to be considered.

There are four consumptions strategies identified, where the differences in objectives, conditions and actions reflect the decision of an organization to trade-off hosting costs, controllability and resource elasticity of IT resources for software and data

- **Software Provision.** This strategy is relevant when the elasticity requirement is high for software and low for data, the controllability concerns are low for software and high for data, and the cost reduction concerns for software are high, while cost reduction is not a priority for data, given the high controllability concerns for data, that is, data are highly sensitive.
- **Storage Provision.** This strategy is relevant when the elasticity requirements is high for data and low for software, while the controllability of software is more critical than for data. This can be the case for data intensive applications, where the results from processing in the application are more critical and sensitive than the data itself. Furthermore, the cost reduction for data resources is a high concern, here as cost for software, given its criticality, is not an issue for the organization within reasonable means.
- **Solution Provision.** This strategy is relevant when the elasticity and cost reduction requirements are high for software and data, but the controllability requirements can be entrusted to the CDC. It is not the case that controllability is an insignificant requirement; it is rather the case that the organization trusts the CDC sufficiently to manage access and usage control of its software and data
- **Redundancy Services.** This strategy can be considered as a hybrid enterprise cloud strategy, where the organization switches between traditional, software, storage or solution management based on changes in its operational conditions and business demands

The strategy is referred to as the “redundancy strategy” because the CDC is used for situations such as disaster recovery, fail-over and load balancing Software, storage or solution services can be implemented using redundancy, such that users are redirected for the purpose of maintaining availability of functionality or performance/response times experienced by the user of the service.

UNIT-4

PLATFORM AS A SERVICE (PAAS)

Platform as a service (PaaS) is a cloud computing model where a third-party provider delivers hardware and software tools to users over the internet. Usually, these tools are needed for application development. A PaaS provider hosts the hardware and software on its own infrastructure. As a result, PaaS frees developers from having to install in-house hardware and software to develop or run a new application.

PaaS does not replace a company's entire IT infrastructure for software development. It is provided through a cloud service provider's hosted infrastructure. Users most frequently access the offerings through a web browser. PaaS can be delivered through public, private and hybrid clouds to deliver services such as application hosting and Java development.

Advantages of PAAS:

1. The PaaS provider will supply much of the infrastructure and other IT services, which users can access anywhere through a web browser. The ability to pay on a recurring (subscription) or per-use basis enables enterprises to eliminate the capital expenses they traditionally have for on-premises hardware and software.
2. Many PaaS products are geared toward software development. These platforms offer compute and storage infrastructures, as well as text editing, version management, compiling and testing services that help developers create new software quickly and efficiently.
3. PaaS architectures keep their underlying infrastructure hidden from developers and other users.

Disadvantages of PAAS:

1. service availability or resilience can be a concern with PaaS. If a provider experiences a service outage or other infrastructure disruption, this can adversely affect customers and result in costly lapses of productivity.
2. Vendor lock-in is another common concern because users cannot easily migrate many of the services and data from one PaaS platform to another competing PaaS platform.
3. Internal changes to a PaaS product are also a potential issue.

INFRASTRUCTURE AS A SERVICE (IAAS)

- Public Infrastructure as a Service providers commonly offer virtual servers containing one or more CPUs, OS, software stack, storage space and communication facilities
- 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,)
- (iv) choice of virtualization platform and operating systems;
- (v) different billing methods and period

Geographic Presence: To improve availability and responsiveness, a provider of worldwide services would typically build several data centers distributed around the world.

- 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.”
- **User Interfaces and Access to Servers.**

Ideally, 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.

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

Automatic Scaling and Load Balancing.

- As mentioned earlier in this chapter, elasticity is a key characteristic of the cloud computing model.
- Applications often need to scale up and down to meet varying load conditions.

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.
 - Amazon EC2 states that “if the annual uptime Percentage for a customer drops below 99.95% for the service year, that customer is eligible to receive a service credit equal to 10% of their bill.³”

Hypervisor and Operating System Choice:

- Traditionally, 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.

Public Cloud and Infrastructure Services

- Public cloud or external cloud describes cloud computing - resources are dynamically provisioned via publicly accessible Web applications/Web services (SOAP or RESTful interfaces) from an off-site third-party provider,
 - who shares resources and bills on a fine-grained utility computing basis,
 - the user pays only for the capacity of the provisioned resources at a particular time

Amazon Elastic Compute Cloud (EC2) is an IaaS service that provides elastic compute capacity in the cloud

Private Cloud and Infrastructure Services

- A **private cloud** aims at providing **public cloud functionality**, but **on private resources**, while maintaining control over an organization’s data and resources to meet security and governance’s requirements in an organization.
- Private clouds exhibit the following characteristics:
 - Allow service provisioning and compute capability for an organization’s users in a self-service manner.
 - Automate and provide well-managed virtualized environments.

- Optimize computing resources, and servers’ utilization.
- Support specific workloads.
- Examples are Eucalyptus and OpenNebula

“Hybrid cloud”

- in which a combination of private/internal and external cloud resources exist together by enabling outsourcing of noncritical services and functions in public cloud and keeping the critical ones internal
- Release resources from a public cloud and to handle sudden demand usage, which is called “cloud bursting

Cloud and Virtualization Standardization Efforts

- Standardization is important to ensure interoperability between
- virtualization management vendors,
- the virtual machines produced by each one of them,
- and cloud computing
- Distributed Management Task Force(DMTF)
- initiated the VMAN (Virtualization Management Initiative),
- delivers broadly supported interoperability and portability standards for managing the virtual computing lifecycle.

OVF (Open Virtualization Format

- VMAN’s OVF (Open Virtualization Format) in a collaboration between industry key players: Dell, HP, IBM, Microsoft, XenSource, and VMware.
- OVF specification provides a common format to package and securely distribute virtual appliances across multiple virtualization platforms.
- VMAN profiles define a consistent way of managing a heterogeneous virtualized environment
- Standardization effort has been initiated by Open Grid Forum (OGF) through organizing an official new working group to deliver a standard API for cloud IaaS, the Open Cloud Computing Interface Working Group (OCCIWG)

VIRTUAL MACHINES PROVISIONING

- Typical life cycle of VM and its major possible states of operation, which make the management and automation of VMs in virtual and cloud environments easier Process. Here, we describe the common and normal steps of provisioning a virtual server:
- 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).

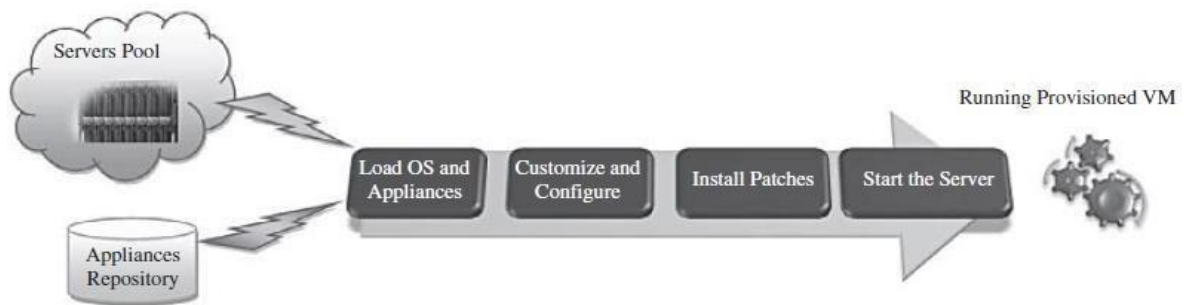


FIGURE 5.4. Virtual machine provision process.

- 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

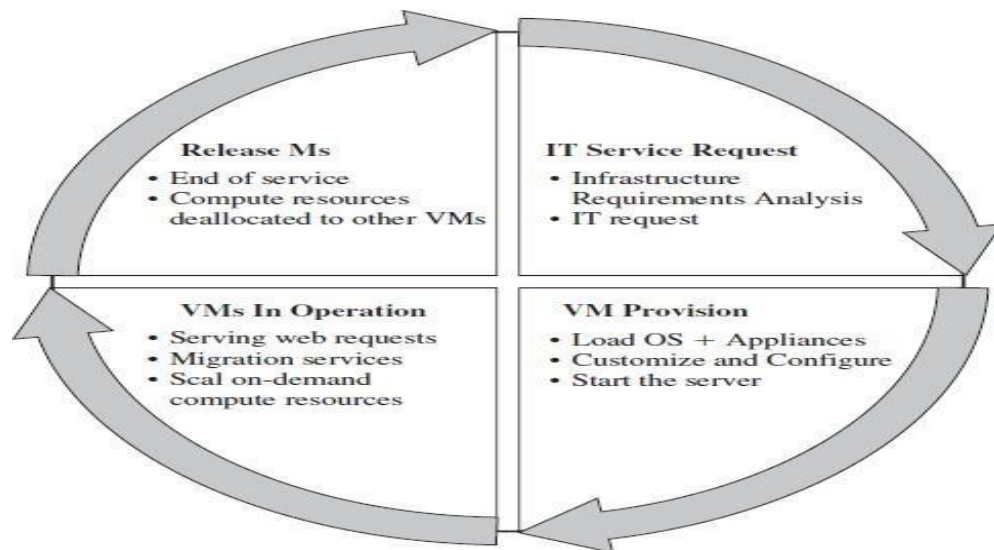


FIGURE 5.3. Virtual machine life cycle.

VIRTUAL MACHINE MIGRATION SERVICES

Migration service,

- in the context of virtual machines, is the process of moving a virtual machine from one host server or storage location to another
- There are different techniques of VM migration,
- hot/live migration,
- cold/regular migration, and
- live storage migration of a virtual machine

VM Migration, SLA and On-Demand Computing:

- virtual machines' migration plays an important role in data centers. once it has been detected that a particular VM is consuming more than its fair share of resources at the expense of other VMs on the same host, it will be eligible, for this machine, to either be moved to another underutilized host or assign more resources for it.
- There should be an integration between virtualization's management tools (with its migrations and performance's monitoring capabilities), and SLA's management tools to achieve balance in resources by migrating and monitoring the workloads, and accordingly, meeting the SLA

Migration of Virtual Machines to Alternate Platforms

- Advantages of having facility in data center's technologies is
- to have the **ability to migrate virtual machines** from one platform to another
- For example, the **VMware converter that handles migrations** between ESX hosts;
- the VMware server; and the VMware workstation.
- The VMware converter can also import from other virtualization platforms, such as Microsoft virtual server machines

Deployment Scenario:

- ConVirt deployment consists of at least one ConVirt workstation,
- where ConVirt is installed and ran, which provides the main console for managing the VM life cycle, managing images, provisioning new VMs, monitoring machine resources, and so on.
- There are two essential deployment scenarios for ConVirt:
- A **basic configuration** in which the Xen or KVM virtualization platform is on the local machine, where ConVirt is already installed; B,
- An **advanced configuration** in which the Xen or KVM is on one or more remote servers.

Installation. The installation process involves the following:

- Installing ConVirt on at least one computer.
- Preparing each managed server to be managed by ConVirt.
 - We have two managing servers with the following Ips: managed server 1, IP:172.16.2.22 and managed server 2, IP:172.16.2.25

Environment, Software, and Hardware. ConVirt 1.1, Linux Ubuntu 8.10, three machines, Dell core 2 due processor, 4G RAM.

- Adding Managed Servers and Provisioning VM.
- Once the installation is done and you are ready to manage your virtual infrastructure, then you can start the ConVirt management console :
- Select any of servers' pools existing (QA Lab in our scenario) and on its context menu, select "Add Server."

- You will be faced with a message asking about the virtualization platform you want to manage (Xen or KVM), as shown in Figure
- Choose KVM, and then enter the managed server information and credentials (IP, username, and password) as shown in Figure
- Once the server is synchronized and authenticated with the management console, it will appear in the left pane/of the ConVirt,

Live Migration Effect on a Running Web Server.

- Clark et al. did evaluate the above migration on an Apache 1.3 Web server; this served static content at a high rate, as illustrated in Figure 5.6.
- The throughput is achieved when continuously serving a single 512-kB file to a set of one hundred concurrent clients.
- This simple example demonstrates that a highly loaded server can be migrated with both controlled impact on live services and a short downtime

VMware Vmotion.

- This allows users to (a) automatically optimize and allocate an entire pool of resources for maximum hardware utilization, flexibility, and availability and (b) perform hardware's maintenance without scheduled downtime along with migrating virtual machines away from failing or underperforming servers

Citrix XenServerXenMotion.

- This is a nice feature of the Citrix XenServer product, inherited from the Xen live migrate utility, which provides the IT administrator with the facility to move a running VM from one XenServer to another in the same pool without interrupting the service

Regular/Cold Migration.

Cold migration is the migration of a powered-off virtual machine.

- Main **differences between live migration and cold migration** are that
- 1) **live migration needs a shared storage** for virtual machines in the server's pool, but cold migration does not;
- 2) live migration for a virtual machine between two hosts, there would be **certain CPU compatibility checks to be applied**; while in cold migration this checks do not apply
- The cold migration process (VMware) can be summarized as follows:

- The **configuration files**, including the NVRAM file (BIOS settings), log files, as well as the disks of the virtual machine, are **moved** from the source host to the destination host's associated storage area.
- The virtual machine is **registered** with the new host.
- After the migration is completed, the **old version** of the virtual machine is **deleted** from the source host.

Live Storage Migration of Virtual Machine.

- This kind of migration constitutes moving the virtual disks or configuration file of a running virtual machine to a new data store without any interruption in the availability of the virtual machine's service

Aneka

- Manjrasoft Aneka is a .NET-based platform and framework designed for building and deploying distributed applications on clouds.
- It provides a set of APIs for transparently exploiting distributed resources and expressing the business logic of applications by using the preferred programming abstractions.
- Aneka also provides support for deploying and managing clouds.
- By using its Management Studio and a set of Web interfaces, it is possible to set up either public or private clouds, monitor their status, update their configuration, and perform the basic management operations.

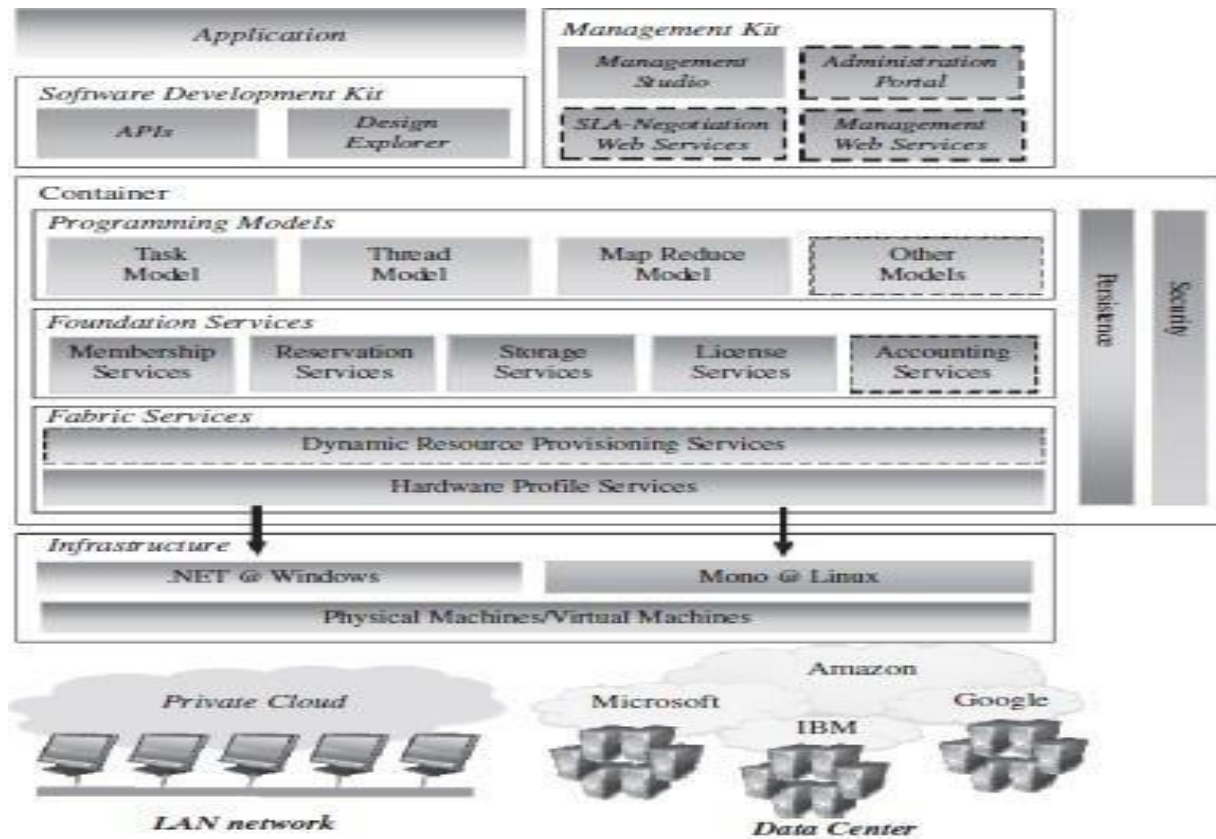


FIGURE 5.22. Manjrasoft Aneka layered architecture [10].

On the Management of Virtual machines for Cloud Infrastructures

There is a growing ecosystem of technologies and tools to build *private clouds*—where in-house resources are virtualized, and internal users can request and manage these resources using interfaces similar or equal to those of public clouds—and *hybrid clouds*—where an organization's private cloud can supplement its capacity using a public cloud. This concept focuses on IaaS clouds and 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; and (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").

Virtualization technologies have been the key enabler of many of these salient characteristics

of IaaS clouds by giving providers a more flexible and generic way of managing their resources. Thus, *virtual infrastructure (VI) management*—the management of virtual machines distributed across a pool of physical resources—becomes a key concern when building an IaaS cloud and poses a number of challenges.

- The first problem is how to manage the virtual infrastructures themselves. Although resource management has been extensively studied, particularly for job management in high-performance computing, managing VMs poses additional problems that do not arise when managing jobs, such as the need to set up custom software environments for VMs, setting up and managing networking for interrelated VMs, and reducing the various overheads involved in using VMs. Thus, VI managers must be able to efficiently orchestrate all these different tasks.
- A particularly interesting problem when provisioning virtual infrastructures is how to deal with situations where the demand for resources is known beforehand.
- IaaS clouds can be used to deploy *services* that will be consumed by users other than the one that deployed the services.

Managing VMs in a pool of distributed physical resources is a key concern in IaaS clouds, requiring the use of a virtual infrastructure manager. To address some of the shortcomings in existing VI solutions, we have developed the open source OpenNebula[13] virtual infrastructure engine. OpenNebula is capable of managing groups of interconnected VMs—with support for the Xen, KVM, and VMWare platforms—within data centers and private clouds that involve a large amount of virtual and physical servers. OpenNebula can also be used to build hybrid clouds by interfacing with remote cloud sites.

UNIT-5

SAAS

- SaaS is a model of software deployment where an application is hosted as a service provided to customers across the Internet.
- SaaS alleviates the burden of software maintenance/support but users relinquish control over software versions and requirements

SaaS Maturity Model

Level 1: Ad-Hoc/Custom – One Instance per customer

Level 2: Configurable per customer

Level 3: configurable & Multi-Tenant-Efficient

Level 4: Scalable, Configurable & Multi-Tenant-Efficient

SaaS INTEGRATION PRODUCTS AND PLATFORMS

- Cloud-centric integration solutions are being developed and demonstrated for showcasing their capabilities for integrating enterprise and cloud applications.
- Composition and collaboration will become critical and crucial for the mass adoption of clouds

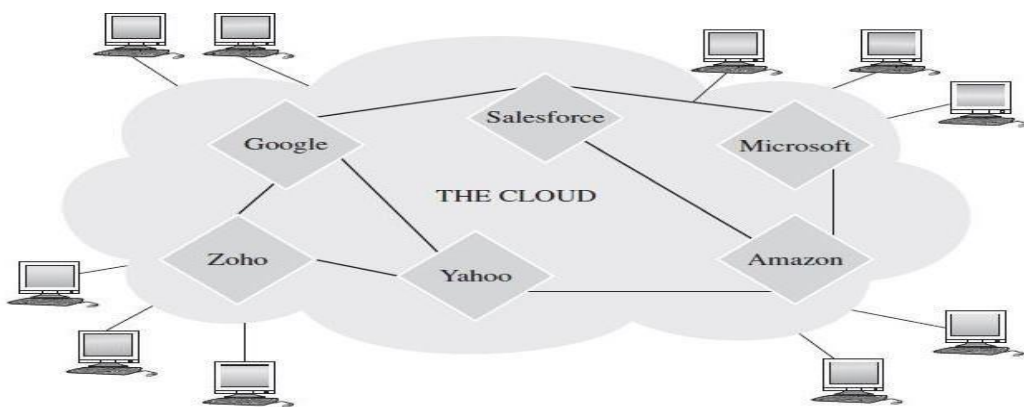


FIGURE 3.4. The Smooth and Spontaneous Cloud Interaction via Open Clouds.

Jitterbit:

Jitterbit is a fully graphical integration solution that provides users a versatile platform suite of productivity tools to reduce the integration efforts sharply. Jitterbit can be used standalone or with existing EAI infrastructures, Help us quickly design, implement, test, deploy, and manage the

integration projects Two major components :

- Jitterbit Integration Environment: An intuitive point-and-click graphical UI that enables to quickly configure, test, deploy and manage integration projects on the Jitterbit server.
- Jitterbit Integration Server: A powerful and scalable run-time engine that processes all the integration operations, fully configurable and manageable from the Jitterbit application.

Linkage with on premise and on demand Applications

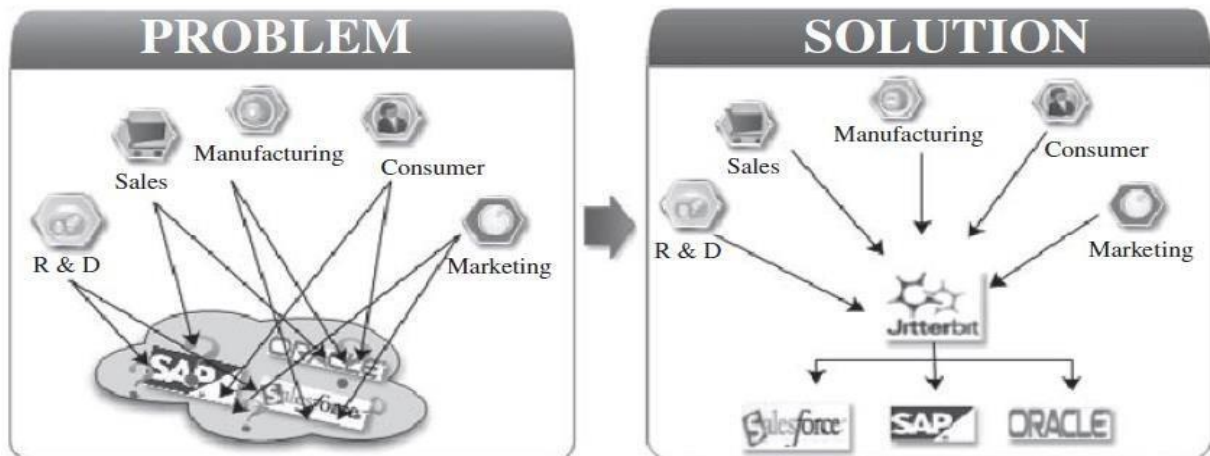


FIGURE 3.5. Linkage of On-Premise with Online and On-Demand Applications.

Google APP Engine

- The app engine is a Cloud-based platform, is quite comprehensive and combines infrastructure as a service (IaaS), platform as a service (PaaS) and software as a service (SaaS).
- The app engine supports the delivery, testing and development of software on demand in a Cloud computing environment that supports millions of users and is highly scalable.
- The company extends its platform and infrastructure to the Cloud through its app engine. It presents the platform to those who want to develop SaaS solutions at competitive costs.

Google is a leader in web-based applications and also offers cloud development services.

- These services come in the form of the Google App Engine, which enables developers to build their own web applications utilizing the same infrastructure that powers Google's powerful applications.

- The Google App Engine provides a fully integrated application environment. Using Google's development tools and computing cloud, App Engine applications are easy to build, easy to maintain, and easy to scale. All you have to do

Features of App Engine

- These are covered by the depreciation policy and the service-level agreement of the app engine. Any changes made to such a feature are backward-compatible and implementation of such a feature is usually stable. These include data storage, retrieval, and search; communications; process management; computation; app configuration and management.
- Data storage, retrieval, and search include features such as HRD migration tool, Google Cloud SQL, logs, datastore, dedicated Memcache, blobstore, Memcache and search.
- Communications include features such as XMPP. channel, URL fetch, mail, and Google Cloud Endpoints.
- Process management includes features like scheduled tasks and task queue Computation includes images.
- App management and configuration cover app identity, users, capabilities, traffic splitting, modules, SSL for custom domains, modules, remote access, and multitenancy

Centralizing email Communications

- The key here is to enable anywhere/anytime access to email.
- Precloud computing, your email access was via a single computer, which also stored all your email messages. For this purpose, you probably used a program like Microsoft Outlook or Outlook Express, installed on your home computer.
- To check your home email from work, it took a bit of juggling and perhaps the use of your ISP's email access web page. That web page was never in sync with the messages on your home PC, of course, which is just the start of the problems with trying to communicate in this fashion.
- A better approach is to use a web-based email service, such as Google's Gmail (mail.google.com), Microsoft's Windows Live Hotmail (mail.live.com), or Yahoo! Mail (mail.yahoo.com). These services place your email inbox in the cloud; you can access it from any computer connected to the Internet.

Collaborating via Web-Based Communication Tools

GMAIL

- Gmail offers a few unique features that set it apart from the web-based email crowd.
- First, Gmail doesn't use folders. With Gmail you can't organize your mail into folders, as

you can with the other services.

- Instead, Gmail pushes the search paradigm as the way to find the messages you want— not a surprise, given Google’s search-centric business model.
- Gmail does, however, let you “tag” each message with one or more labels. This has the effect of creating virtual folders, as you can search and sort your messages by any of their labels.
- In addition, Gmail groups together related email messages in what Google calls conversations

Yahoo! Mail Yahoo! Mail (mail.yahoo.com)

- is another web mail service, provided by the popular Yahoo! search site.
- The basic Yahoo! Mail is free and can be accessed from any PC, using any web browser.
- Yahoo! also offers a paid service called Yahoo! Mail Plus that lets you send larger messages and offers offline access to your messages via POP email clients

Web Mail Services

- AOL Mail (mail.aol.com)
- BigString (www.bigstring.com) E
- xcite Mail (mail.excite.com)
- FlashMail (www.flashmail.com)
- GMX Mail (www.gmx.com)
- Inbox.com (www.inbox.com)
- Lycos Mail (mail.lycos.com)
- Mail.com (www.mail.com)
- Zoho Mail (zoho.mail.com)

Data Security

- Information in a cloud environment has much more dynamism and fluidity than information that is static on a desktop or in a network folder

- 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
- The idea of content-centric or information-centric protection, being an inherent part of a data object is a development out of the idea of the “de-perimeterization” of the enterprise.
- This idea was put forward by a group of Chief Information Officers (CIOs) who formed an organization called the Jericho Forum
- Cloud data security is the combination of technology solutions, policies, and procedures that you implement to protect cloud-based applications and systems, along with the associated data and user access.
- The core principles of information security and data governance—data confidentiality, integrity, and availability (known as the CIA triad)—also apply to the cloud.
- When it comes to data, the cloud poses a variety of risks that need to be addressed as part of security strategy. The biggest risks—as increasingly rely on the cloud for collecting, storing, and processing critical data—are cyberattacks and data breaches.
- Some of the common cloud-related risks that organizations face include:
 1. Regulatory non-compliance
 2. Data loss and Data leaks
 3. Loss of customer trust and brand reputation
 4. business interruption
 5. financial issues

Current state of data security in Cloud:

Data privacy and security are placed under a lot of scrutiny these days thanks to the numerous information leaks that became public knowledge in recent years. The independent study from Sophos of 3,521 IT managers leveraging the public cloud across 26 countries and six continents reveals many insights into the world of cloud security, with some of the key takeaways being:

- “[70%] of organizations reported they were hit by malware, ransomware, data theft, account compromise attempts, or cryptojacking in the last year.”
- Data loss/leakage is the topmost concern with 44% of organizations reporting “data loss as

one of their top three focus areas.”

- Multi-cloud organizations report more security incidents than single platform organizations.
- Europe’s General Data Protection Regulation (GDPR) may be partly responsible for European organizations seeing the “lowest attack rates of all regions.”
- 75% of organizations do not see staff expertise as a top priority despite the prevalence of cyberattacks.
- 66% of attacks were exploitations of misconfigurations and 33% used stolen credentials to gain access.

CLOUD COMPUTING AND IDENTITY

Digital identity

- holds the key to flexible data security within a cloud Environment
- A digital identity represents who we are and how we interact with others on-line.
- **Access, identity, and risk** are three variables that 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 of the 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.

Identity, Reputation, and Trust

- Reputation is a real-world commodity; that is a basic requirement of human-to-human relationships
- Our basic societal communication structure is built upon the idea of reputation and trust.
- Reputation and its counter value, trust, is easily transferable to a digital realm:
 - eBay, for example, having partly built a successful business model on the strength of a ratings system, builds up the reputation of its buyers and sellers through successful (or unsuccessful) transactions.

- These types of reputation systems can be extremely useful when used with a digital identity.
- They can be used to associate varying levels of trust with that identity, which in turn can be used to define the level (granular variations) of security policy applied to data resources that the individuals wishes to access.

User-Centric Identity:

- Digital identities are a mechanism for identifying an individual, particularly within a cloud environment ; identity ownership being placed upon the individual is known as user-centric identity
- It allows users to consent and control how their identity (and the individual identifiers making up the identity, the claims) is used.
- This reversal of ownership away from centrally managed identity platforms(enterprise-centric) has many advantages.
- This includes the potential to improve the privacy aspects of a digital identity, by giving an individual the ability to apply permission policies based on their identity and to control which aspects of that identity are divulged
- An identity may be controllable by the end user, to the extent that the user can then decide what information is given to the party relying on the identity

Information Card:

- Information cards permit a user to present to a Web site or other service (relying party) one or more claims, in the form of a software token, which may be used to uniquely identify that user.
- They can be used in place of user name/ passwords, digital certificates, and other identification systems, when user identity needs to be established to control access to a Web site or other resource, or to permit digital signing

Information cards are part of an identity meta-system consisting of:

1. **Identity providers (IdP)**, who provision and manage information cards,with specific claims, to users.
2. **Users** who own and utilize the cards to gain access to Web sites and other resources that support information cards.
3. **An identity selector/service**, which is a piece of software on the user's desktop or in

thecloud that allows a user to select and manage their cards.

4. **Relying parties.** These are the applications, services, and so on, that can use an information card to authenticate a person and to then authorize an action such as logging onto a Web site, accessing a document, signing content, and so on

Each information card is associated with a set of claims which can be used to identify the user.

These claims include identifiers such as name, email address, post code

Using Information Cards to Protect Data

- Information cards are built around a set of open standards devised by a consortium that includes Microsoft, IBM, Novell, and so on.
- The original remit of the cards was to create a type of single sign on system for the Internet, to help users to move away from the need to remember multiple passwords.
- However, the information card system can be used in many more ways.
- Because an information card is a type of digital identity, it can be used in the same way that other digital identities can be used.

For example, an information card can be used to digitally sign data and content and to control access to data and content. One of the more sophisticated uses of an information card is the advantage given to the cards by way of the claims system.

Cloud Computing and Data Security Risk

- 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

Data security risks are compounded by the open nature of cloud computing.

- Access control becomes a much more fundamental issue in cloud-based systems because of the accessibility of the data
- Information-centric access control (as opposed to access control lists) can help to balance improved accessibility with risk, by associating access rules with different data objects within an open and accessible platform, without losing the Inherent usability of that platform
- A further area of risk associated not only with cloud computing, but also with traditional network computing, is the use of content after access.
- The risk is potentially higher in a cloud network, for the simple reason that the information is outside of your corporate walls.

Data-centric mashups are those

- that are used to perform business processes around data creation and dissemination—by their very nature, can be used to hijack data, leaking sensitive information and/or affecting integrity of that data
- Cloud computing, more than any other form of digital communication technology, has created a need to ensure that protection is applied at the inception of the information, in a content centric manner, ensuring that a security policy becomes an integral part of that data throughout its life cycle.

Encryption

- is a vital component of the protection policy, but further controls over the access of that data and on the use of the data must be met.
- In the case of mashups, the controlling of access to data resources, can help to alleviate the security concerns by ensuring that mashup access is authenticated.
- Linking security policies, as applied to the use of content, to the access control method

offer a way of continuing protection of data, post access and throughout the life cycle; this type of data security philosophy must be incorporated into the use of cloud computing to alleviate security risks.