MODULE-1

1a. Explain the cloud computing reference model with a neat diagram.

Ans: A fundamental characteristic of cloud computing is the capability to deliver, on demand, a variety of IT services that are quite diverse from each other. This variety creates different perceptions of what cloud computing is among users. Despite this lack of uniformity, it is possible to classify cloud computing services offerings into three major categories: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). These categories are related to each other as described in Figure 1.5, which provides an organic view of cloud computing.

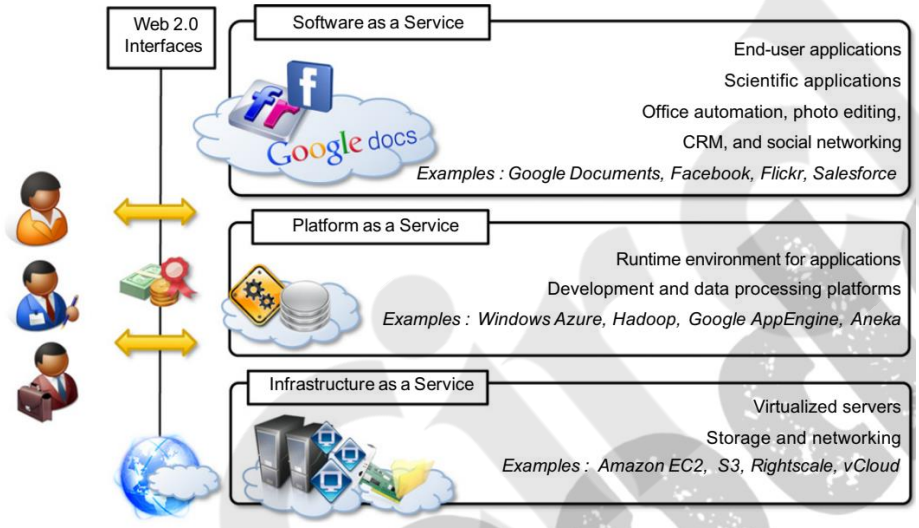


FIGURE 1.5
The Cloud Computing Reference Model.

Infrastructure-as-a-Service solutions deliver infrastructure on demand in the form of virtual hardware, storage, and networking. Virtual hardware is utilized to provide compute on demand in the form of virtual machine instances. These are created at users' request on the provider's infrastructure, and users are given tools and interfaces to configure the software stack installed in the virtual machine. The pricing model is usually defined in terms of dollars per hour, where the hourly cost is influenced by the characteristics of the virtual hardware. Virtual storage is delivered in the form of raw disk space or object store.

Platform-as-a-Service solutions are the next step in the stack. They deliver scalable and elastic runtime environments on demand and host the execution of applications. These services are backed by a core middleware platform that is responsible for creating the abstract environment where applications are deployed and executed. It is the responsibility of the service provider to provide scalability and to manage fault tolerance, while users are requested to focus on the logic of the application developed by leveraging the provider's APIs and libraries.

Software-as-a-Service solutions provide applications and services on demand. Most of the common functionalities of desktop applications such as office automation, document management, photo editing and customer relationship management (CRM) software are replicated on the provider's infrastructure and made more scalable and accessible through a browser on demand. These applications are shared across multiple users whose interaction is isolated from the other users.

1b. Explain the differences between public, private and hybrid cloud deployment models.

Ans: The cloud deployment model identifies the specific type of cloud environment based on ownership, scale, and access, as well as the cloud's nature and purpose. The location of the servers you're utilizing and who controls them are defined by a cloud deployment model.

Public Cloud

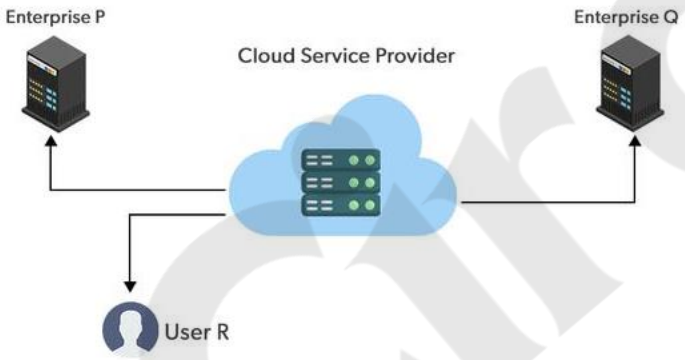


Fig: Public Cloud

The public cloud makes it possible for anybody to access systems and services. The public cloud may be less secure as it is open to everyone. The public cloud is one in which cloud infrastructure services are provided over the internet to the general people or major industry groups. The infrastructure in this cloud model is owned by the entity that delivers the cloud services, not by the consumer. It is a type of cloud hosting that allows customers and users to easily access systems and services.

Private Cloud:

The private cloud deployment model is the exact opposite of the public cloud deployment model. It's a one-on-one environment for a single user (customer). There is no need to share your hardware with anyone else. The cloud platform is implemented in a cloud-based secure environment that is protected by powerful firewalls and under the supervision of an organization's IT department. The private cloud gives greater flexibility of control over cloud resources.

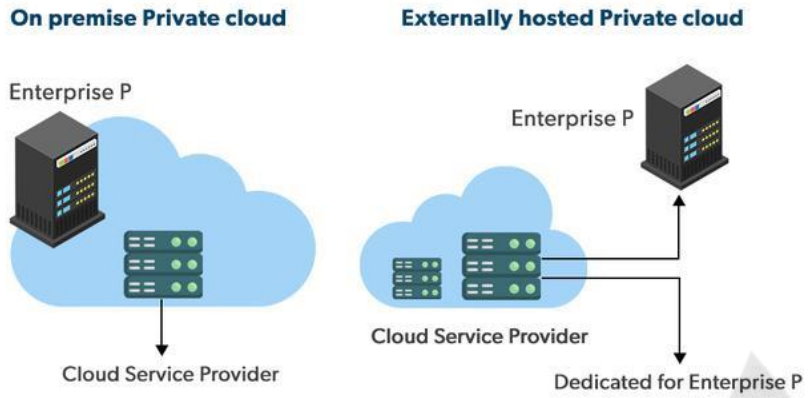


fig: Private Cloud

Advantages of the Private Cloud Model

- **Better Control:** You are the sole owner of the property. You gain complete command over service integration, IT operations, policies, and user behaviour.
- **Data Security and Privacy:** It's suitable for storing corporate information to which only authorized staff have access. By segmenting resources within the same infrastructure, improved access and security can be achieved.
- **Supports Legacy Systems:** This approach is designed to work with legacy systems that are unable to access the public cloud.
- **Customization:** Unlike a public cloud deployment, a private cloud allows a company to tailor its solution to meet its specific needs.

Disadvantages of the Private Cloud Model

- **Less scalable:** Private clouds are scaled within a certain range as there is a smaller number of clients.
- **Costly:** Private clouds are more costly as they provide personalized facilities.

Hybrid Cloud:

By bridging the public and private worlds with a layer of proprietary software, hybrid cloud computing gives the best of both worlds. With a hybrid solution, you may host the app in a safe environment while taking advantage of the public cloud's cost savings. Organizations can move data and applications between different clouds using a combination of two or more cloud deployment methods, depending on their needs.

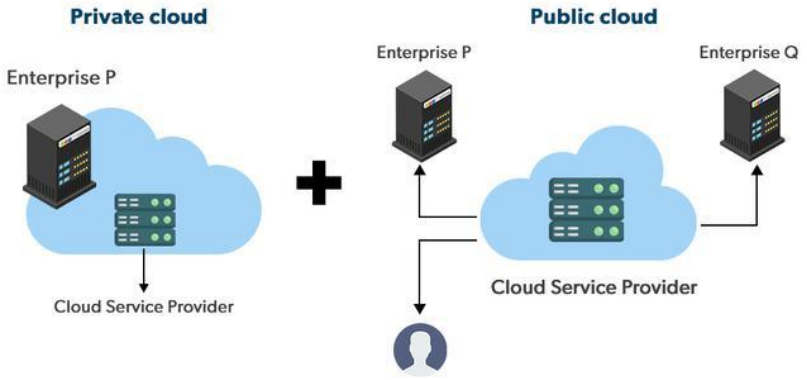


fig: Hybrid Cloud

Advantages of the Hybrid Cloud Model

- **Flexibility and control:** Businesses with more flexibility can design personalized solutions that meet their particular needs.
- **Cost:** Because public clouds provide scalability, you'll only be responsible for paying for the extra capacity if you require it.
- **Security:** Because data is properly separated, the chances of data theft by attackers are considerably reduced.

Disadvantages of the Hybrid Cloud Model

- **Difficult to manage:** Hybrid clouds are difficult to manage as it is a combination of both public and private cloud. So, it is complex.
- **Slow data transmission:** Data transmission in the hybrid cloud takes place through the public cloud so latency occurs.

1c. Elaborate the various cloud computing characteristics and its benefits.

Ans: Cloud computing has some interesting characteristics that bring benefits to both cloud service consumers (CSCs) and cloud service providers (CSPs). These characteristics are:

- **No up-front commitments:** Cloud computing allows consumers to access services without requiring substantial up-front investments in hardware or infrastructure. Traditional IT setups often involve purchasing servers, software licenses, and other physical resources, which can be expensive and have long-term commitments.
- **On-demand access:** One of the key advantages of cloud computing is the ability to access resources (such as computing power, storage, and software applications) on demand, at any time. This means that consumers can quickly provision or de-provision resources without waiting for manual configuration or deployment.
- **Nice pricing:** Cloud services typically follow flexible pricing models such as pay-per-use, subscription-based, or tiered pricing. This provides significant cost savings for consumers, as they

only pay for the services they actually consume. This contrasts with traditional IT infrastructure, where organizations have to purchase and maintain expensive hardware, often underutilized.

- **Simplified application acceleration and scalability:** Cloud computing enables rapid deployment and scaling of applications, helping businesses accelerate time-to-market for their products and services. With cloud services, businesses can deploy applications across a global network of data centres, ensuring low latency and faster access for end users.
- **Efficient resource allocation:** Cloud service providers have the ability to allocate resources (such as storage, computing power, and bandwidth) dynamically and efficiently across many customers. By pooling resources in a virtualized environment, CSPs can ensure that infrastructure is fully utilized while offering consumers the flexibility to scale resources based on their needs.
- **Energy efficiency:** Cloud computing data centres often operate on a much larger scale than traditional IT infrastructure. CSPs invest in energy-efficient technologies and optimized cooling methods, which reduce the energy consumption per unit of computation. By leveraging shared resources, cloud data centres achieve higher resource density, resulting in more efficient energy use.
- **Seamless creation and use of third-party services:** Cloud platforms make it easier for consumers to integrate third-party services such as APIs, software tools, or even entire applications into their own systems. Cloud ecosystems often provide access to marketplaces or pre-configured environments where consumers can quickly add services like payment processors, analytics tools, or content delivery networks (CDNs).

2a. List & Explain the various cloud computing platforms and technologies.

Ans: Development of a cloud computing application happens by leveraging platforms and frameworks that provide different types of services, from the bare-metal infrastructure to customizable applications serving specific purposes.

- **Amazon web services (AWS):** AWS offers comprehensive cloud IaaS services ranging from virtual compute, storage, and networking to complete computing stacks. AWS is mostly known for its compute and storage-on-demand services, namely Elastic Compute Cloud (EC2) and Simple Storage Service (S3). EC2 provides users with customizable virtual hardware that can be used as the base infrastructure for deploying computing systems on the cloud. It is possible to choose from a large variety of virtual hardware configurations, including GPU and cluster instances. EC2 instances are deployed either by using the AWS console, which is a comprehensive Web portal for accessing AWS services, or by using the Web services API available for several programming languages.
- **Google AppEngine:** Google AppEngine is a scalable runtime environment mostly devoted to executing Web applications. These take advantage of the large computing infrastructure of Google

to dynamically scale as the demand varies over time. AppEngine provides both a secure execution environment and a collection of services that simplify the development of scalable and high-performance Web applications. These services include in-memory caching, scalable data store, job queues, messaging, and corn tasks. Developers can build and test applications on their own machines using the AppEngine software development kit (SDK), which replicates the production runtime environment and help test and profile applications.

- **Microsoft Azure:** Microsoft Azure is a cloud operating system and a platform for developing applications in the cloud. It provides a scalable runtime environment for Web applications and distributed applications in general. Applications in Azure are organized around the concept of roles, which identify a distribution unit for applications and embody the application's logic. Currently, there are three types of roles: Web role, worker role, and virtual machine role. The Web role is designed to host a Web application, the worker role is a more generic container of applications and can be used to perform workload processing, and the virtual machine role provides a virtual environment in which the computing stack can be fully customized, including the operating systems.
- **Hadoop:** Apache Hadoop is an open-source framework that is suited for processing large data sets on commodity hardware. Hadoop is an implementation of MapReduce, an application programming model developed by Google, which provides two fundamental operations for data processing: map and reduce. The former transforms and synthesizes the input data provided by the user; the latter aggregates the output obtained by the map operations. Hadoop provides the runtime environment, and developers need only provide the input data and specify the map and reduce functions that need to be executed.
- **Force.com and Salesforce.com:** Force.com is a cloud computing platform for developing social enterprise applications. The platform is the basis for SalesForce.com, a Software-as-a-Service solution for customer relationship management. Force.com allows developers to create applications by composing ready-to-use blocks; a complete set of components supporting all the activities of an enterprise are available. It is also possible to develop your own components or integrate those available in AppExchange into your applications.
- **Manjrasoft Aneka:** Manjrasoft Aneka is a cloud application platform for rapid creation of scalable applications and their deployment on various types of clouds in a seamless and elastic manner. It supports a collection of programming abstractions for developing applications and a distributed runtime environment that can be deployed on heterogeneous hardware (clusters, networked desktop computers, and cloud resources). Developers can choose different abstractions to design their application: tasks, distributed threads, and map-reduce.

2b. What are the major distributed computing technologies that led to cloud computing.

Ans: The three major distributed computing technologies that led to cloud computing are as follows:

- **Mainframes.** These were the first examples of large computational facilities leveraging multiple processing units. Mainframes were powerful, highly reliable computers specialized for large data movement and massive input/output (I/O) operations. They were mostly used by large organizations for bulk data processing tasks such as online transactions, enterprise resource planning, and other operations involving the processing of significant amounts of data.
- **Clusters:** Cluster computing started as a low-cost alternative to the use of mainframes and supercomputers. The technology advancement that created faster and more powerful mainframes and supercomputers eventually generated an increased availability of cheap commodity machines as a side effect. These machines could then be connected by high-bandwidth network and controlled by specific software tools that manage them as a single system. Starting in the 1980s, clusters become the standard technology for parallel and high-performance computing.
- Grids. Grid computing appeared in the early 1990s as an evolution of cluster computing. In an analogy to the power grid, grid computing proposed a new approach to access large computational power, huge storage facilities, and a variety of services. Users can "consume" resources in the same way as they use other utilities such as power, gas, and water. Grids initially developed as aggregations of geographically dispersed clusters by means of Internet connections. These clusters belonged to different organizations, and arrangements were made among them to share the computational power.

2c. Describe the main characteristics of a service-oriented computing.

Ans: The main characteristics od a service-oriented computing are as follows:

- **Core Reference Model for Cloud Computing:** Service-oriented computing (SOC) forms the core reference model for cloud computing systems. In cloud environments, services are the fundamental building blocks. These services can range from simple functions to complex business processes, and they are designed to be loosely coupled, reusable, platform-independent, and location-transparent.
- **Describing and Platform-Agnostic Services:** A service in SOC is an abstraction that represents a self-describing component capable of performing a function. This service can be executed on any platform because it is platform-agnostic. The service exposes its functionality through a network-accessible protocol, which allows any client to interact with it, regardless of the technology stack used by the service or the client.
- **Loose Coupling and Reusability:** One of the key principles of SOC is loose coupling. Services are loosely coupled, meaning they interact with each other without tightly binding their

implementations. This promotes reusability, as services can be integrated into various systems or applications without needing to be re-engineered for each use case.

- **Platform Independence and Accessibility:** SOC promotes platform independence, allowing services to be accessed from different environments. This feature increases the accessibility of services, enabling them to be consumed by a broad range of clients. Services can be found in global service registries and consumed in a location-transparent manner, meaning clients do not need to be concerned about where a service is hosted or its underlying platform.
- **Quality of Service (QoS):** In SOC, Quality of Service (QoS) refers to a set of functional and non-functional attributes used to evaluate and measure a service's performance. These attributes can include response time, security (such as encryption and authentication), reliability, scalability, transactional integrity, and availability. QoS requirements are agreed upon between the service provider and the consumer via a Service Level Agreement (SLA), which defines the acceptable levels of service performance.
- **Software-as-a-Service (SaaS):** SOC introduces the concept of Software-as-a-Service (SaaS), which provides software applications over the internet through a subscription model. SaaS leverages multitenancy, where multiple clients share the same application infrastructure, achieving economies of scale. The SaaS provider maintains the application, infrastructure, and upgrades, freeing the client from costly maintenance and complex upgrades.
- **Benefits of SOC in Cloud Computing:** SOC enables cloud computing to deliver complex business processes and transactions as individual services. Applications can be composed dynamically from existing services, and services can be reused across different systems. This composability allows businesses to build customized solutions by assembling services on demand, resulting in significant cost savings and flexibility.

MODULE-2

3a. Explain the characteristics of virtualized environments.

Ans: Virtualization refers to the process of creating a virtual version of something, whether it be hardware, software environments, storage, or networks. In a virtualized environment, multiple virtual instances of resources can be created from a single physical resource, allowing for more efficient usage, isolation, and flexibility. The concept of virtualization can be applied to different domains, such as hardware, storage, and networking, and it facilitates the creation of virtual resources that function independently, often indistinguishable from the original physical ones. In a virtualized environment, there are three primary components:

- **Guest**: The guest refers to the virtualized system or environment that interacts with the virtualization layer, rather than directly with the physical host. It could be a virtual machine (VM), a virtual storage system, or a virtual network.
- **Host**: The host is the physical environment or the underlying system where the virtualization takes place. In hardware virtualization, this is typically the physical server or machine that hosts the virtual instances (VMs).
- **Virtualization Layer**: The virtualization layer (also called the hypervisor or virtual machine manager) is responsible for managing and creating the virtual environments. It sits between the host and the guest, controlling the distribution of resources and ensuring that the virtual machines or virtual environments function independently of the underlying hardware.

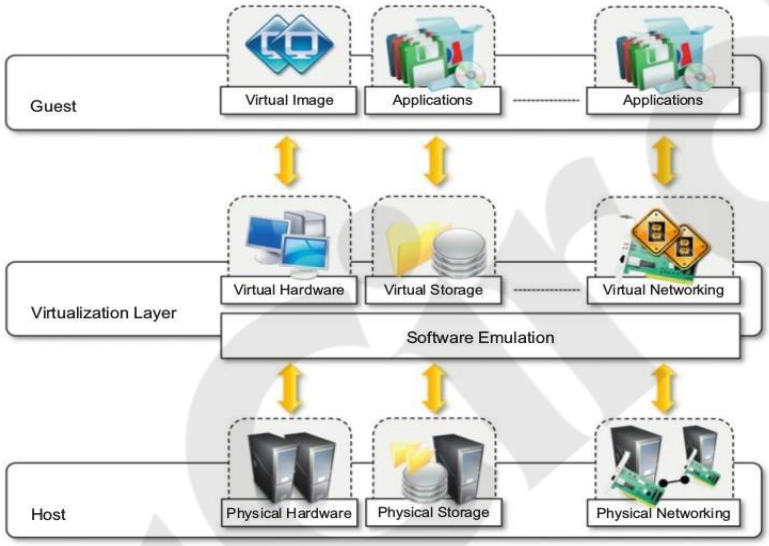


FIGURE 3.1
The virtualization reference model.

The main common characteristic of all these different implementations is the fact that the virtual environment is created by means of a software program. The ability to use software to emulate such a wide variety of environments creates a lot of opportunities, previously less attractive because of excessive overhead introduced by the virtualization layer.

- **Increased security:** The ability to control the execution of a guest in a completely transparent manner opens new possibilities for delivering a secure, controlled execution environment. The virtual machine represents an emulated environment in which the guest is executed. All the operations of the guest are generally performed against the virtual machine, which then translates and applies them to the host.

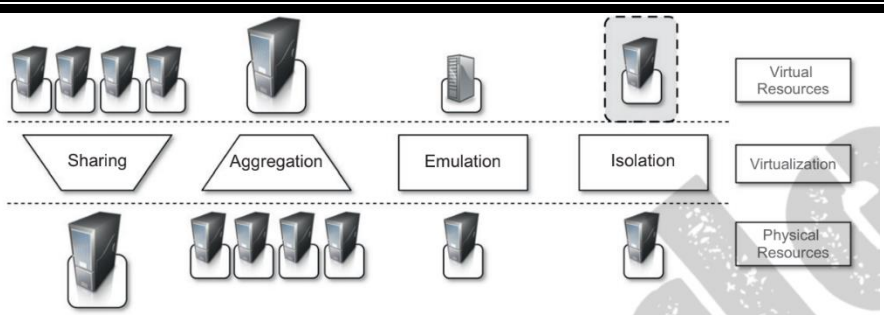


FIGURE 3.2
Functions enabled by managed execution.

- **Managed execution:** Virtualization of the execution environment not only allows increased security, but a wider range of features also can be implemented. In particular, sharing, aggregation, emulation, and isolation are the most relevant features (see Figure 3.2).
  - ✓ **Sharing** Virtualization allows the creation of a separate computing environments within the same host. In this way it is possible to fully exploit the capabilities of a powerful guest, which would otherwise be underutilized.
  - ✓ **Aggregation** Not only is it possible to share physical resource among several guests, but virtualization also allows aggregation, which is the opposite process. A group of separate hosts can be tied together and represented to guests as a single virtual host.
  - ✓ **Emulation** Guest programs are executed within an environment that is controlled by the virtualization layer, which ultimately is a program. This allows for controlling and tuning the environment that is exposed to guests.
  - ✓ **Isolation:** Virtualization allows providing guests whether they are operating systems, applications, or other entities with a completely separate environment, in which they are executed. The guest program performs its activity by interacting with an abstraction layer, which provides access to the underlying resources.
- **Portability:** The concept of portability applies in different ways according to the specific type of virtualization considered. In the case of a hardware virtualization solution, the guest is packaged into a virtual image that, in most cases, can be safely moved and executed on top of different virtual machines. Except for the file size, this happens with the same simplicity with which we can display a picture image in different computers.

3b. Give the taxonomy of virtualization techniques.

Ans: Virtualization covers a wide range of emulation techniques that are applied to different areas of computing. A classification of these techniques helps us better understand their characteristics and use of different scenarios.

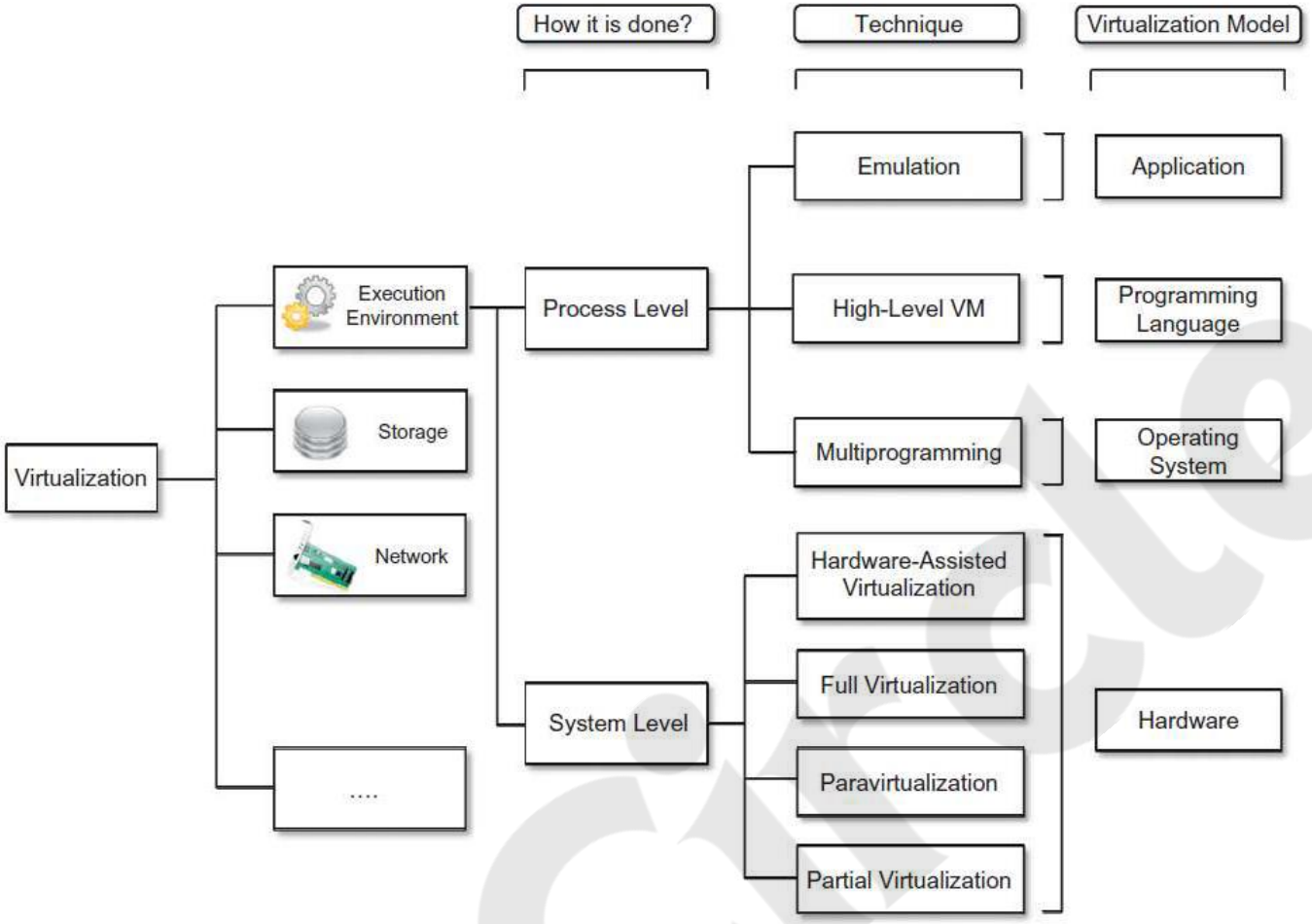


Fig: A taxonomy of virtualization techniques

The first classification discriminates against the service or entity that is being emulated. Virtualization is mainly used to emulate execution environments, storage, and networks. Among these categories, execution virtualization constitutes the oldest, most popular, and most developed area. Therefore, it deserves major investigation and a further categorization. In particular we can divide these execution virtualization techniques into two major categories by considering the type of host they require. Process-level techniques are implemented on top of an existing operating system, which has full control of the hardware. System-level techniques are implemented directly on hardware and do not require or require a minimum of support from an existing operating system.

Emulation Targets Virtualization techniques are mainly applied to emulate different services or entities. Specifically, virtualization is used to emulate three primary areas:

- **Execution Environments:** These are environments where programs, including operating systems and applications, can run as if they were on actual physical hardware.
- **Storage:** Virtualizing storage separates physical storage locations from the logical view users or programs interact with.
- **Networks:** Virtualizing networks allows different networks to operate as though they are independent of the underlying physical infrastructure.

How is it Done?

1. **Process-Level Virtualization**
    - Process-level virtualization allows virtualization within the execution environment. This form of virtualization is hosted on an existing operating system and provides a virtualized process space where applications can run.

The main techniques used here include:

- **Emulation:** This technique mimics a different system entirely, allowing applications meant for one system to run on another by imitating its architecture.
- **High-Level Virtual Machines (VMs):** These VMs operate at a high level and abstract the details of the hardware. They are particularly suited for programming languages like Java, where the code runs on a virtual machine instead of directly on hardware.
- **Multiprogramming:** This involves running multiple processes on a single system in a time-sharing manner, where the OS switches between different processes. This is an example of virtualization at the operating system level.

2. **System-Level Virtualization**

System-level virtualization operates directly on the hardware without needing an existing operating system to host it. It is more efficient in accessing the hardware resources. Key techniques under system-level virtualization are:

- **Hardware-Assisted Virtualization:** This technique uses the hardware's support to improve virtualization performance. It often involves using a hypervisor that interacts directly with the hardware to manage virtual machines.
- **Full Virtualization:** Here, the guest operating system runs in isolation as though it were directly using the hardware, without needing modification.
- **Paravirtualization:** In contrast to full virtualization, paravirtualization requires modification of the guest operating system so that it can interact more efficiently with the virtual environment.
- **Partial Virtualization:** Only part of the hardware resources is virtualized, meaning some applications may need to run directly on the host.

Virtualization Models

The virtualization models describe the environment created by the virtualization techniques:

- **Application-Level Virtualization:** This model allows individual applications to run in a virtual environment separate from the underlying system. This is commonly achieved through emulation.
- **Programming Language-Level Virtualization:** Programming language-level virtualization makes it possible for applications written in a particular language to run in a virtual machine designed for that language. High-level VMs such as the Java Virtual Machine (JVM) are examples of this type of virtualization model.
- **Operating System-Level Virtualization:** This model allows multiple user environments to run on a single OS, managed by the multiprogramming technique. It ensures that resources are shared but isolated between environments.
- **Hardware-Level Virtualization:** In system-level virtualization, hardware-level models are prevalent. These models allow multiple operating systems to share a single hardware platform through full, paravirtualization, or partial virtualization techniques.

3c. What is virtualization and what are its benefits.

Ans: Virtualization is the process of running a virtual instance of a computer system in a layer separate from the actual hardware. It is often inefficient and costly for organizations to deploy multiple servers to keep pace with their storage and processing needs. Instead, virtualization provides the ability to create multiple simulated environments from a single, physical hardware system.

Benefits of Virtualization:

- **Resource Efficiency:** Virtualization enables multiple virtual instances to run on a single physical machine, making better use of available resources (such as CPU, memory, and storage). This resource consolidation reduces the need for excess hardware, leading to better resource utilization and lower operational costs.
- **Cost Savings:** By reducing the number of physical servers, organizations can cut hardware, maintenance, space, and energy costs. Virtualization minimizes the need for physical infrastructure while still enabling flexibility and scalability.
- **Scalability and Flexibility**: Virtualized environments are highly scalable. Resources can be added or removed dynamically based on demand, allowing organizations to easily scale up or down. It also allows virtual machines (VMs) or containers to be moved across different physical servers, facilitating better management of resources.
- **Isolation:** Virtualization ensures that virtual machines or containers are isolated from one another, meaning that an issue in one VM or container (such as a crash, malware infection, or resource overload) does not impact others. This isolation enhances security and stability.

- **Disaster Recovery:** Virtualization offers powerful disaster recovery capabilities. Virtual machines can be backed up, migrated, and replicated to different locations with minimal effort. In the event of a system failure, a virtualized environment allows for quick failover to another server or data center, ensuring minimal downtime.
- **Simplified Management:** Centralized management of virtualized resources simplifies administration. Administrators can monitor, allocate, and manage resources from a single console. Additionally, automation tools can be used for deploying, scaling, and managing virtual machines, saving time and reducing human error.
- **Improved Testing and Development:** Virtualization enables the creation of isolated environments for testing and development without needing additional physical hardware. Developers can create snapshots, clones, and different configurations quickly, making it easier to test new software, patches, and configurations in a controlled environment.

4a. Explain virtualization and cloud computing and pros and cons of virtualization.

Ans: Virtualization plays an important role in cloud computing since it allows for the appropriate degree of customization, security, isolation, and manageability that are fundamental for delivering IT services on demand. Virtualization technologies are primarily used to offer configurable computing environments and storage. Network virtualization is less popular and, in most cases, is a complementary feature, which is naturally needed in build virtual computing systems. Particularly important is the role of virtual computing environment and execution virtualization techniques.

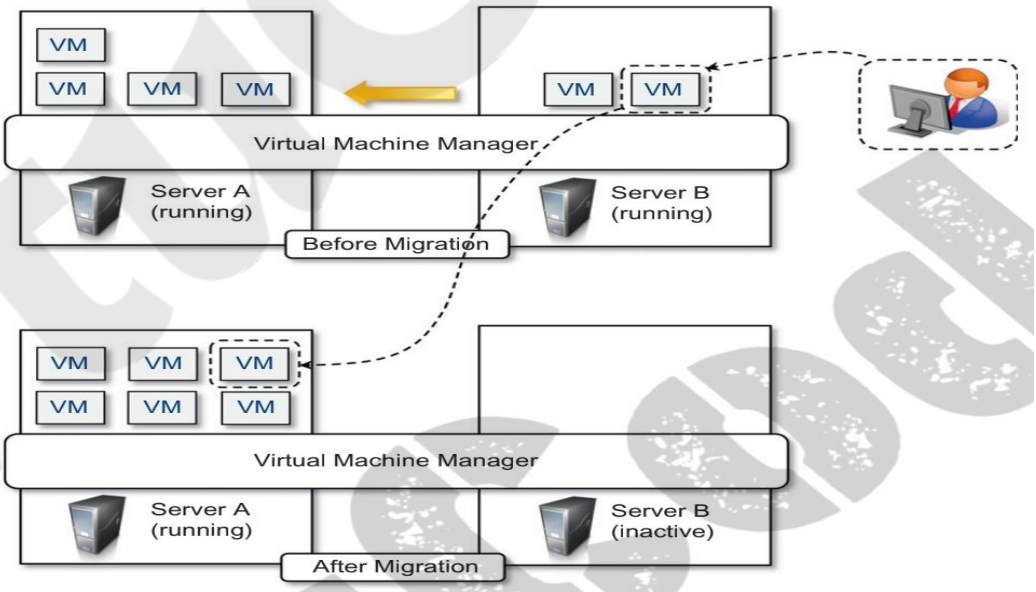


FIGURE 3.10
Live migration and server consolidation.

Since virtualization allows us to create isolated and controllable environments, it is possible to serve these environments with the same resource without them interfering with each other. If the underlying resources

are capable enough, there will be no evidence of such sharing. This opportunity is particularly attractive when resources are underutilized, because it allows reducing the number of active resources by aggregating virtual machines over a smaller number of resources that become fully utilized.

Server consolidation and virtual machine migration are principally used in the case of hardware virtualization, even though they are also technically possible in the case of programming language virtualization (see Figure 3.9). Storage virtualization constitutes an interesting opportunity given by virtualization technologies, often complementary to the execution of virtualization. Even in this case, vendors backed by large computing infrastructures featuring huge storage facilities can harness these facilities into a virtual storage service, easily partitionable into slices.

These slices can be dynamic and offered as a service. Again, opportunities to secure and protect the hosting infrastructure are available, as are methods for easy accountability of such services. Finally, cloud computing revamps the concept of desktop virtualization, initially introduced in the mainframe era. The ability to recreate the entire computing stack—from infrastructure to application services—on demand opens the path to having a complete virtual computer hosted on the infrastructure of the provider and accessed by a thin client over a capable Internet connection.

Pros and cons of virtualization

Advantages of virtualization

- **Managed execution and isolation** are perhaps the most important advantages of virtualization. In the case of techniques supporting the creation of virtualized execution environments, these two characteristics allow building secure and controllable computing environments. A virtual execution environment can be configured as a sandbox, thus preventing any harmful operation to cross the borders of the virtual host.
- **Portability** is another advantage of virtualization, especially for execution virtualization techniques. Virtual machine instances are normally represented by one or more files that can be easily transported with respect to physical systems. Moreover, they also tend to be self-contained since they do not have other dependencies besides the virtual machine manager for their use.
- Portability and self-containment also contribute to **reducing the costs of maintenance**, since the number of hosts is expected to be lower than the number of virtual machine instances. Since the guest program is executed in a virtual environment, there is very limited opportunity for the guest program to damage the underlying hardware.

Advantages of virtualization

- **Performance degradation:** Performance is definitely one of the major concerns in using virtualization technology. Since virtualization interposes an abstraction layer between the guest and the host, the guest can experience increased latencies. For instance, in the case of hardware virtualization, where the intermediate emulates a bare machine on top of which an entire system can be installed, the causes of performance degradation can be traced back to the overhead introduced by the following activities:
  - ✓ Maintaining the status of virtual processors
  - ✓ Support of privileged instructions (trap and simulate privileged instructions)
  - ✓ Support of paging within VM
  - ✓ Console functions
- **Inefficiency and degraded user experience:** Virtualization can sometime lead to an inefficient use of the host. In particular, some of the specific features of the host cannot be exposed by the abstraction layer and then become inaccessible. In the case of hardware virtualization, this could happen for device drivers: The virtual machine can sometime simply provide a default graphic card that maps only a subset of the features available in the host.
- **Security holes and new threats:** Virtualization opens the door to a new and unexpected form of phishing. The capability of emulating a host in a completely transparent manner led the way to malicious programs that are designed to extract sensitive information from the guest. AMD processor family, moves the execution of the installed OS within a virtual machine.

4b. Explain hypervisors and its types.

Ans: A hypervisor is a software layer that enables virtualization by creating and managing virtual machines (VMs) on a physical machine. It sits between the hardware and the operating systems, allowing multiple virtual operating systems to run on a single physical system. The hypervisor controls the hardware resources and allocates them to each virtual machine, making sure that the VMs operate independently and securely from one another.

Types of Hypervisors

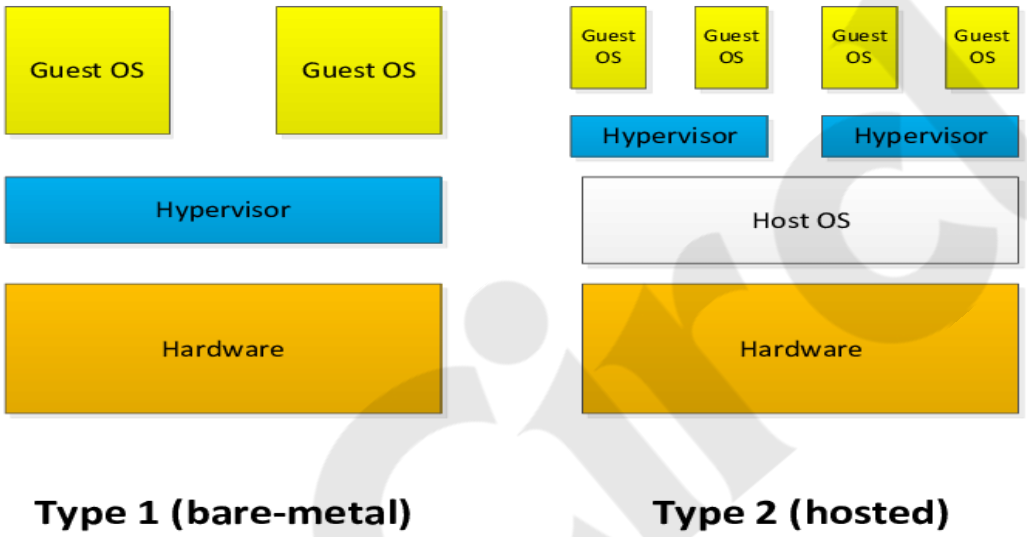
Hypervisors can be broadly classified into two types:

1. **Type 1 Hypervisor (Bare-metal Hypervisor):** A Type 1 hypervisor is directly installed on the physical hardware, without requiring a host operating system. It has full control over the hardware resources and manages the virtual machines directly. This type of hypervisor is typically used in server environments and is known for its high performance and efficiency because it operates with minimal overhead.
   - **Direct access to hardware:** Since it runs directly on the physical machine, there is no intermediary operating system, which allows better performance and resource allocation.

- **Efficiency and stability:** Type 1 hypervisors generally offer higher stability, performance, and security, as they interact directly with the hardware.
- **Common use cases:** Used in data centres, cloud environments, and large-scale virtualized environments.

Examples of Type 1 Hypervisors:

- **VMware ESXi:** A widely used hypervisor in enterprise environments for server virtualization.
- **Microsoft Hyper-V:** A hypervisor developed by Microsoft for both enterprise and cloud computing environments.



2. **Type 2 Hypervisor (Hosted Hypervisor):** A **Type 2 hypervisor** is installed on top of a host operating system. It runs as an application within the host OS and relies on the underlying operating system to manage hardware resources. Type 2 hypervisors are typically used in less resource-intensive environments like personal or development environments, as they tend to have more overhead compared to Type 1 hypervisors.
    - **Relies on host OS:** The host operating system manages the hardware resources, and the hypervisor runs as a program within the host OS.
    - **Higher overhead:** Since it operates through the host OS, Type 2 hypervisors usually have more overhead compared to Type 1 hypervisors, leading to slightly lower performance.
    - **Common use cases:** Primarily used in desktop environments or for testing and development purposes where resource efficiency is less critical.

Examples of Type 2 Hypervisors:

- **VMware Workstation**: A popular hypervisor for running virtual machines on a desktop or laptop.

- **Oracle VirtualBox:** A free, open-source hypervisor used for running virtual machines on personal computers.

4c. Discuss machine reference model of execution virtualization.

Ans: Virtualizing an execution environment at different levels of the computing stack requires a reference model that defines the interfaces between the levels of abstractions, which hide implementation details. From this perspective, virtualization techniques actually replace one of the layers and intercept the calls that are directed toward it. Therefore, a clear separation between layers simplifies their implementation, which only requires the emulation of the interfaces and a proper interaction with the underlying layer.

ISA is the interface between hardware and software, and it is important to the operating system (OS) developer (System ISA) and developers of applications that directly manage the underlying hardware (User ISA). The application binary interface (ABI) separates the operating system layer from the applications and libraries, which are managed by the OS. ABI covers details such as low-level data types, alignment, and call conventions and defines a format for executable programs. System calls are defined at this level.

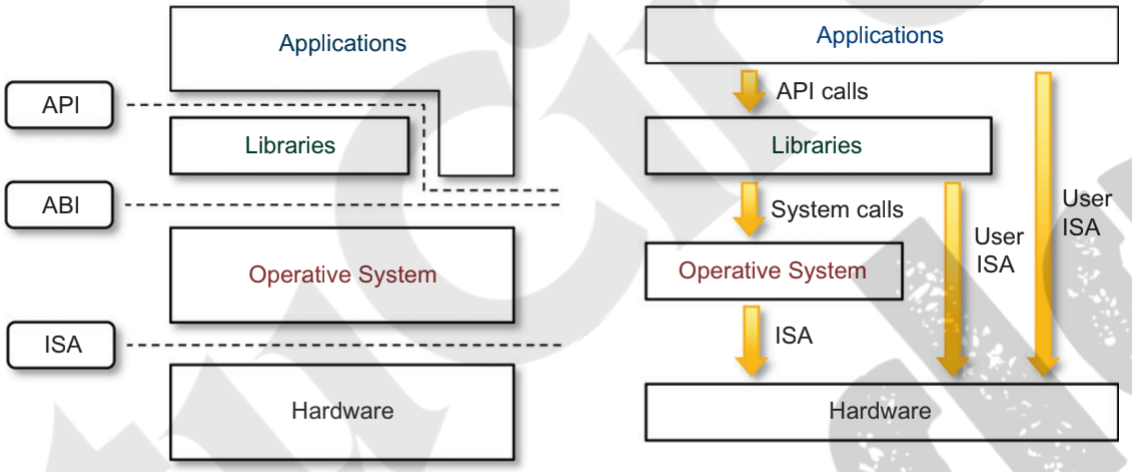


FIGURE 3.4
A machine reference model.

For any operation to be performed in the application-level API, ABI and ISA are responsible for making it happen. The high-level abstraction is converted into machine-level instructions to per- form the actual operations supported by the processor. The machine-level resources, such as processor registers and main memory capacities, are used to perform the operation at the hardware level of the central processing unit (CPU). This layered approach simplifies the development and implementation of computing systems and simplifies the implementation of multitasking and the coexistence of multiple executing environments.

In fact, such a model not only requires limited knowledge of the entire computing stack, but it also provides ways to implement a minimal security model for managing and accessing shared resources. Some types of architecture feature more than one class of privileged instructions and implement a finer control of how these instructions can be accessed. For instance, a possible implementation features a hierarchy of privileges (see Figure 3.5) in the form of ring-based security: Ring 0, Ring 1, Ring 2, and Ring 3; Ring 0 is in the most privileged level and Ring 3 in the least privileged level. Ring 0 is used by the kernel of the OS, rings 1 and 2 are used by the OS-level services, and Ring 3 is used by the user. Recent systems support only two levels, with Ring 0 for supervisor mode and Ring 3 for user mode.

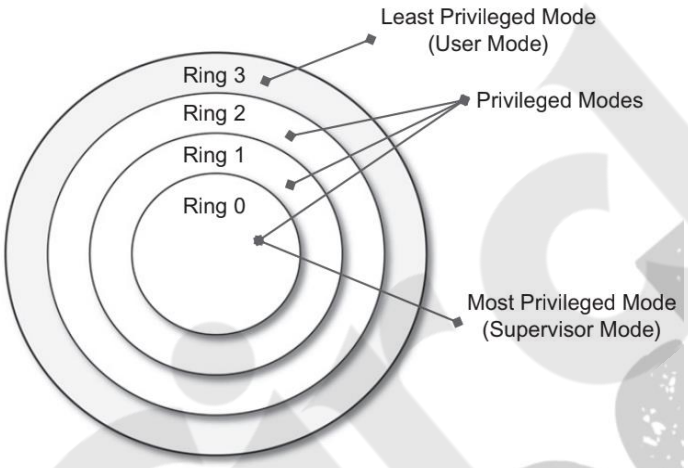


FIGURE 3.5
Security rings and privilege modes.

All the current systems support at least two different execution modes: supervisor mode and user mode. The first mode denotes an execution mode in which all the instructions (privileged and nonprivileged) can be executed without any restriction. This mode, also called master mode or kernel mode, is generally used by the operating system (or the hypervisor) to perform sensitive operations on hardware-level resources. In user mode, there are restrictions to control the machine-level resources. If code running in user mode invokes the privileged instructions, hardware interrupts occur and trap the potentially harmful execution of the instruction.

MODULE-3

5a. Briefly Explain cloud computing architecture with a neat diagram.

Ans: It is possible to organize all the concrete realizations of cloud computing into a layered view covering the entire stack (see Figure 4.1), from hardware appliances to software systems. Cloud resources are harnessed to offer "computing horsepower" required for providing services. Often, this layer is implemented using a data centre in which hundreds and thousands of nodes are stacked together. Cloud infrastructure can be heterogeneous in nature because a variety of resources, such as clusters and even

networked PCs, can be used to build it. Moreover, database systems and other storage services can also be part of the infrastructure.

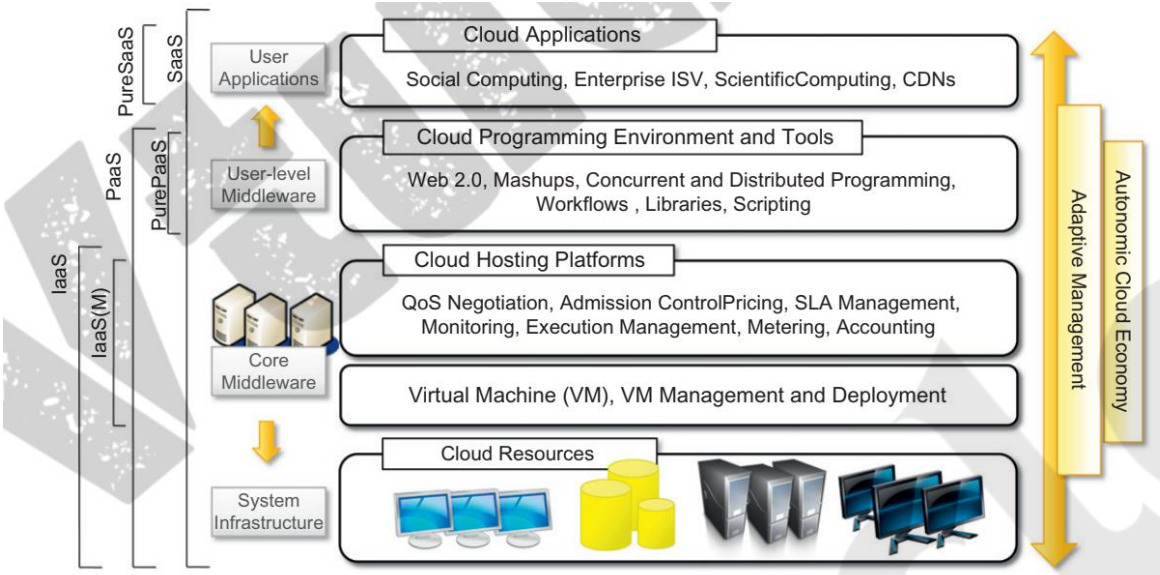


FIGURE 4.1

The cloud computing architecture.

The physical infrastructure is managed by the core middleware, the objectives of which are to provide an appropriate runtime environment for applications and to best utilize resources. At the bottom of the stack, virtualization technologies are used to guarantee runtime environment customization, application isolation, sandboxing, and quality of service. Hardware virtualization is most commonly used at this level. Hypervisors manage the pool of resources and expose the distributed infrastructure as a collection of virtual machines.

By using virtual machine technology, it is possible to finely partition the hardware resources such as CPU and memory and to virtualize specific devices, thus meeting the requirements of users and applications. This solution is generally paired with storage and network virtualization strategies, which allow the infrastructure to be completely virtualized and controlled. According to the specific service offered to end users, other virtualization techniques can be used; for example, programming-level virtualization helps in creating a portable runtime environment where applications can be run and controlled.

The combination of cloud hosting platforms and resources is generally classified as a Infrastructure-as-a-Service (IaaS) solution. We can organize the different examples of IaaS into two categories: Some of them provide both the management layer and the physical infrastructure; others provide only the management layer (IaaS (M)). In this second case, the management layer is often integrated with other IaaS solutions that provide physical infrastructure and adds value to them.

IaaS solutions are suitable for designing the system infrastructure but provide limited services to build applications. Such service is provided by cloud programming environments and tools, which form a new layer for offering users a development platform for applications. The range of tools include Web-based interfaces, command-line tools, and frameworks for concurrent and distributed programming. In this scenario, users develop their applications specifically for the cloud by using the API exposed at the user-level middleware.

Internet. Other applications belonging to this layer are those that strongly leverage the Internet for their core functionalities that rely on the cloud to sustain a larger number of users; this is the case of gaming portals and, in general, social networking websites. As a vision, any service offered in the cloud computing style should be able to adaptively change and expose an autonomic behaviour, in particular for its availability and performance.

5b. Explain IAAS with a neat diagram.

Ans: Infrastructure- and Hardware-as-a-Service (IaaS/HaaS) solutions are the most popular and developed market segment of cloud computing. They deliver customizable infrastructure on demand. The available options within the IaaS offering umbrella range from single servers to entire infrastructures, including network devices, load balancers, and database and Web servers. The main technology used to deliver and implement these solutions is hardware virtualization: one or more virtual machines opportunely configured and interconnected define the distributed system on top of which applications are installed and deployed.

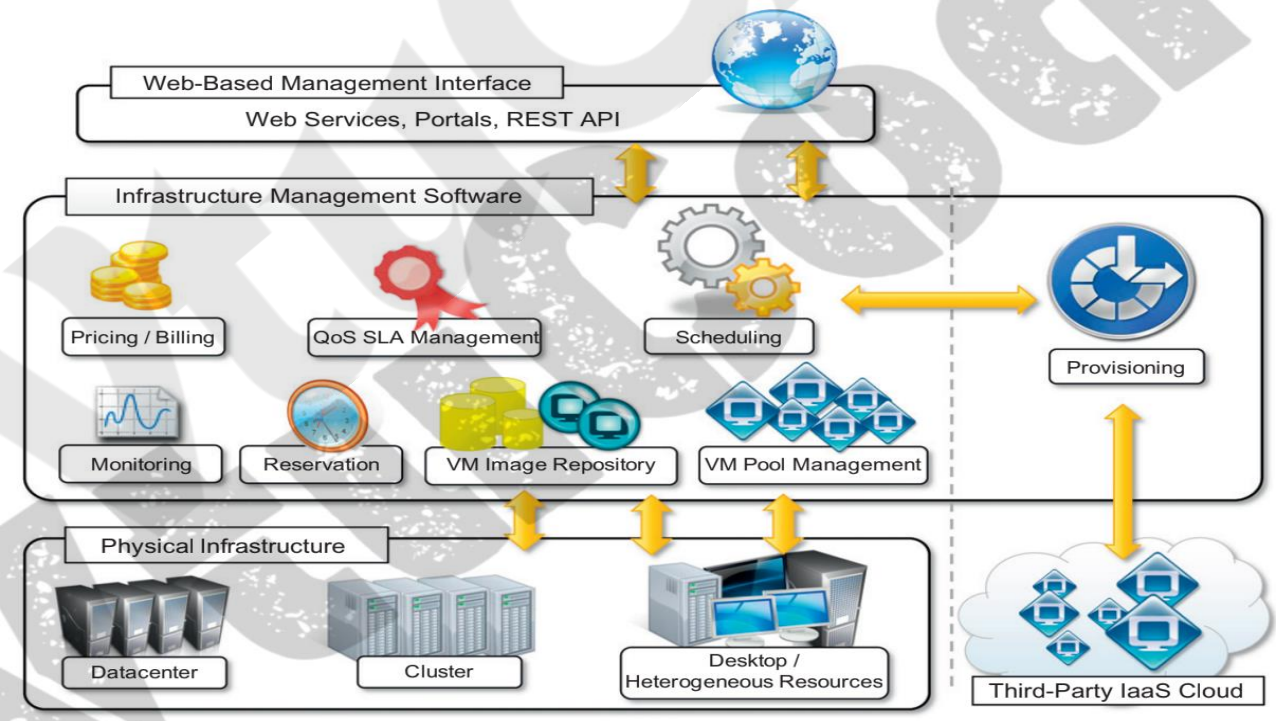


FIGURE 4.2
Infrastructure-as-a-Service reference implementation.

A central role is played by the scheduler, which is in charge of allocating the execution of virtual machine instances. The scheduler interacts with the other components that perform a variety of tasks:

- The pricing and billing component takes care of the cost of executing each virtual machine instance and maintains data that will be used to charge the user.
- The monitoring component tracks the execution of each virtual machine instance and maintains data required for reporting and analysing the performance of the system.
- The reservation component stores the information of all the virtual machine instances that have been executed or that will be executed in the future.
- If support for QoS-based execution is provided, a QoS/SLA management component will maintain a repository of all the SLAs made with the users; together with the monitoring component, this component is used to ensure that a given virtual machine instance is executed with the desired quality of service.
- The VM repository component provides a catalog of virtual machine images that users can use to create virtual instances. Some implementations also allow users to upload their specific virtual machine images.
- A VM pool manager component is responsible for keeping track of all the live instances.

The bottom layer is composed of the physical infrastructure, on top of which the management layer operates. As previously discussed, the infrastructure can be of different types; the specific infrastructure used depends on the specific use of the cloud. A service provider will most likely use a massive datacentre containing hundreds or thousands of nodes.

5c. What is SAAS. Explain its characteristics and its initial benefits.

Ans: Software-as-a-Service (SaaS) is a software delivery model that provides access to applications through the Internet as a Web-based service. It provides a means to free users from complex hardware and software management by offloading such tasks to third parties, which build applications accessible to multiple users through a Web browser. The core characteristics of SaaS:

- The product sold to customer is application access.
- The application is centrally managed.
- The service delivered is one-to-many.
- The service delivered is an integrated solution delivered on the contract, which means provided as promised.

Initially the SaaS model was of interest only for lead users and early adopters. The benefits delivered at that stage were the following:

- Software cost reduction and total cost of ownership (TCO) were paramount
- Service-level improvements

- Rapid implementation
- Standalone and configurable applications
- Rudimentary application and data integration
- Subscription and pay-as-you-go (PAYG) pricing

6a. Explain PAAS with a neat diagram.

Ans: Platform-as-a-Service (PaaS) solutions provide a development and deployment platform for running applications in the cloud. They constitute the middleware on top of which applications are built. A general overview of the features characterizing the PaaS approach is given in Figure 4.3.

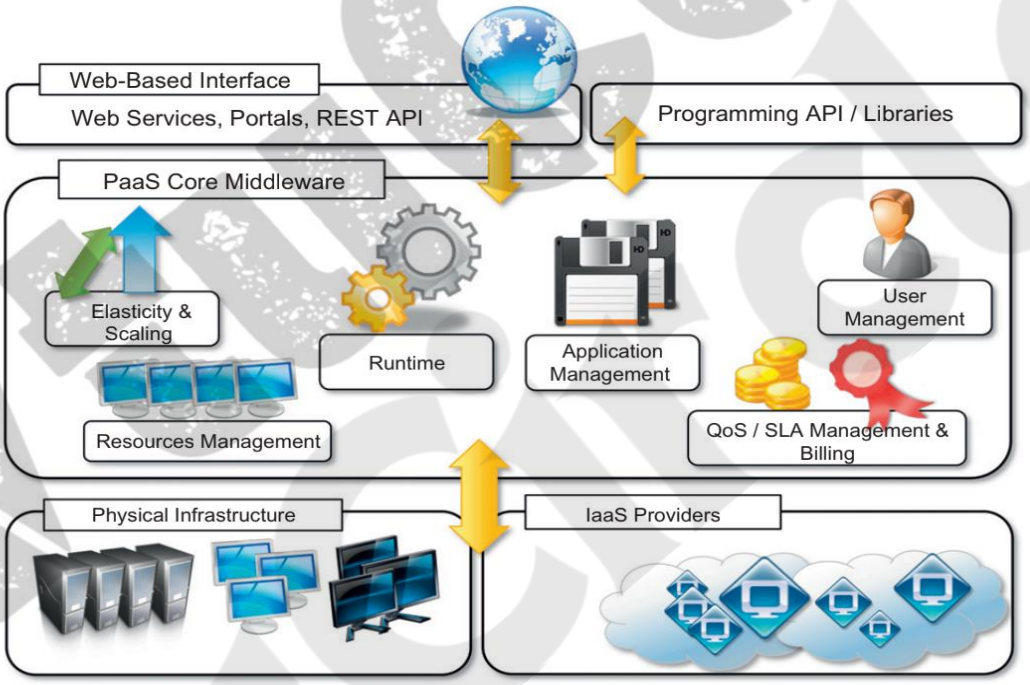


FIGURE 4.3
The Platform-as-a-Service reference model.

Application management is the core functionality of the middleware. PaaS implementations provide applications with a runtime environment and do not expose any service for managing the underlying infrastructure. They automate the process of deploying applications to the infrastructure, configuring application components, provisioning and configuring supporting technologies such as load balancers and databases, and managing system change based on policies set by the user. Developers design their systems in terms of applications and are not concerned with hardware (physical or virtual), operating systems, and other low-level services.

Developers generally have the full power of programming languages such as Java, .NET, Python, or Ruby, with some restrictions to provide better scalability and security. In this case the traditional development environments can be used to design and develop applications, which are then deployed on the cloud by

using the APIs exposed by the PaaS provider. Specific components can be offered together with the development libraries for better exploiting the services offered by the PaaS environment. Sometimes a local runtime environment that simulates the conditions of the cloud is given to users for testing their applications before deployment. This environment can be restricted in terms of features, and it is generally not optimized for scaling.

PaaS solutions can offer middleware for developing applications together with the infrastructure or simply provide users with the software that is installed on the user premises.

- **Runtime framework:** This framework represents the "software stack" of the PaaS model and the most intuitive aspect that comes to people's minds when they refer to PaaS solutions. The runtime framework executes end-user code according to the policies set by the user and the provider.
- **Abstraction:** PaaS solutions are distinguished by the higher level of abstraction that they provide. Whereas in the case of IaaS solutions the focus is on delivering "raw" access to virtual or physical infrastructure, in the case of PaaS the focus is on the applications the cloud must support. This means that PaaS solutions offer a way to deploy and manage applications on the cloud rather than a bunch of virtual machines on top of which the IT infrastructure is built and configured.
- **Automation:** PaaS environments automate the process of deploying applications to the infrastructure, scaling them by provisioning additional resources when needed. This process is performed automatically and according to the SLA made between the customers and the provider. This feature is normally not native in IaaS solutions, which only provide ways to provision more resources.
- **Cloud services:** PaaS offerings provide developers and architects with services and APIs, helping them to simplify the creation and delivery of elastic and highly available cloud applications. These services are the key differentiators among competing PaaS solutions and generally include specific components for developing applications, advanced services for application monitoring, management, and reporting.

6b. Describe the fundamental features of the economic and business model behind cloud computing.

Ans The economic and business model behind cloud computing is fundamentally built on flexibility, cost-efficiency, and scalability, shifting traditional IT infrastructure and software management into an operational, pay-as-you-go model.

- **Reduction in Capital Costs**: Cloud computing eliminates the need for businesses to purchase expensive IT infrastructure and software as capital assets. Instead, companies can lease resources (such as servers and storage) and subscribe to software on a pay-as-you-go basis.

- **Depreciation and Lifetime Costs**: The traditional IT model involves capital assets that depreciate over time, such as hardware losing value and software becoming outdated. Cloud computing removes the need for businesses to manage depreciation, as the hardware and software are owned and maintained by the cloud provider.
- **Pay-as-You-Go Model**: One of the most compelling features of cloud computing is the pay-as-you-go model, where businesses pay only for the services they use, as opposed to committing to large, upfront capital expenditures. This model provides significant financial flexibility, allowing companies to scale their IT resources based on demand, optimizing costs based on current business needs.
- **Operational Cost Control**: Cloud computing converts capital expenses into operational costs, which can be more easily managed and controlled. Businesses can lease IT infrastructure and pay for software subscriptions, aligning costs with actual usage. This operational expenditure model allows businesses to better align IT costs with their revenue or operational activity, improving financial predictability and avoiding unnecessary investments in underutilized resources.
- **Elimination of Maintenance and Administrative Costs**: With cloud computing, companies reduce or completely eliminate costs related to maintaining physical IT infrastructure, including data centres, electricity, cooling, and maintenance staff. Cloud service providers take care of these operational aspects, so businesses no longer need to maintain an in-house IT department or support centre for infrastructure management, leading to lower administrative and operational expenses.
- **Scalability and Flexibility**: Cloud computing offers scalability by allowing businesses to adjust their resource allocation on demand. For example, companies can quickly scale up to accommodate peak demand and scale down when resources are no longer needed. This flexibility helps businesses avoid the financial burden of over-investing in IT infrastructure or having idle resources.
- **Elimination of Indirect Costs**: Cloud computing can also eliminate indirect costs such as software licensing fees, support, and carbon emissions. With cloud-based software, businesses typically pay a subscription fee, bypassing large upfront licensing costs. Moreover, the shared data centres used by cloud providers are often more energy-efficient, potentially leading to reduced carbon footprints.
- **Cloud Pricing Models**: Cloud providers offer various pricing strategies that cater to different business needs:
  - ✓ **Tiered Pricing**: Customers pay based on predefined service tiers with fixed specifications and service levels (e.g., Amazon EC2).

- ✓ **Per-Unit Pricing**: Charges are based on specific services like data transfer or memory usage (e.g., GoGrid).
- ✓ **Subscription-Based Pricing**: Common for Software-as-a-Service (SaaS), where businesses pay a recurring fee for using software.

- **Business Flexibility and Focus**: By outsourcing IT infrastructure and software management to cloud providers, businesses can focus more on their core activities rather than managing IT resources. This leads to increased business agility and the ability to innovate without being bogged down by technical infrastructure concerns.

6c. List and Explain some of the challenges in cloud computing.

Ans: Cloud computing offers numerous benefits, but it also comes with various challenges. Here are some of the key challenges in cloud computing:

1. **Cloud Interoperability and Standards**
   - **Challenge:** The lack of standardization across cloud platforms creates interoperability issues, leading to vendor lock-in. This makes it difficult for customers to switch providers or integrate different services.
   - **Efforts:** Initiatives like the Cloud Computing Interoperability Forum (CCIF) and Open Cloud Manifesto aim to standardize cloud technologies, particularly for IaaS vendors, with efforts like the Open Virtualization Format (OVF) attempting to improve compatibility between virtual machines.
2. **Scalability and Fault Tolerance**
   - **Challenge**: Ensuring that cloud systems can scale effectively and handle large volumes of traffic without performance degradation is a critical issue. Additionally, fault tolerance, which ensures the system remains operational even during failures, is essential for maintaining uptime and reliability.
   - **Efforts**: Designing cloud middleware that can scale across various dimensions (performance, size, load) and be resilient to failure is key to addressing these issues.
3. **Security, Trust, and Privacy**
   - **Challenge**: The use of cloud services, particularly those involving virtualized environments, raises concerns about data privacy, security, and trust. Cloud providers have access to sensitive data, and the complex stack of third-party services increases the potential risks of unauthorized access or data breaches.
   - **Concerns**: Lack of control over data and processes makes it difficult to ensure compliance with privacy regulations and to trust the provider. Identifying liability in cases of privacy violations is also challenging.

4. **Organizational Aspects**
   - **Challenge**: Cloud computing changes how IT services are consumed and managed, requiring enterprises to adjust their organizational structure, processes, and roles. IT departments face the challenge of adapting to a model where many services are outsourced to the cloud.
   - **Concerns**: Questions arise about the role of the IT department, compliance issues, loss of control over service management, and the impact on employees' skills and responsibilities as IT services move to the cloud.

MODULE-4

7a. Explain operating system security and virtual machine security.

Ans: Operating System Security

Operating system (OS) security is crucial for protecting the integrity, confidentiality, and availability of data and system resources. It encompasses a set of mechanisms that prevent unauthorized access, data tampering, and malicious attacks from affecting the system. Based on the content provided, OS security can be described through the following key aspects:

1. **Mandatory OS Security:**
   - **Access Control:** It manages how system objects (e.g., files, devices, processes) are accessed by users or applications. Access control policies ensure that only authorized entities can perform specific actions.
   - **Authentication:** Verifying the identity of users or processes is a critical part of OS security. Proper authentication prevents unauthorized access to the system.
   - **Cryptography:** This involves securing data through encryption and other cryptographic techniques, ensuring that sensitive information remains confidential and protected from tampering.
2. **Trusted Applications:** Security-related applications, such as those handling authentication or data encryption, must be configured to use the minimum privileges necessary. This prevents them from accessing unnecessary system resources, reducing potential attack surfaces.
3. **Discretionary vs. Mandatory Security:**
   - **Discretionary Security**: Involves user-defined access control, where users can grant or deny access to resources. However, this can lead to security breaches due to human error or carelessness.
   - **Mandatory Security:** Managed by system administrators, mandatory security policies are enforced strictly by the OS. This provides a more robust and controlled environment compared to discretionary security, which is prone to misconfigurations.

4. **Limitations in Commercial OS:** Some operating systems, such as Windows NT, have inadequate fine-grained security mechanisms. Applications may inherit privileges from other programs they invoke, which can expose the system to unauthorized access or attacks.
5. **Protecting Against Malicious Code:** OS security mechanisms need to guard against malicious software like viruses, worms, and other forms of mobile code (e.g., Java applets). While Java's security manager restricts certain actions, it cannot completely prevent attacks such as code tampering.
6. **Platform Vulnerability:** The security of an entire platform can be compromised if the weakest application is exploited. Even if OS-level security features (like authentication) are in place, they may not be sufficient for securing communications or transactions without additional protections.
7. **Application-Specific Security:** Sometimes, security needs to extend beyond the OS, especially for specific applications like digital signatures used in e-commerce. In such cases, additional security mechanisms are required to ensure the integrity and confidentiality of the transactions.
8. **Access Control Decomposition:** OS security should decompose access control into distinct components (such as the enforcer and decider roles) to ensure a clear and secure process for making access decisions. This helps to avoid ambiguity and potential security gaps.

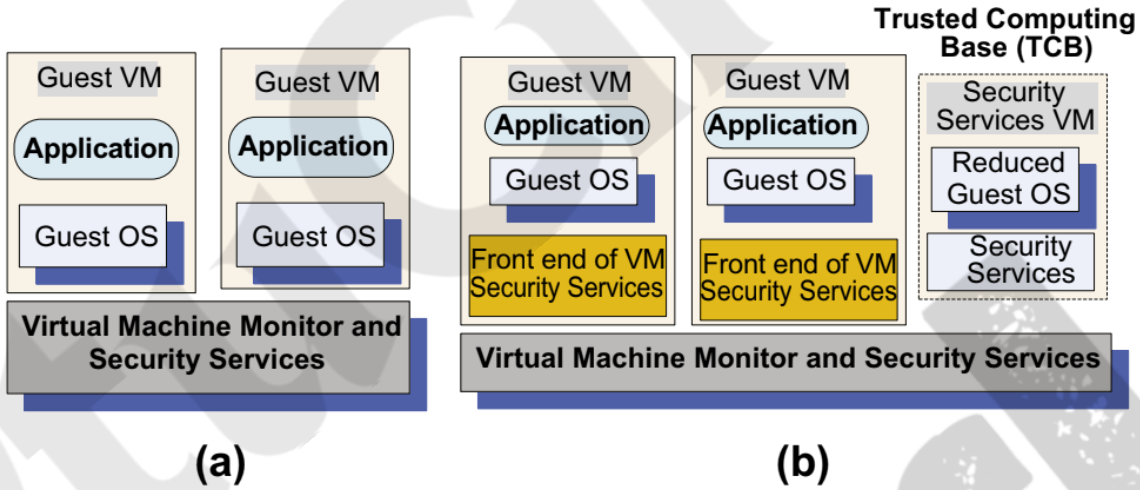


FIGURE 9.2

(a) Virtual security services provided by the VMM. (b) A dedicated security VM.

Virtual Machine Security

Virtual Machine (VM) security focuses on securing virtualized environments where multiple VMs run on the same physical hardware. It ensures that virtual machines are isolated from each other, preventing malicious activities within one VM from affecting others. Based on the content provided, VM security can be described as follows:

1. **Traditional VM Security Model:**

- In this model, a Virtual Machine Monitor (VMM) controls access to hardware resources and enforces security measures to isolate virtual machines. The VMM ensures that each VM operates independently and securely, with minimal risk of interference from other VMs.

2. **Security Services in VMs:**
   - Trusted Computing Base (TCB): The TCB is a critical part of the VM environment. If the TCB is compromised, the security of the entire system is at risk. The VMM itself can provide security services or rely on a dedicated security VM to enforce policies.
   - Cloning and Replication: One advantage of virtualization is the ability to clone VMs, which can be used to test suspicious applications or simulate attacks. Cloning allows security professionals to observe how a potentially malicious application behaves without compromising the original system.
3. **Challenges in VM Security:**
   - The VMM has access to the state of all VMs, but it only sees raw data regarding the guest OS. This can make it difficult for the VMM to manage security at a higher logical level (such as at the file system level).
   - Sophisticated attackers can fingerprint virtual environments, avoiding detection by honeypots set up within VMs. They can also target VM logs that contain sensitive data, so these logs need to be tightly protected.
4. **Costs and Overheads:**
   - Virtualization provides a higher level of security through better isolation, but it comes with costs. These include higher hardware resource requirements (e.g., more CPU cycles, memory, disk, and network bandwidth), as well as the complexity and overhead involved in maintaining and developing VMMs.
5. **VM-Based Intrusion Prevention:**
   - **VMM-Based Threats:**
     - **Resource Starvation and Denial of Service:** Misconfigurations or rogue VMs can cause resource starvation, affecting other VMs running on the same host.
     - **VM Side-Channel Attacks:** A rogue VM could exploit weaknesses in the isolation of inter-VM traffic, leading to data leakage or attacks on other VMs.
     - **Buffer Overflow Attacks:** Similar to traditional OS-based attacks, buffer overflows can affect VMs if vulnerabilities exist within the guest OS or VMM.
   - **VM-Based Threats:**
     - **Deployment of Rogue or Insecure VMs:** If proper access controls aren't in place, unauthorized users can deploy insecure VMs, compromising the security of the system.

- **Tampered VM Images:** VM images must be protected from tampering, as insecure or modified images could introduce vulnerabilities when deployed.

6. **Security Group Threats (NIST):**
   - The NIST security group identifies several threats related to VMMs and VMs, including insecure VM images, rogue VM deployments, and attacks exploiting poor configuration of access controls.

7b. Explain the security risks posed by shared images and management OS.

Ans: The security risks posed by shared images and management operating systems (OS) are significant and multifaceted, as illustrated by the analysis of over 5,000 Amazon Machine Images (AMIs) in the AWS ecosystem. These risks affect both the users who utilize public AMIs and the providers who create and share them. Here's an in-depth look at the various security threats:

1. **Backdoors and Leftover Credentials**

- **SSH Keys and Passwords:** A major risk in shared AMIs is the presence of leftover credentials, such as SSH keys and passwords, which allow unauthorized remote access. For example, the creator of an AMI may leave their own SSH public keys or passwords in the image, creating a backdoor that enables an attacker to access any instance of that AMI.
- **Recovery of Credentials:** Attackers can recover private credentials using simple tools (e.g., John the Ripper for password cracking) and gain unauthorized access to the system. This backdoor can be exploited even further if the image creator has left their credentials accessible without any password protection or encryption.

2. **Unsolicited Connections**

- **Forwarding of Sensitive Data:** Some AMIs contain syslog daemons or other services configured to send sensitive information to external servers, such as login details, IP addresses, or event logs. These unsolicited connections can reveal privileged data or allow attackers to track activity, potentially leading to further exploitation.
- **Hard to Distinguish Malicious from Legitimate Connections:** While some outgoing connections may be legitimate (e.g., software update connections), others may be malicious, and it can be challenging to differentiate between them without detailed inspection. This poses a risk of data leakage.

3. **Malware**

- **Trojan and Spyware Infections:** The audit of public AMIs discovered that some Windows AMIs were infected with Trojans, including variants that allowed attackers to monitor activity (keylogging) or steal sensitive data.

- **Exploitation of Vulnerabilities:** Some of these infected AMIs had vulnerabilities that allowed malware to persist or propagate, further compromising the security of instances created from these images.

4. **Recovery of Sensitive Data from Deleted Files**
   - **Unclean Deletion of Sensitive Information:** When an AMI is created, sensitive information may be left behind in deleted files or free disk space. If proper secure deletion methods (e.g., shred or wipe) are not used, attackers can recover deleted files, including credentials, API keys, and other sensitive data.
   - **Disk Blocks Containing Sensitive Information:** In the case of images created using block-level tools (e.g., ec2-bundle-image), the file system may contain free disk blocks that still hold remnants of deleted files. These blocks can be recovered using standard utilities, exposing sensitive data.
5. **Exposed User and System Information**
   - **Browser and Shell History:** Some AMIs contained history files (e.g., ~/.bash_history, ~/.sh_history) that revealed sensitive commands run by the user, including API keys, DNS passwords, and other private credentials. Such information could easily be exploited by attackers who gain access to the AMI image.
   - **IP Address Leaks:** The audit found that some AMIs contained logs revealing IP addresses of other systems owned by the same user. This information could be used to track or target other systems in the same network, leading to broader attacks.
   - **Browser History and DNS Management:** Some AMIs also contained browser history and other potentially sensitive data. In particular, browser history could expose the domains a user has interacted with, while unprotected API keys could allow attackers to gain unauthorized access to cloud resources.
6. **Vulnerabilities in the Management Operating System**
   - **Lack of Security Patches:** Many of the images audited were outdated, with some being several years old. Older AMIs are more likely to contain known software vulnerabilities that can be exploited by attackers.
   - **Unpatched OS and Software Vulnerabilities:** Outdated operating systems or software in AMIs, especially ones that haven't been patched in a long time, pose significant risks. The audit found that 98% of Windows AMIs had critical vulnerabilities, making them prime targets for exploitation by attackers.
7. **Misconfiguration of Security Settings**
   - **Unprotected Access:** Some AMIs were found to have misconfigured security settings, such as improperly set SSH or RDP access controls, or weak password policies. This makes them vulnerable to brute-force attacks or unauthorized access.

Mitigation Strategies:

To mitigate these risks, both image providers and users should adopt the following best practices:

- Ensure proper cleanup of credentials and keys before sharing an AMI, including resetting SSH keys, passwords, and ensuring no sensitive files remain.
- Use secure deletion tools to wipe data properly, ensuring that deleted files cannot be recovered.
- Regularly update and patch images to ensure they don't contain known vulnerabilities.
- Run security audits and scans to identify malware, backdoors, and configuration issues.
- Implement proper access controls, such as using strong passwords, disabling password-based logins where possible, and ensuring SSH keys are protected by passphrases.
- Use the cloud-init script to regenerate keys and reset system information upon image instantiation, preventing man-in-the-middle attacks.

8a. Explain the concept of privacy impact assessment and its importance in cloud computing.

Ans: A Privacy Impact Assessment (PIA) is a tool used to identify, assess, and manage privacy risks in information systems, particularly when new technologies or processes that involve the handling of personal data are being developed or deployed. The PIA aims to ensure that privacy considerations are integrated into the design and implementation of a system or project from the outset, rather than being addressed reactively after problems arise. In cloud computing, a PIA is of particular importance due to the unique privacy challenges associated with storing and processing data on remote servers operated by Cloud Service Providers (CSPs). Cloud environments often involve the outsourcing of data storage and management to third-party providers, which creates several privacy concerns, including:

- **Loss of Control:** Once data is stored on a CSP's servers, the individual or organization that owns the data loses control over its physical location, how long it is stored, and the access rights to that data.
- **Unauthorized Secondary Use:** Cloud service providers may use personal data for purposes beyond its original intent, such as for targeted advertising. A PIA helps ensure that any secondary use of data complies with privacy policies and regulations.
- **Data Proliferation:** Cloud environments often involve multiple copies of data stored across various servers and locations. This proliferation can increase the risk of unauthorized access or data breaches.
- **Dynamic Provisioning and Outsourcing**: The flexible nature of cloud services allows for dynamic provisioning, where resources (e.g., servers, storage) can be allocated and deallocated as needed. This outsourcing often involves subcontractors, whose access to data may not be clearly defined.

The importance of PIA in cloud computing can be summarized as follows:

1. **Risk Mitigation**: By identifying privacy risks early in the process, a PIA helps organizations mitigate potential threats such as data breaches or misuse of personal information.
2. **Compliance with Regulations**: Cloud service providers and organizations must comply with various privacy laws and regulations, such as the GDPR in the EU, which require strict protection of personal data.
3. **Transparency**: Conducting a PIA promotes transparency in how personal data is handled, stored, and shared by cloud service providers. This transparency builds trust with users, who may be more willing to adopt cloud services if they are assured that their privacy is being respected.
4. **Informed Decision-Making**: A PIA provides organizations with the information necessary to make informed decisions about the use of cloud services. It helps them choose the most appropriate cloud provider, negotiate data protection terms, and design systems that respect privacy from the start.
5. **Enhanced Security**: Privacy and security often go hand-in-hand. By identifying privacy risks, a PIA also helps identify potential security vulnerabilities in the cloud environment, enabling organizations to strengthen their security measures.

8b. Explain the following associated with cloud computing i) cloud security risks ii) Security: the top concern for cloud users.

Ans:

i) cloud security risks

Cloud computing introduces several security risks due to the shared and distributed nature of its resources. These risks can impact both cloud users and providers, and can be categorized into the following key areas:

1. **Data Breaches and Loss:** Sensitive data stored in the cloud is at risk of being accessed by unauthorized parties, either through hacking, internal threats, or inadequate security measures by the cloud provider.
2. **Account Hijacking:** Attackers may steal or compromise user credentials, gaining unauthorized access to cloud services and causing potential damage, including data theft, deletion, or manipulation. This is often facilitated by weak authentication methods or social engineering tactics like phishing.
3. **Insider Threats:** Cloud providers have privileged access to the data stored on their systems. Malicious insiders—employees or contractors—could exploit their access to steal, alter, or leak sensitive information.
4. **Insecure APIs and Interfaces:** Cloud services often rely on APIs (Application Programming Interfaces) to interact with other applications. If these APIs are poorly designed or insecure, attackers can exploit vulnerabilities to gain unauthorized access to cloud resources or compromise cloud-based applications.

5. **Denial-of-Service (DoS) and DDoS Attacks:** Cloud services are susceptible to DoS and Distributed Denial-of-Service (DDoS) attacks, where malicious actors overload cloud resources with excessive traffic, causing legitimate users to be denied access to the services.

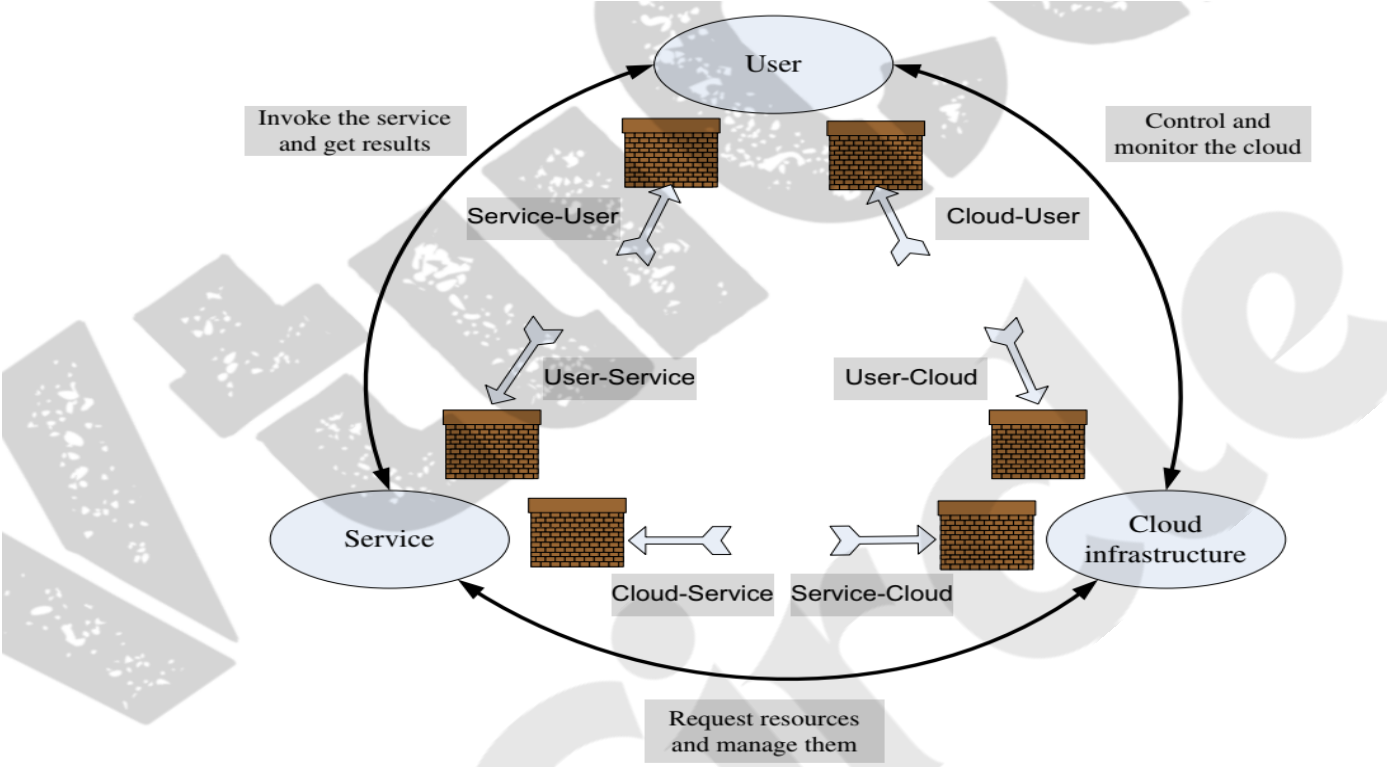


FIGURE 9.1

Surfaces of attacks in a cloud computing environment.

ii) Security: the top concern for cloud users

Security is the top concern for cloud users due to several factors:

1. **Loss of Control:** Users who are used to controlling their own systems and storing sensitive information behind a corporate firewall must extend trust to the Cloud Service Provider (CSP).
2. **Unauthorized Access and Data Theft:** Data in the cloud is vulnerable to unauthorized access and data theft, especially during storage. As data is kept in storage for extended periods, it becomes a prime target for cyberattacks, and threats can occur when users access or store sensitive information remotely.
3. **Risks from Insider Threats:** A major security risk arises from rogue employees of a CSP. Since users don't have visibility into the CSP's hiring practices or internal security measures, the potential for insider attacks is significant.
4. **Data Lifecycle and Deletion Issues:** Users cannot fully verify whether data that is meant to be deleted has been erased completely. Even if data is deleted, there is no guarantee that it has been properly wiped from storage media, and new users might be able to recover it.

5. **Multitenancy Risks:** In a multitenancy model, where multiple users share the same infrastructure, the security risks are amplified. If a vulnerability in the system affects one tenant, it could potentially compromise the data of others.

MODULE-5

9a. Explain the core components of Google app engine.

Ans: AppEngine is a platform for developing scalable applications accessible through the Web (see Figure 9.2). The platform is logically divided into four major components: infrastructure, the runtime environment, the underlying storage, and the set of scalable services that can be used to develop applications.

- **Infrastructure:** AppEngine hosts Web applications, and its primary function is to serve users requests efficiently. To do so, AppEngine's infrastructure takes advantage of many servers available within Google datacentres.
- **Runtime environment:** The runtime environment represents the execution context of applications hosted on AppEngine. With reference to the AppEngine infrastructure code, which is always active and running, the runtime comes into existence when the request handler starts executing and terminates once the handler has completed.

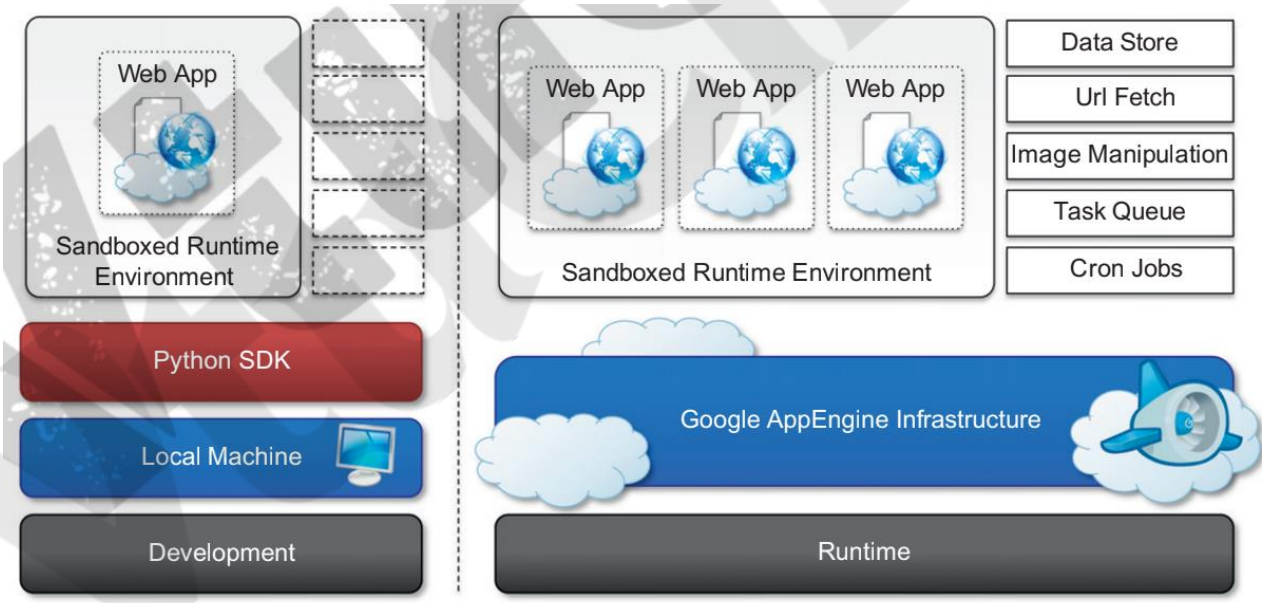


FIGURE 9.2

Google AppEngine platform architecture.

- **Sandboxing:** One of the major responsibilities of the runtime environment is to provide the application environment with an isolated and protected context in which it can execute without causing a threat to the server and without being influenced by other applications. In other words, it provides applications with a sandbox.

- **Supported runtimes:** Currently, it is possible to develop AppEngine applications using three different languages and related technologies: Java, Python, and Go. AppEngine currently supports Java 6, and developers can use the common tools for Web application development in Java, such as the Java Server Pages (JSP), and the applications interact with the environment by using the Java Servlet standard.
- **Storage** AppEngine provides various types of storage, which operate differently depending on the volatility of the data. There are three different levels of storage: in memory-cache, storage for semi structured data, and long-term storage for static data. In this section, we describe Datastore and the use of static file servers.
- **Static file servers:** Web applications are composed of dynamic and static data. Dynamic data are a result of the logic of the application and the interaction with the user. Static data often are mostly constituted of the components that define the graphical layout of the application (CSS files, plain HTML files, JavaScript files, images, icons, and sound files) or data files.
- **Datastore:** Datastore is a service that allows developers to store semi structured data. The service is designed to scale and optimized to quickly access data. Datastore can be considered as a large object database in which to store objects that can be retrieved by a specified key.
- **Application services:** Applications hosted on AppEngine take the most from the services made available through the runtime environment.
  - ✓ **UrlFetch:** Web 2.0 has introduced the concept of composite Web applications. Different resources are put together and organized as meshes within a single Web page. Meshes are fragments of HTML generated in different ways.
  - ✓ **MemCache:** AppEngine provides developers with access to fast and reliable storage, which is Datastore. Despite this, the main objective of the service is to serve as a scalable and long-term storage, where data are persisted to disk redundantly in order to ensure reliability and availability of data against failures.
  - ✓ **Mail and instant messaging:** Communication is another important aspect of Web applications. It is common to use email for following up with users about operations performed by the application. Email can also be used to trigger activities in Web applications.
  - ✓ **Account management:** Web applications often keep various data that customize their interaction with users. These data normally go under the user profile and are attached to an account.
- **Compute services:** Web applications are mostly designed to interface applications with users by means of a ubiquitous channel, that is, the Web. Most of the interaction is performed synchronously: Users navigate the Web pages and get instantaneous feedback in response to their actions.

- **Application life cycle:** AppEngine provides support for almost all the phases characterizing the life cycle of an application: testing and development, deployment, and monitoring. The SDKs released by Google provide developers with most of the functionalities required by these tasks.
- **Application development and testing:** Developers can start building their Web applications on a local development server. This is a self- contained environment that helps developers tune applications without uploading them to AppEngine.
- **Application deployment and management:** Once the application has been developed and tested, it can be deployed on AppEngine with a simple click or command-line tool. Before performing such task, it is necessary to create an application identifier, which will be used to locate the application from the Web browser by typing the address http:// , application-id..appspot.com.

9b. Discuss in detail the following media applications of cloud computing technologies. i) Animoto ii) Maya Rendering with Aneka iii) Video encoding on cloud.

Ans:

i) Animoto

Animoto2 is an example of a cloud-based media application that allows users to easily create videos by combining images, music, and video clips. Here's a detailed explanation of how it works and the infrastructure supporting it:

1. User Interface and Functionality

- **Video Creation**: Users can quickly create videos by selecting a theme, uploading photos and **videos**, arranging them in the desired sequence, and choosing a song for the soundtrack. The video is then rendered by Animoto's cloud system.
- **Automated Rendering**: A key feature of Animoto is the use of artificial intelligence (AI) to automatically select animation and transition effects based on the user's media. Users only need to organize their content in the desired order, and the system handles the rest.
- **Flexibility**: If users are not satisfied with the first rendering, they can re-render the video, and the AI engine will choose a different animation style, giving a different result each time.
- **Free and Paid Versions:** Animoto allows users to create 30-second videos for free. To create longer videos or access additional templates, users can opt for a paid subscription.

2. Infrastructure and Scalability

Animoto uses a robust and scalable cloud infrastructure primarily based on Amazon Web Services (AWS). Here's a breakdown of the components involved:

- **Amazon EC2**: Used for the web front-end and worker nodes. These instances run the application and handle tasks such as processing user requests and rendering videos.

- **Amazon S3:** Stores the user-uploaded content (photos, videos, and music) needed for creating videos.
- **Amazon SQS**: Manages communication between different components of the system, including the sending and receiving of rendering requests through message queues.

Auto-Scaling: The system uses Rightscale to monitor and manage the system's auto-scaling capabilities. Rightscale adjusts the number of worker nodes based on system load, ensuring that the infrastructure can grow or shrink dynamically to meet demand.

3. Process Flow

- **Front-End Nodes**: These nodes collect the user's photos, videos, and music, storing them in S3. When the user completes the storyboard, a video rendering request is placed in the SQS queue.
- **Worker Nodes**: These nodes pick up the rendering tasks from the SQS queue and begin the video rendering process. Once completed, they place a notification message in a different SQS queue, signalling the completion of the video.
- **User Notification**: After the rendering process finishes, the system notifies the user via email.

4. Scalability and Reliability

The scalable architecture of Animoto ensures that it can handle high demand during peak times by leveraging cloud resources efficiently. The infrastructure is designed to be reliable, with the ability to process large volumes of rendering requests without service interruptions.

ii) Maya Rendering with Aneka

Interesting applications of media processing are found in the engineering disciplines and the movie production industry. Operations such as rendering of models are now an integral part of the design workflow, which has become computationally demanding.

- **Rendering in Engineering**: Rendering is crucial in the design process, as engineers rely on high-quality 3D images to identify design flaws and refine prototypes. However, 3D rendering, especially when it involves many frames and cameras, is a time-consuming task that can span several days.
- **Cloud Computing for Engineering**: Cloud computing, particularly private cloud solutions, provide the computational power necessary for handling such intensive tasks. GoFront turned its local network of desktops into a private cloud infrastructure.
- **Implementation of Aneka Cloud**: The private cloud was powered by Aneka, a platform that helps manage cloud resources efficiently. Engineers use a specialized client interface to submit rendering jobs, which include the number of frames, cameras, and other parameters.
- **Optimizing Off-Peak Resources**: By utilizing off-peak hours (such as overnight) when desktops are otherwise idle, GoFront maximized resource usage without impacting the day-to-day operations of the department. This not only saves time but also ensures that resources are used efficiently.

- **Impact on Rendering Time**: The most significant result of adopting this private cloud solution is the drastic reduction in rendering time, from days to mere hours.

iii)Video encoding on cloud

Video encoding and transcoding are essential tasks for converting videos into formats suitable for playback on various devices and platforms. These processes are computationally intensive and require significant storage, which makes cloud computing an ideal solution due to its scalability and resource availability.

- **Cloud-Based Video Transcoding**: Video encoding and transcoding benefit from cloud technologies, offering scalability, computational power, and storage for processing videos efficiently.
- **Increased Demand**: The growing use of mobile devices and internet access has led to higher demand for video content, with various formats required for playback across different devices.
- **Encoding.com**: Encoding.com provides an on-demand video transcoding service, leveraging cloud technologies like Amazon Web Services (AWS) and Rackspace to process and store videos.
- **Access and Features**: Users can access the service via websites, APIs, desktop applications, and watched folders, with additional features like adding watermarks, logos, and converting audio and images.
- **Flexible Pricing**: Encoding.com offers various pricing models, including monthly subscriptions, pay-as-you-go options, and special pricing for high-volume users, serving over 2,000 customers and processing 10 million videos.

10a. Explain in detail about the application of cloud computing in i) Healthcare: ECG analysis in the cloud ii) Geoscience: satellite image processing

Ans:

i) Healthcare: ECG analysis in the cloud

Healthcare is a domain in which computer technology has found several and diverse applications: from supporting the business functions to assisting scientists in developing solutions to cure diseases.

- **Cloud-based ECG Monitoring**: Cloud technologies enable remote monitoring and analysis of ECG data, supporting doctors in providing more effective diagnostic processes. This system allows continuous monitoring of a patient's heart data without requiring them to visit the hospital.
- **Scalable and Elastic Infrastructure**: The cloud infrastructure is elastic, meaning it can scale dynamically based on demand. This flexibility ensures that hospitals and doctors can avoid costly investments in large computing systems, optimizing budgets.
- **Data Analysis and Alerts**: ECG data is transmitted from wearable devices to a cloud-hosted Web service for real-time analysis. The system detects abnormalities in the heart's waveform and notifies doctors and first-aid personnel immediately when intervention is needed.

- **Cost Efficiency**: Cloud services operate on a pay-per-use model, which reduces the financial burden of upfront investments. Pricing is based on actual service usage, offering flexible and cost-effective options for healthcare providers.
- **Ubiquity and Integration**: Cloud-based systems are accessible from any device via simple interfaces (like SOAP and REST). This ensures constant accessibility and facilitates integration with other hospital systems, making healthcare services more connected and reliable.

ii) Geoscience: satellite image processing

Geoscience applications collect, produce, and analyze massive amounts of geospatial and nonspatial data. As the technology progresses and our planet becomes more instrumented (i.e., through the deployment of sensors and satellites for monitoring), the volume of data that needs to be processed increases significantly.

- **Data Volume in Geoscience**: Geoscience applications generate vast amounts of geospatial and nonspatial data through technologies like sensors and satellites. The growth of this data requires advanced processing capabilities.
- **Role of GIS**: Geographic Information Systems (GIS) play a critical role in managing and analyzing geographically referenced data, which is essential for various fields such as agriculture, civil security, and natural resources management.
- **Cloud Computing for Geoscience**: Cloud computing is ideal for handling the massive data volumes in geoscience applications, offering scalable infrastructure to process and analyze data efficiently, supporting decision-making processes.
- **Satellite Data Processing**: Satellite remote sensing generates large amounts of raw image data that need intensive computation and data transfer. Cloud computing facilitates the processing of these images by providing the required infrastructure for storage and computational tasks.
- **Example: Department of Space, India**: The Department of Space, India, has developed a cloud-based solution using SaaS and PaaS to process satellite images. The system uses a Xen private cloud and the Aneka platform to dynamically scale resources and manage the workflow efficiently, offloading local computing facilities.

10b. Explain Amazon web services (AWS) in detail.

Ans: Amazon Web Services (AWS) is a platform that allows the development of flexible applications by providing solutions for elastic infrastructure scalability, messaging, and data storage. The platform is accessible through SOAP or RESTful Web service interfaces and provides a Web-based console where users can handle administration and monitoring of the resources required, as well as their expenses computed on a pay-as-you-go basis.

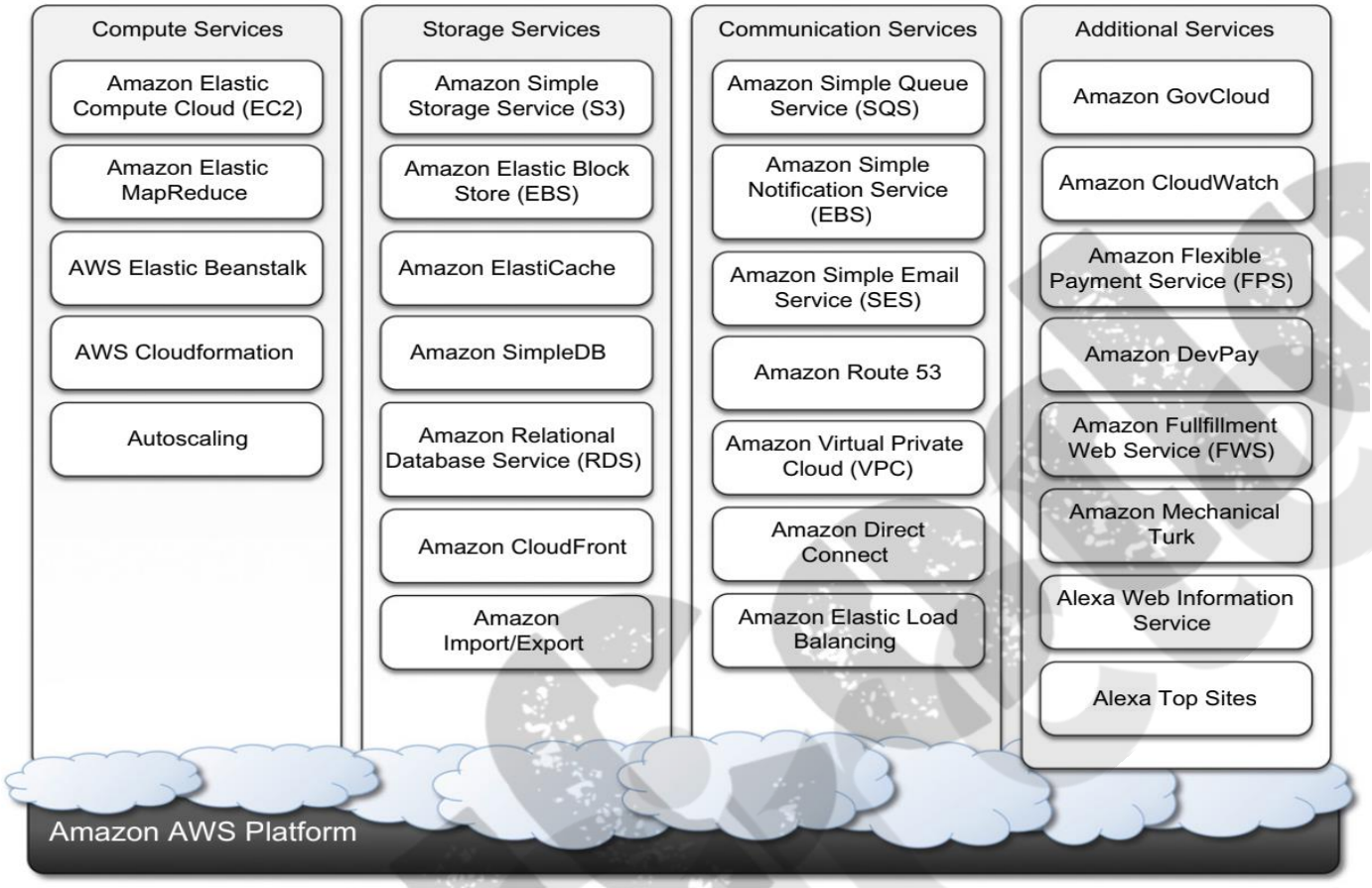


FIGURE 9.1

Amazon Web Services ecosystem.

Compute services: Compute services constitute the fundamental element of cloud computing systems. The fundamental service in this space is Amazon EC2, which delivers an IaaS solution that has served as a reference model for several offerings from other vendors in the same market segment. Amazon EC2 allows deploying servers in the form of virtual machines created as instances of a specific image. Images come with a preinstalled operating system and a software stack, and instances can be configured for memory, number of processors, and storage.

- **Amazon machine images:** Amazon Machine Images (AMIs) are templates from which it is possible to create a virtual machine. They are stored in Amazon S3 and identified by a unique identifier in the form of ami-xxxxxx and a manifest XML file.
- **EC2 instances:** EC2 instances represent virtual machines. They are created using AMI as templates, which are specialized by selecting the number of cores, their computing power, and the installed memory.
- **EC2 environment:** EC2 instances are executed within a virtual environment, which provides them with the services they require to host applications. The EC2 environment is in charge of allocating

addresses, attaching storage volumes, and configuring security in terms of access control and network connectivity.

- **Advanced compute services:** EC2 instances and AMIs constitute the basic blocks for building an IaaS computing cloud. On top of these, Amazon Web Services provide more sophisticated services that allow the easy packaging and deploying of applications and a computing platform that supports the execution of MapReduce-based applications.

Storage services: AWS provides a collection of services for data storage and information management. The core service in this area is represented by Amazon Simple Storage Service (S3). This is a distributed object store that allows users to store information in different formats.

- **S3 key concepts:** As the name suggests, S3 has been designed to provide a simple storage service that's accessible through a Representational State Transfer (REST) interface, which is quite similar to a distributed file system but which presents some important differences that allow the infrastructure to be highly efficient.
- **Amazon elastic block store:** The Amazon Elastic Block Store (EBS) allows AWS users to provide EC2 instances with persistent storage in the form of volumes that can be mounted at instance startup. They accommodate up to 1 TB of space and are accessed through a block device interface, thus allowing users to format them according to the needs of the instance they are connected to (raw storage, file system, or other).
- **Amazon ElastiCache:** ElastiCache is an implementation of an elastic in-memory cache based on a cluster of EC2 instances. It provides fast data access from other EC2 instances through a Memcached-compatible protocol so that existing applications based on such technology do not need to be modified and can transparently migrate to ElastiCache.
- **Structured storage solutions:** Enterprise applications quite often rely on databases to store data in a structured form, index, and perform analytics against it. Traditionally, RDBMS have been the common data back-end for a wide range of applications, even though recently more scalable and lightweight solutions have been proposed.
- **Amazon CloudFront:** CloudFront is an implementation of a content delivery network on top of the Amazon distributed storage infrastructure. It leverages a collection of edge servers strategically located around the globe to better serve requests for static and streaming Web content so that the transfer time is reduced as much as possible.

Communication services: Amazon provides facilities to structure and facilitate the communication among existing applications and services residing within the AWS infrastructure. These facilities can be organized into two major categories: virtual networking and messaging.

- **Virtual networking:** Virtual networking comprises a collection of services that allow AWS users to control the connectivity to and between compute and storage services. Amazon Virtual Private Cloud (VPC) and Amazon Direct Connect provide connectivity solutions in terms of infrastructure; Route 53 facilitates connectivity in terms of naming.
- **Messaging:** Messaging services constitute the next step in connecting applications by leveraging AWS capabilities. The three different types of messaging services offered are Amazon Simple Queue Service (SQS), Amazon Simple Notification Service (SNS), and Amazon Simple Email Service (SES).
- **Additional services:** Besides compute, storage, and communication services, AWS provides a collection of services that allow users to utilize services in aggregation. The two relevant services are Amazon CloudWatch and Amazon Flexible Payment Service (FPS).