2 Cloud Computing Architecture

2.1 Cloud Reference Model

- The cloud computing reference model is an abstract model that divides a cloud computing environment into abstraction layers and cross-layer functions to characterize and standardize its functions. This reference model divides cloud computing activities and functions into three cross-layer functions and five logical layers.
- Each of these layers describes different things that might be present in a cloud computing environment, such as computing systems, networking, storage equipment, virtualization software, security measures, control and management software, and so forth. It also explains the connections between these organizations. The five layers are the Physical layer, virtual layer, control layer, service orchestration layer, and service layer.
- The National Institute of Standards and Technology (NIST) is an organization designed by the US government (USG) agency for the adoption and development of cloud computing standards.
- The cloud computing reference model is a general high-level architecture and is meant for a cloud computing reference architecture provided, which outlines the primary performer/actor and the understanding of the cloud computing needs, uses, features, and standards. An overview of the NIST primary players, as indicated in the figure below. Each performer is an entity that might be a person or a cloud computing activity and role. The NIST cloud computing reference architecture identifies five organizations that take part in a transaction or process and complete duties in cloud computing.

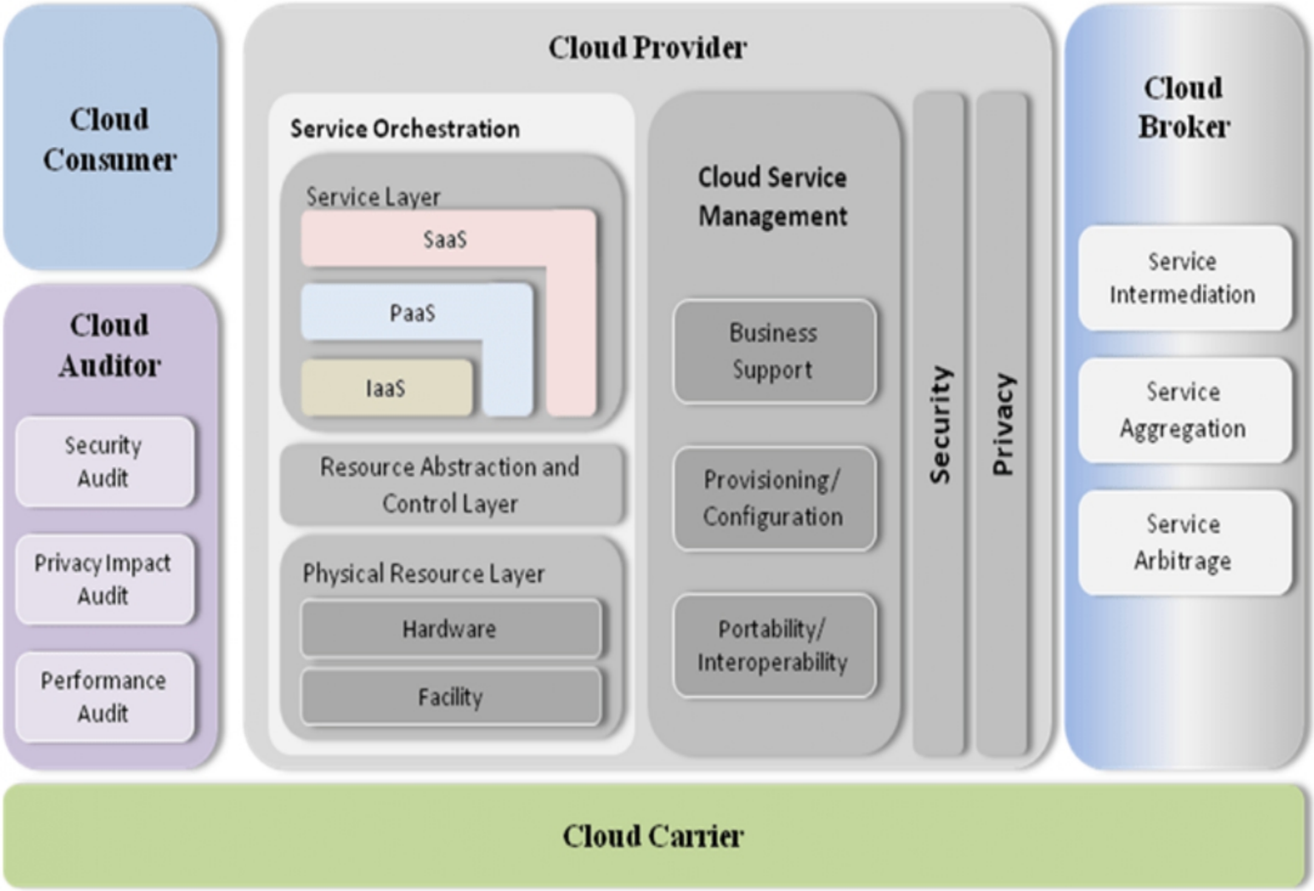


Figure 2-1: Cloud Reference Model

- **Cloud Consumer**

- The end user that the cloud computing service is designed to support is the cloud consumer. An individual or corporation with a working relationship with a cloud provider and utilizing its services is referred to as a cloud consumer. A cloud customer peruses a cloud provider's service catalog, makes the proper service request, enters into a service agreement with the cloud provider, and then utilizes the service. The cloud customer may be charged for the service provided, in which case payment arrangements must be made. They need to have a cloud Service Level Agreement (SLA).

- **Cloud Provider**

- Any individual, group, or other entity in charge of making a service accessible to cloud users is a cloud provider. A cloud provider creates the requested software, platforms, and infrastructure services, manages the technical infrastructure needed to supply the services, provisions the services at agreed-upon service levels, and safeguards the services' security and privacy.
- Through service interfaces and virtual network interfaces that aid in resource abstraction, the cloud provider implements the cloud software to make computing resources accessible to cloud consumers that use the infrastructure as a service.

- **Cloud Carrier**

- A cloud carrier serves as an intermediary between cloud providers and customers, facilitating connectivity and transport of cloud services. Customers can access the cloud through the network, telecommunication, and other access equipment provided by cloud carriers. Customers of cloud services, for instance, can get them through network access devices, including laptops, mobile phones, PCs, and mobile Internet devices (MIDs), among others. Network and telecommunication carriers typically handle the distribution of cloud services, while a transport agent is a company that arranges for the physical delivery of storage devices like high-capacity hard drives.
- Remember that a cloud provider will establish service level agreements (SLAs) with a cloud carrier to provide services at a level consistent with the SLAs offered to cloud consumers. The cloud provider may also demand that the cloud carrier provide dedicated and encrypted connections between cloud consumers and cloud providers.

- **Cloud Auditor**

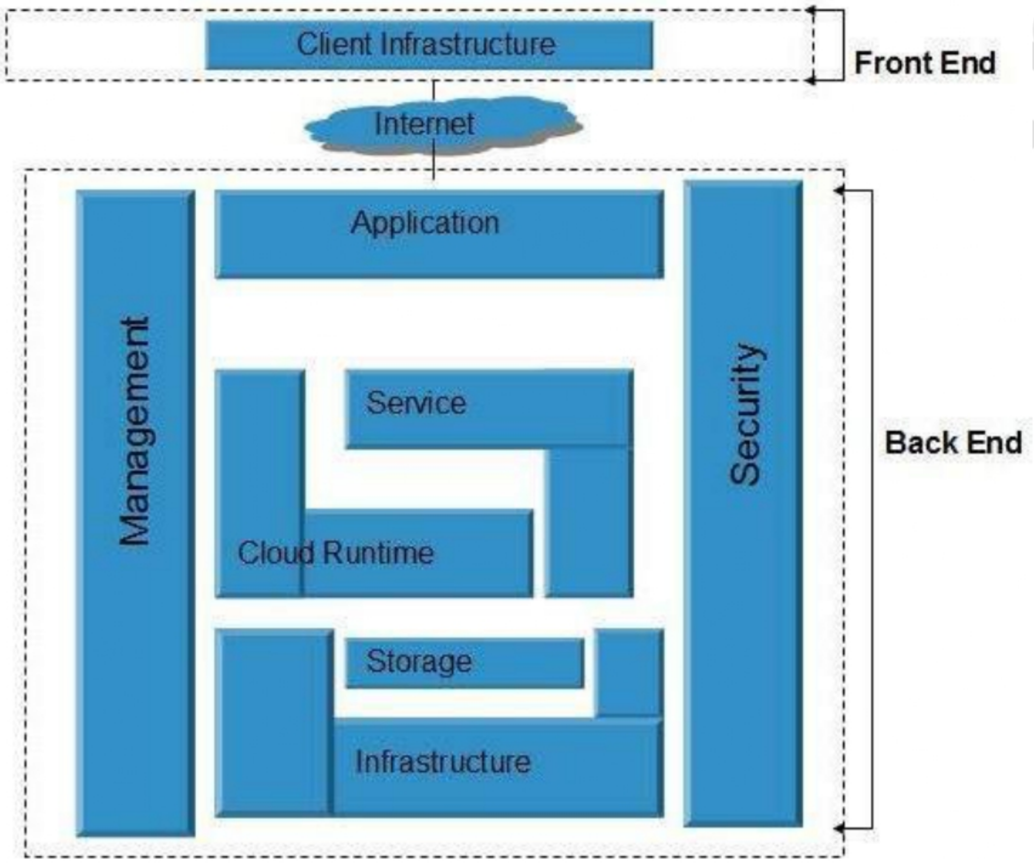
- An unbiased evaluation of cloud services, information system operations, performance, and the security of a cloud computing implementation can be done by a cloud auditor. A cloud auditor can assess a cloud provider's services in terms of performance, service level agreement compliance, privacy implications, and security controls.
- The management, operational, and technical precautions or countermeasures used inside an organizational information system to ensure the privacy, availability, and integrity of the system and its data are known as security controls.
- To do a security audit, a cloud auditor can evaluate the information system's security controls to see how well they are being implemented, functioning as intended, and achieving the required results in relation to the system's security needs. Verifying compliance with law and security policy should be part of the security audit.

- **Cloud Broker**

- An organization called a "Cloud Broker" controls how cloud services are used, performed, and delivered and negotiates contracts between cloud providers and cloud users. The integration of cloud services could become too difficult for cloud consumers to handle as cloud computing develops. Instead of contacting a cloud provider directly in certain circumstances, a cloud consumer may request cloud services through a cloud broker. A single point of access for controlling numerous cloud services is offered by cloud brokers. The capacity to offer a single consistent interface to numerous different providers, whether the interface is for commercial or technical objectives, separates a cloud broker from a cloud service provider. Cloud Brokers provide services in three categories.

2.2 Cloud Computing Architecture

- Cloud Computing architecture comprises of many cloud components, which are loosely coupled. We can broadly divide the cloud architecture into two parts:
  - Front End
  - Back End
- Each of the ends is connected through a network, usually Internet. The following diagram shows the graphical view of cloud computing architecture:



- **Front End**
- The front end refers to the client part of cloud computing system. It consists of interfaces and applications that are required to access the cloud computing platforms, Example - Web Browser.
  - It provides applications and the interfaces that are required for the cloud-based service.
  - It consists of client's side applications, which are web browsers such as Google Chrome and Internet Explorer.
  - Cloud infrastructure is the only component of the front-end.
  - Cloud infrastructure consists of hardware and software components such as data storage, server, virtualization software, etc.
  - It also provides a Graphical User Interface to the end-users to perform respective tasks.

- **Back End**
- The back End refers to the cloud itself. It consists of all the resources required to provide cloud computing services. It comprises of huge data storage, virtual machines, security mechanism, services, deployment models, servers, etc.
- It is responsible for monitoring all the programs that run the application on the front-end

- It has a large number of data storage systems and servers. The back-end is an important and huge part of the whole cloud computing architecture, as shown below:
- The components of the back-end cloud architecture are mentioned below.

  Application
  - It can either be a software or a platform
  - Depending upon the client requirement, the application provides the result to the enduser (with resources) in the back end.

  Service
  - Service is an essential component in cloud architecture
  - Its responsibility is to provide utility in the architecture
  - In a Cloud, few widely used services among the end-users are storage application development environments and web services
  - Service in backend refers to the major three types of cloud based services like SaaS, PaaS and IaaS. Also manages which type of service the user accesses.

  Runtime Cloud
  - Runtime cloud in backend provides the execution and Runtime platform/environment to the Virtual machine.

  Storage
  - It stores and maintains data like files, videos, documents, etc. over the internet
  - Some of the popular examples of storage services are below:
    - ✓ Amazon S3
    - ✓ Oracle Cloud-Storage
    - ✓ Microsoft Azure Storage
  - Its capacity varies depending upon the service providers available in the market

  Infrastructure
  - Cloud infrastructure includes hardware and software components such as servers, storage, network devices, virtualization software, and other storage resources that are needed to support the cloud computing model

  Management
  - Its task is to allot specific resources to a specific task; it simultaneously performs various functions of the cloud environment.
  - It helps in the management of components like application, task, service, security, data storage, and cloud infrastructure
  - In simple terms, it establishes coordination among the cloud resources

  Security
  - Security is an integral part of back-end cloud infrastructure
  - It provides secure cloud resources, systems, files, and infrastructure to end-users
  - Also, it implements security management to the cloud server with virtual firewalls which results in preventing data loss

Benefits of Cloud Computing Architecture

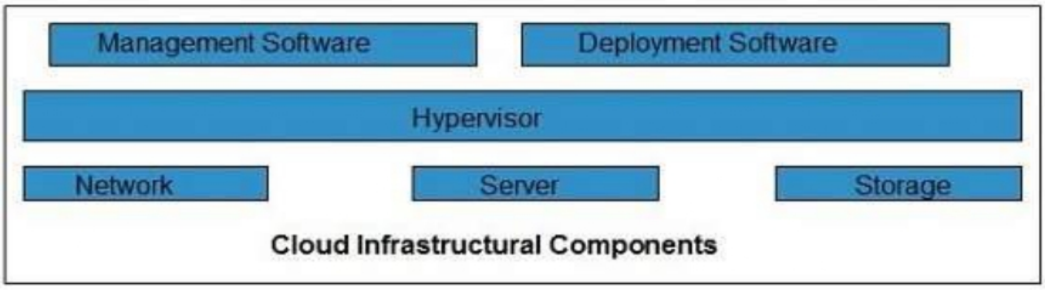
The cloud computing architecture is designed in such a way that:

- It solves latency issues and improves data processing requirements

- It reduces IT operating costs and gives good accessibility to access data and digital tools
- It helps businesses to easily scale up and scale down their cloud resources
- It has a flexibility feature which gives businesses a competitive advantage
- It results in better disaster recovery and provides high security
- It automatically updates its services
- It encourages remote working and promotes team collaboration

Cloud infrastructure

- Cloud infrastructure consists of servers, storage devices, network, cloud management software, deployment software, and platform virtualization.



Cloud Infrastructural Components

Hypervisor

- Hypervisor is a firmware or low-level program that acts as a Virtual Machine Manager. It allows to share the single physical instance of cloud resources between several tenants.
- It is a virtual machine monitor which provides Virtual Operating Platforms to every user
- It also manages guest operating systems in the cloud
- It runs a separate virtual machine on the back end which consists of software and hardware
- Its main objective is to divide and allocate resources

Management Software

- Its responsibility is to manage and monitor cloud operations with various strategies to increase the performance of the cloud
- Some of the operations performed by the management software are:
  - ✓ compliance auditing
  - ✓ management of overseeing disaster
  - ✓ contingency plans

Deployment Software

- It consists of all the mandatory installations and configurations required to run a cloud service
- All deployment of cloud services is performed using a deployment software
- The three different models which can be deployed are the following:

SaaS - Software as a service hosts and manages applications of the end-user.

Example: Gmail

PaaS - Platform as a service helps developers to build, create, and manage applications.

Example: Microsoft Azure

IaaS - Infrastructure as a service provides services on a pay-as-you-go pricing model

Network

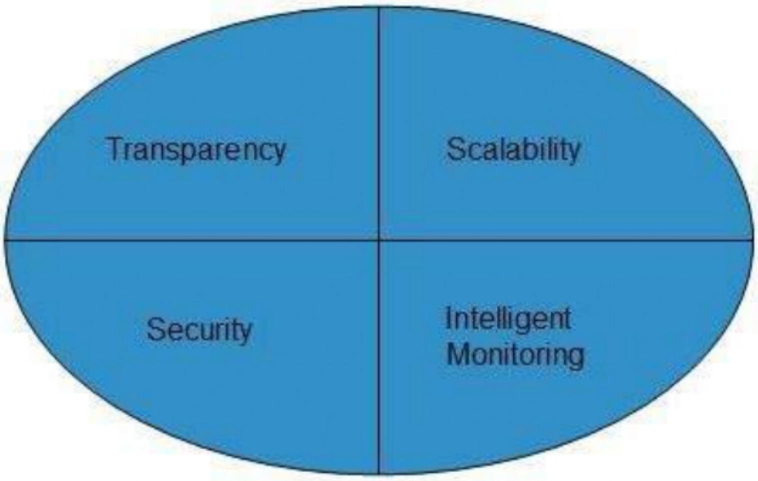
- It connects the front-end and back-end. Also, allows every user to access cloud resources
- It helps users to connect and customize the route and protocol
- It is a virtual server which is hosted on the cloud computing platform
- It is highly flexible, secure, and cost-effective

Cloud Storage

- Here, every bit of data is stored and accessed by a user from anywhere over the internet
- It is scalable at run-time and is automatically accessed
- Data can be modified and retrieved from cloud storage over the web

Infrastructural Constraints

Fundamental constraints that cloud infrastructure should implement are shown in the following diagram:



Transparency

- Virtualization is the key to share resources in cloud environment. But it is not possible to satisfy the demand with single resource or server. Therefore, there must be transparency in resources, load balancing and application, so that we can scale them on demand.

Scalability

- Scaling up an application delivery solution is not that easy as scaling up an application because it involves configuration overhead or even re-architecting the network. So, application delivery solution is need to be scalable which will require the virtual infrastructure such that resource can be provisioned and de-provisioned easily.

Intelligent Monitoring

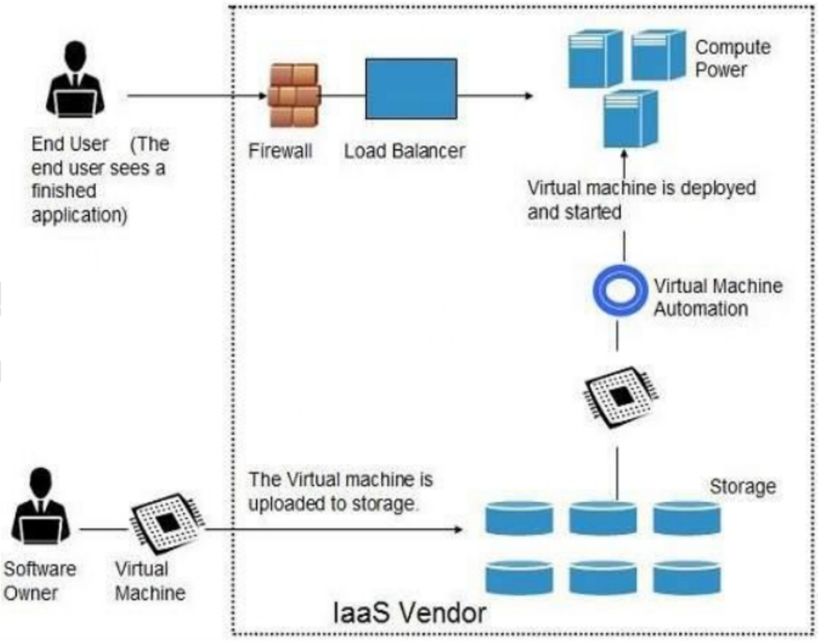
To achieve transparency and scalability, application solution delivery will need to be capable of intelligent monitoring.

Security

The mega data center in the cloud should be securely architected. Also the control node, an entry point in mega data center, also needs to be secure.

2.2.1 IaaS | Infrastructure as a Service

- Infrastructure-as-a-Service provides access to fundamental resources such as physical machines, virtual machines, virtual storage, etc. Apart from these resources, the IaaS also offers:
    - Virtual machine disk storage
    - Virtual local area network (VLANs)
    - Load balancers
    - IP addresses
    - Software bundles
- All of the above resources are made available to end user via server virtualization. Moreover, these resources are accessed by the customers as if they own them.
- Infrastructure as service or IaaS is the basic layer in cloud computing model.
- IaaS delivers customizable infrastructure on demand.
- IaaS examples can be categorized in two categories
    - IaaS Management layer
    - IaaS Physical infrastructure
- Some service providers provide both above categories and some provides only management layer.
- IaaS management layer also required integration with other IaaS solutions that provide physical infrastructure.
- Some examples:
    - Amazon Web Services (AWS)
    - Microsoft Azure
    - Google Compute Engine (GCE)

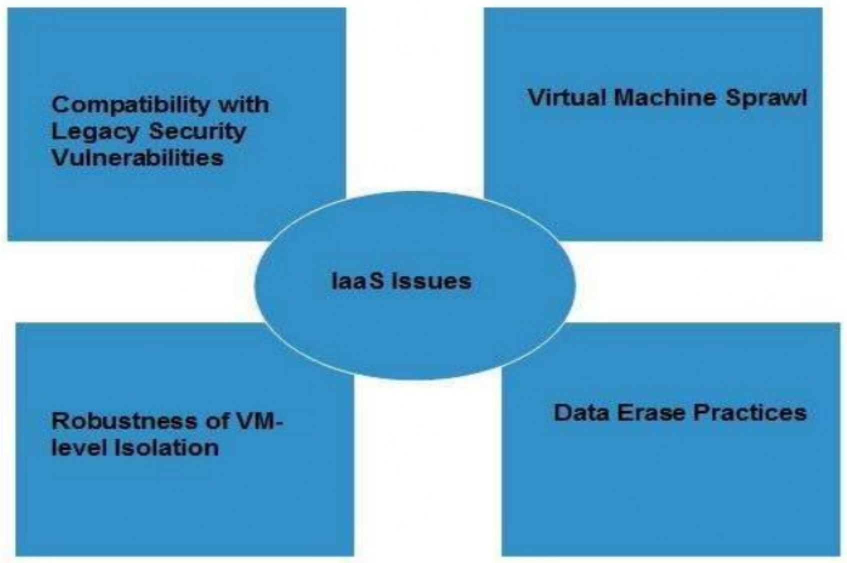

End User (The end user sees a finished application)
Firewall
Load Balancer
Compute Power
Virtual machine is deployed and started
Virtual Machine Automation
The Virtual machine is uploaded to storage.
Storage
Software Owner
Virtual Machine
IaaS Vendor

- **Benefits**
    - IaaS allows the cloud provider to freely locate the infrastructure over the Internet in a cost-effective manner. Some of the key benefits of IaaS are listed below:
    - Full control of the computing resources through administrative access to VMs.
    - Flexible and efficient renting of computer hardware.
    - Portability, interoperability with legacy applications.

- ✓ **Full control over computing resources through administrative access to VMs**
  - o IaaS allows the customer to access computing resources through administrative access to virtual machines in the following manner:
    - ▪ Customer issues administrative command to cloud provider to run the virtual machine or to save data on cloud server.
    - ▪ Customer issues administrative command to virtual machines they owned to start web server or to install new applications.
- ✓ **Flexible and efficient renting of computer hardware**
  - o IaaS resources such as virtual machines, storage devices, bandwidth, IP addresses, monitoring services, firewalls, etc. are made available to the customers on rent. The payment is based upon the amount of time the customer retains a resource. Also with administrative access to virtual machines, the customer can run any software, even a custom operating system.
  - o Portability, interoperability with legacy applications
  - o It is possible to maintain legacy between applications and workloads between IaaS clouds. For example, network applications such as web server or e-mail server that normally runs on customer-owned server hardware can also run from VMs in IaaS cloud.

Issues

- ➢ IaaS shares issues with PaaS and SaaS, such as Network dependence and browser based risks. It also has some specific issues, which are mentioned in the following diagram:


Compatibility with Legacy Security Vulnerabilities
Virtual Machine Sprawl
IaaS Issues
Robustness of VM-level Isolation
Data Erase Practices

Cloud Computing IaaS Issues

- o Compatibility with legacy security vulnerabilities
- o Because IaaS offers the customer to run legacy software in provider's infrastructure, it exposes customers to all of the security vulnerabilities of such legacy software.

Virtual Machine sprawl

- o The VM can become out-of-date with respect to security updates because IaaS allows the customer to operate the virtual machines in running, suspended and off state. However, the provider can automatically update such VMs, but this mechanism is hard and complex.

Robustness of VM-level isolation

- o IaaS offers an isolated environment to individual customers through hypervisor. Hypervisor is a software layer that includes hardware support for virtualization to split a physical computer into multiple virtual machines.

Data erase practices

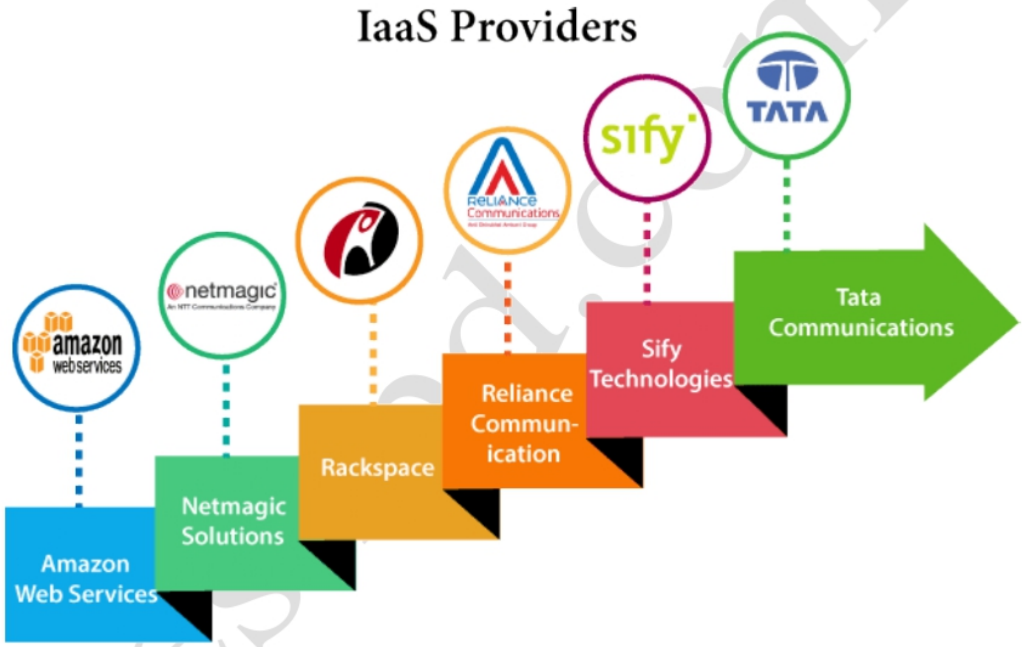
- The customer uses virtual machines that in turn use the common disk resources provided by the cloud provider. When the customer releases the resource, the cloud provider must ensure that next customer to rent the resource does not observe data residue from previous customer.

Characteristics

Here are the characteristics of IaaS service model:

- Virtual machines with pre-installed software.
- Virtual machines with pre-installed operating systems such as Windows, Linux, and Solaris.
- On-demand availability of resources.
- Allows to store copies of particular data at different locations.
- The computing resources can be easily scaled up and down.

Top Iaas Providers who are providing IaaS cloud computing platform



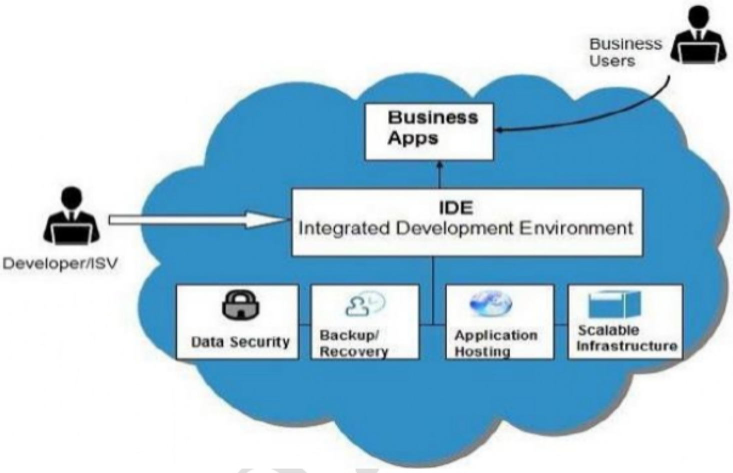
IaaS Vendor	Iaas Solution	Details
Amazon Web Services	Elastic, Elastic Compute Cloud (EC2) MapReduce, Route 53, Virtual Private Cloud, etc.	The cloud computing platform pioneer, Amazon offers auto scaling, cloud monitoring, and load balancing features as part of its portfolio.

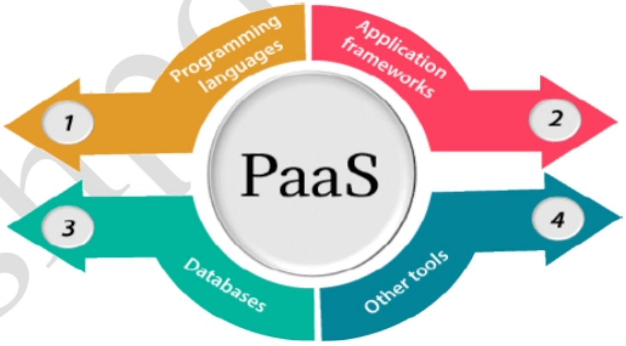
Netmagic Solutions	Netmagic IaaS Cloud	Netmagic runs from data centers in Mumbai, Chennai, and Bangalore, and a virtual data center in the United States. Plans are underway to extend services to West Asia.
Rackspace	Cloud servers, cloud files, cloud sites, etc.	The cloud computing platform vendor focuses primarily on enterprise-level hosting services.
Reliance Communications	Reliance Internet Data Center	RIDC supports both traditional hosting and cloud services, with data centers in Mumbai, Bangalore, Hyderabad, and Chennai. The cloud services offered by RIDC include IaaS and SaaS.
Sify Technologies	Sify IaaS	Sify's cloud computing platform is powered by HP's converged infrastructure. The vendor offers all three types of cloud services: IaaS, PaaS, and SaaS.
Tata Communications	InstaCompute	InstaCompute is Tata Communications' IaaS offering. InstaCompute data centers are located in Hyderabad and Singapore, with operations in both countries.

2.2.2 PaaS | Platform as a service

- PaaS provides a computing platform with a programming language execution environment.
- PaaS provide a development and deployment platform for running applications in the cloud.
- PaaS constitute the middleware on top of which applications are built.
- Application management is the core functionality of the middleware.
- PaaS provides run time environments for the applications.
- PaaS provides
    - Applications deployment
    - Configuring application components
    - Provisioning and configuring supporting technologies
- For users PaaS interfaces can be in the form of a Web-based interface or in the form of programming APIs and libraries.
- PaaS solutions generally include the infrastructure as well.
- Pure PaaS offered only the user-level middleware.

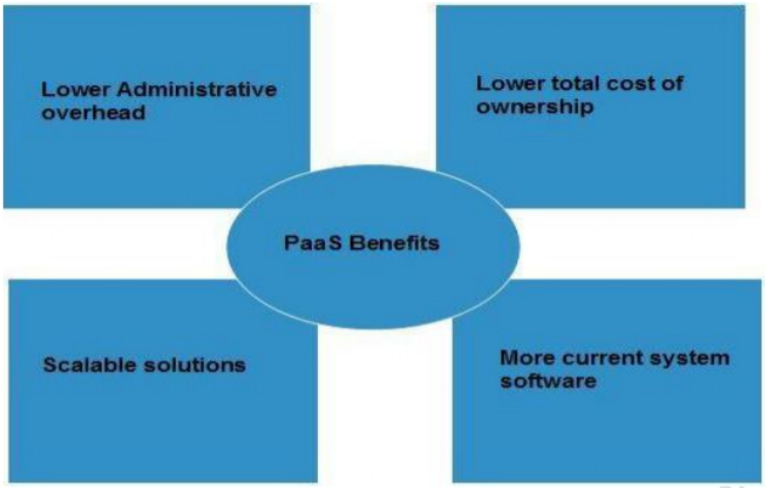
- PaaS offers the runtime environment for applications. It also offers development & deployment tools, required to develop applications.
- PaaS has a feature of point-and-click tools that enables non-developers to create web applications.
- Google's App Engine, Force.com are examples of PaaS offering vendors. Developer may log on to these websites and use the built-in API to create web-based applications.
- But the disadvantage of using PaaS is that the developer lock-in with a particular vendor. For example, an application written in Python against Google's API using Google's App Engine is likely to work only in that environment. Therefore, the vendor lock-in is the biggest problem in PaaS.
- The following diagram shows how PaaS offers an API and development tools to the developers and how it helps the end user to access business applications.





- PaasS classification:
  - PaaS-I: Runtime environment with Web-hosted application development platform. Rapid application prototyping. For example Force.com which is a combination of middleware and infrastructure product type.
  - PaaS-II: Runtime environment for scaling Web applications. The runtime could be enhanced by additional components that provide scaling capabilities. For example Google AppEngine which is a combination of middleware and infrastructure product type. Appscale is middlware product type.
  - PaaS-III: Middleware and programming model for developing distributed applications in the cloud. For example Microsoft Azure which is a combination of middleware and infrastructure product type. Manjrasoft Aneka is a middleware product type.
- Some examples:
  - Google App Engine
  - Force.com

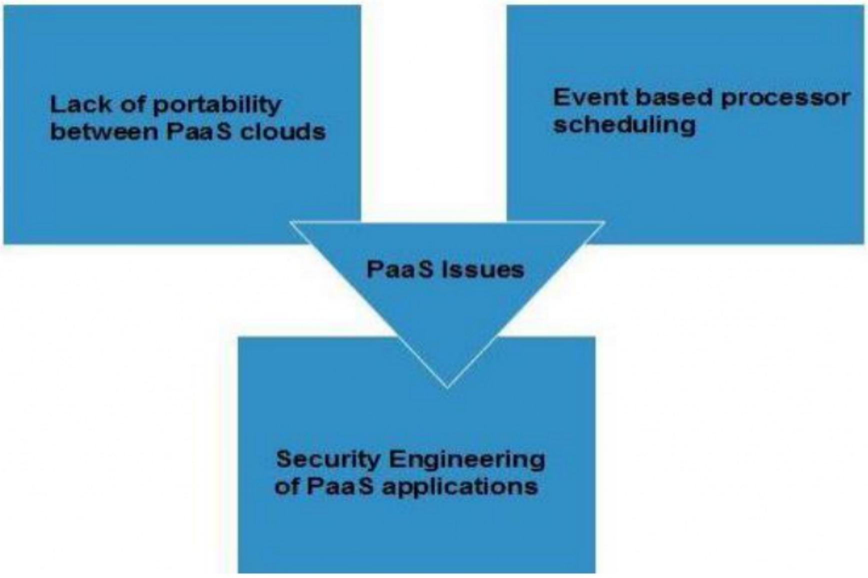
Benefits:


Lower Administrative overhead
Lower total cost of ownership
PaaS Benefits
Scalable solutions
More current system software

- ✓ **Lower administrative overhead**
  Consumer need not to bother much about the administration because it's the responsibility of cloud provider.
- ✓ **Lower total cost of ownership**
  Consumer need not purchase expensive hardware, servers, power and data storage.
- ✓ Scalable solutions
  It is very easy to scale up or down automatically based on application resource demands.
- ✓ **More current system software**
  It is the responsibility of the cloud provider to maintain software versions and patch installations

Issues

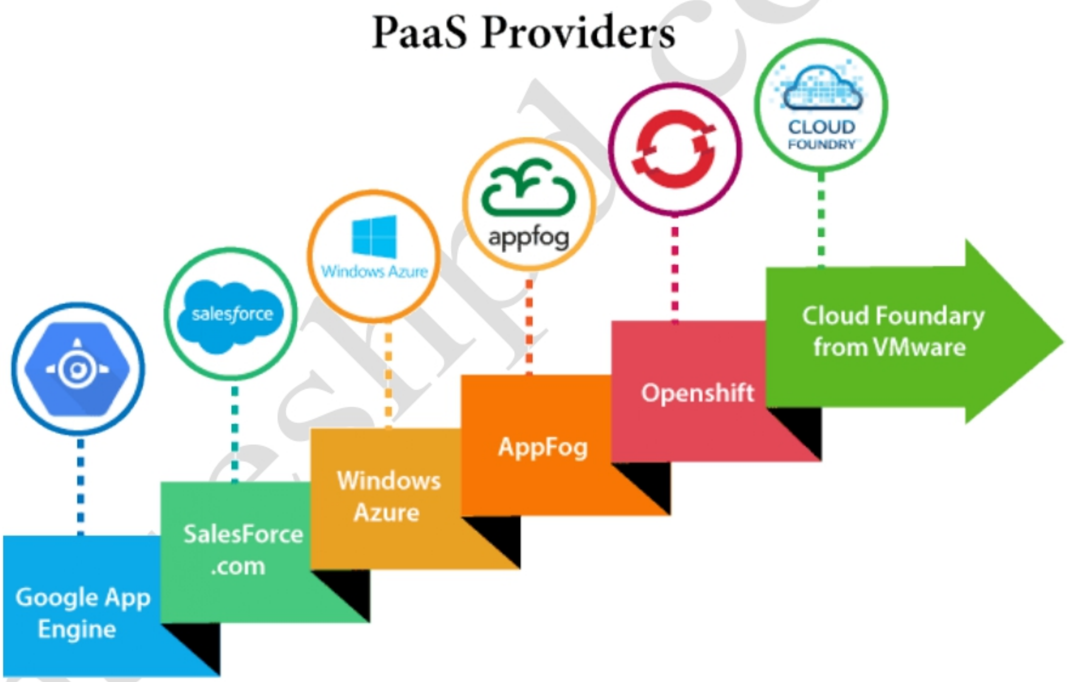
- ➢ Like SaaS, PaaS also place significant burdens on consumer's browsers to maintain reliable and secure connections to the provider systems. Therefore, PaaS shares many of the issues of SaaS.
- ➢ However, there are some specific issues associated with PaaS as shown in the following diagram:


Lack of portability between PaaS clouds
Event based processor scheduling
PaaS Issues
Security Engineering of PaaS applications

- ✓ **Lack of portability between paas clouds**
  Although standard languages are used yet the implementations of platforms services may vary. For example, file, queue, or hash table interfaces of one platform may differ from another, making it difficult to transfer workloads from one platform to another.
- ✓ **Event based processor scheduling**
  The PaaS applications are event oriented which poses resource constraints on applications, i.e., they have to answer a request in a given interval of time.
- ✓ **Security engineering of paas applications**
  Since the PaaS applications are dependent on network, PaaS applications must explicitly use cryptography and manage security exposures.

Characteristics

- ➢ PaaS offers browser based development environment. It allows the developer to create database and edit the application code either via Application Programming Interface or point-and-click tools.
- ➢ PaaS provides built-in security, scalability, and web service interfaces.
- ➢ PaaS provides built-in tools for defining workflow and approval processes and defining business rules.
- ➢ It is easy to integrate with other applications on the same platform.
- ➢ PaaS also provides web services interfaces that allow us to connect the applications outside the platform.

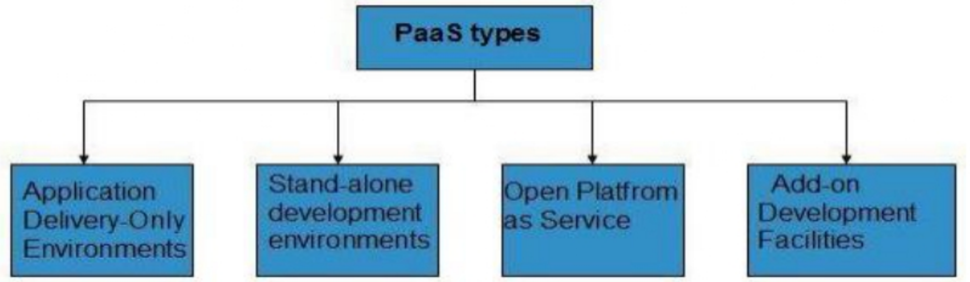


Providers	Services
Google App Engine (GAE)	App Identity, URL Fetch, Cloud storage client library, Logservice

Salesforce.com	Faster implementation, Rapid scalability, CRM Services, Sales cloud, Mobile connectivity, Chatter.
Windows Azure	Compute, security, IoT, Data Storage.
AppFog	Justcloud.com, SkyDrive, GoogleDocs
Openshift	RedHat, Microsoft Azure.

PaaS Types

- Based on the functions, the PaaS can be classified into four types as shown in the following diagram:


PaaS types
Application Delivery-Only Environments
Stand-alone development environments
Open Platfrom as Service
Add-on Development Facilities

- ✓ **Stand-alone development environments**
  The Stand-alone PaaS works as an independent entity for a specific function. It does not include licensing, technical dependencies on specific SaaS applications.
- ✓ **Application delivery-only environments**
  The Application Delivery PaaS includes on-demand scaling and application security.
- ✓ **Open platform as a service**
  Open PaaS offers open source software that helps a PaaS provider to run applications.
- ✓ **Add-on development facilities**
  The Add-on PaaS allows to customize the existing SaaS platform.

2.2.3 SaaS | Software as a service

- Software as a service (SaaS) allows users to connect to and use cloud-based apps over the Internet.
- SaaS is the service with which end users interact directly.
- It provides a means to free users from complex hardware and software management.
- In SaaS customer do not new to purchase the software and required the license.
- They simply access the application website, enter their credentials and billing details, and can instantly use the application.
- Customer can customize their software.
- Application is available to the customer on demand.
- SaaS can be considered as a "one-to-many" software delivery model.
- In SaaS applications are built as per the user needs.
- SaaS model allows to provide software application as a service to the end users. It refers to a software that is deployed on a hosted service and is accessible via Internet.

- Some of the SaaS applications are not customizable such as an Office Suite. But SaaS provides us Application Programming Interface (API), which allows the developer to develop a customized application.
- Some examples:
  - Gmail
  - Google drive
  - Dropbox

Benefits

- Using SaaS has proved to be beneficial in terms of scalability, efficiency, performance and much more. Some of the benefits are listed below:

✓ **Modest software tools**

The SaaS application deployment requires a little or no client side software installation which results in the following benefits:

- No requirement for complex software packages at client side
- Little or no risk of configuration at client side
- Low distribution cost

✓ **Efficient use of software licenses**

The client can have single license for multiple computers running at different locations which reduces the licensing cost. Also, there is no requirement for license servers because the software runs in the provider's infrastructure.

✓ **CENTRALIZED MANAGEMENT & DATA**

The data stored by the cloud provider is centralized. However, the cloud providers may store data in a decentralized manner for sake of redundancy and reliability.

✓ **PLATFORM RESPONSIBILITIES MANAGED BY PROVIDERS**

All platform responsibilities such as backups, system maintenance, security, hardware refresh, power management, etc., are performed by the cloud provider. The consumer need not to bother about them.

✓ **Multitenant solutions**

Multitenancy allows multiple users to share single instance of resources in virtual isolation. Consumers can customize their application without affecting the core functionality.

Issues

- There are several issues associated with SaaS, some of them are listed below:

✓ **Browser based risks**

If the consumer visits malicious website and browser becomes infected, and the subsequent access to SaaS application might compromise the consumer's data.To avoid such risks, the consumer can use multiple browsers and dedicate a specific browser to access SaaS applications or can use virtual desktop while accessing the SaaS applications.

✓ **Network dependence**

The SaaS application can be delivered only when network is continuously available. Also network should be reliable but the network reliability cannot be guaranteed either by cloud provider or the consumer.

✓ **Lack of portability between saas clouds**

Transferring workloads from one SaaS cloud to another is not so easy because work flow, business logics, user interfaces, support scripts can be provider specific.

Characteristics

Here are the characteristics of SaaS service model:

- SaaS makes the software available over the Internet.
- The Software are maintained by the vendor rather than where they are running.
- The license to the software may be subscription based or usage based. And it is billed on recurring basis.
- SaaS applications are cost effective since they do not require any maintenance at end user side.
- They are available on demand.
- They can be scaled up or down on demand.

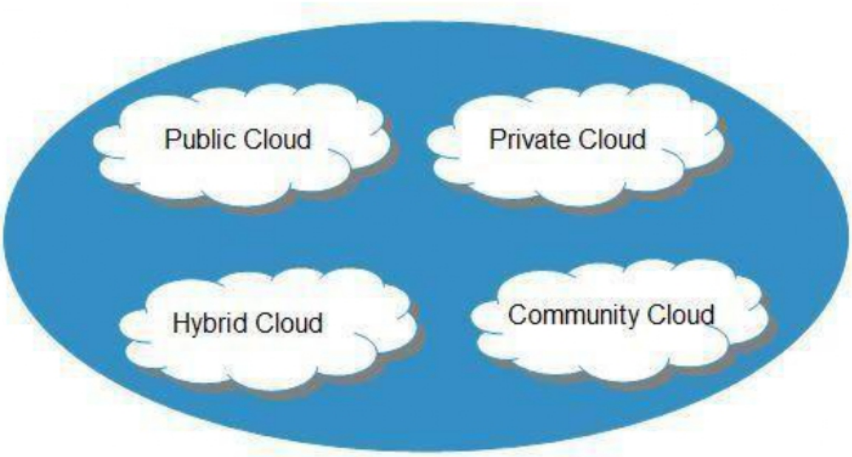
- They are automatically upgraded and updated.
- SaaS offers share data model. Therefore, multiple users can share single instance of infrastructure. It is not required to hard code the functionality for individual users.
- All users are running same version of the software.

SaaS Providers
Microsoft office365
Google Apps
GoTo Metting
Constant Contact
Oracle CRM
NetSuite
Workday, Inc
Salesforce .com

Provider	Services
Salseforce.com	On-demand CRM solutions
Microsoft Office 365	Online office suite
Google Apps	Gmail, Google Calendar, Docs, and sites
NetSuite	ERP, accounting, order management, CRM, Professionals Services Automation (PSA), and e-commerce applications.
GoToMeeting	Online meeting and video-conferencing software
Constant Contact	E-mail marketing, online survey, and event marketing
Oracle CRM	CRM applications
Workday, Inc	Human capital management, payroll, and financial management.

2.3 Deployment models

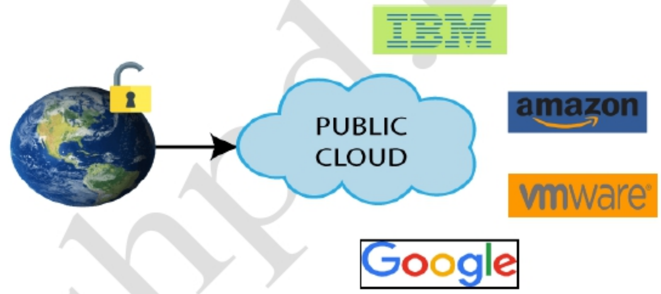
Deployment models define the type of access to the cloud, i.e., how the cloud is located? Cloud can have any of the four types of access: public, private, hybrid and community.

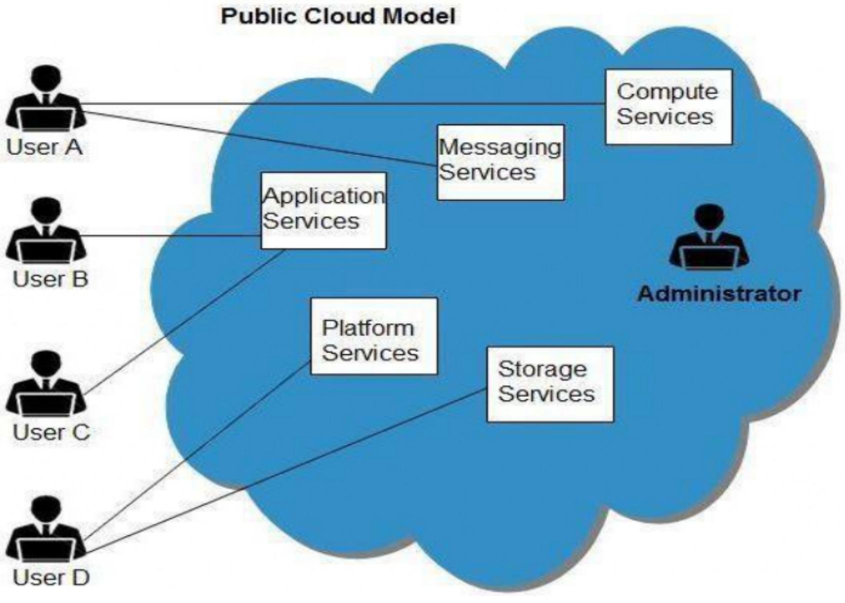


2.3.1 Public cloud:

The public cloud allows systems and services to be easily accessible to general public.

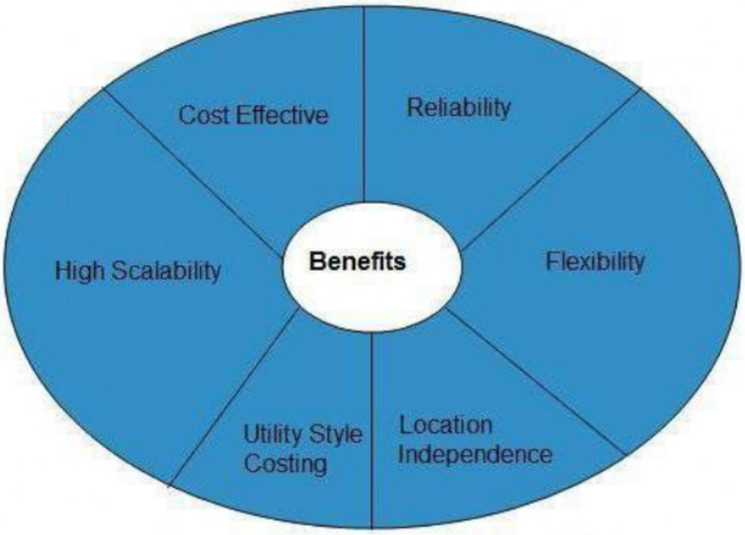
Amazon Elastic Compute Cloud (EC2), Microsoft Azure, IBM's Blue Cloud, Sun Cloud, and Google Cloud are examples of the public cloud.





Benefits

- There are many benefits of deploying cloud as public cloud model. The following diagram shows some of those benefits:



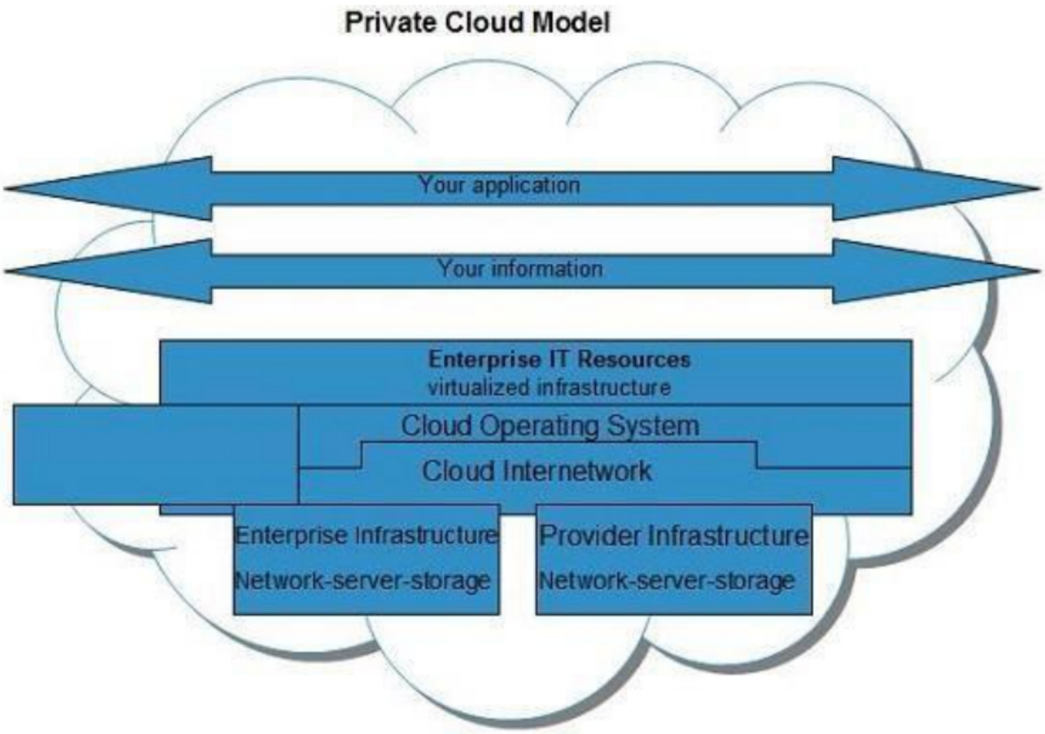
- ✓ **Cost effective**
  Since public cloud share same resources with large number of consumer, it has low cost.
- ✓ **Reliability**
  Since public cloud employs large number of resources from different locations, if any of the resource fail, public cloud can employ another one.
- ✓ **Flexibility**
  It is also very easy to integrate public cloud with private cloud, hence gives consumers a flexible approach.
- ✓ **Location independence**
  Since, public cloud services are delivered through internet, therefore ensures location independence.
- ✓ **Utility style costing**
  Public cloud is also based on pay-per-use model and resources are accessible whenever consumer needs it.
- ✓ **High scalability**
  Cloud resources are made available on demand from a pool of resources, i.e., they can be scaled up or down according the requirement.

Disadvantages

- Here are the disadvantages of public cloud model:
  - ✓ **Low security**
    In **public cloud model,** data is hosted off-site and resources are shared publicly, therefore does not ensure higher level of security.
  - ✓ **Less customizable**
    It is comparatively less customizable than private cloud

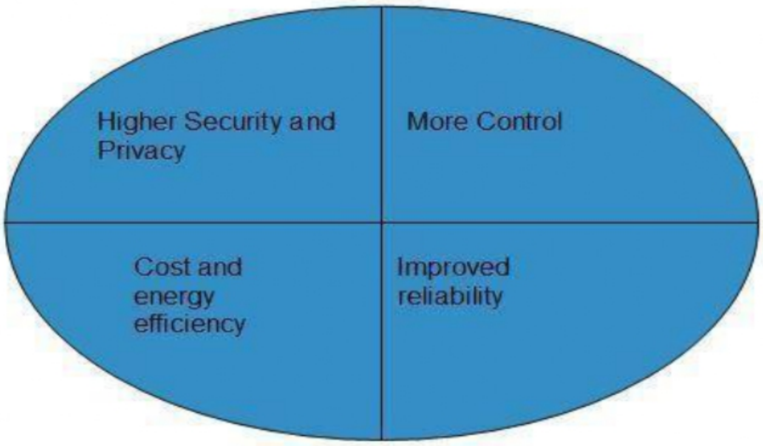
2.3.2 Private cloud:

- The private cloud allows systems and services to be accessible within an organization. The private cloud is operated only within a single organization. However, it may be managed internally or by third-party. The chief advantage of these systems is that the enterprise retains full control Over corporate data, security guidelines, and system performance:



- **Benefits**

  There are many benefits of deploying cloud as private cloud model. The following diagram shows some of those benefits:



- ✓ **Higher security and privacy**

  Private cloud operations are not available to general public and resources are shared from distinct pool of resources, therefore, ensures high security and privacy.
- ✓ **More control**

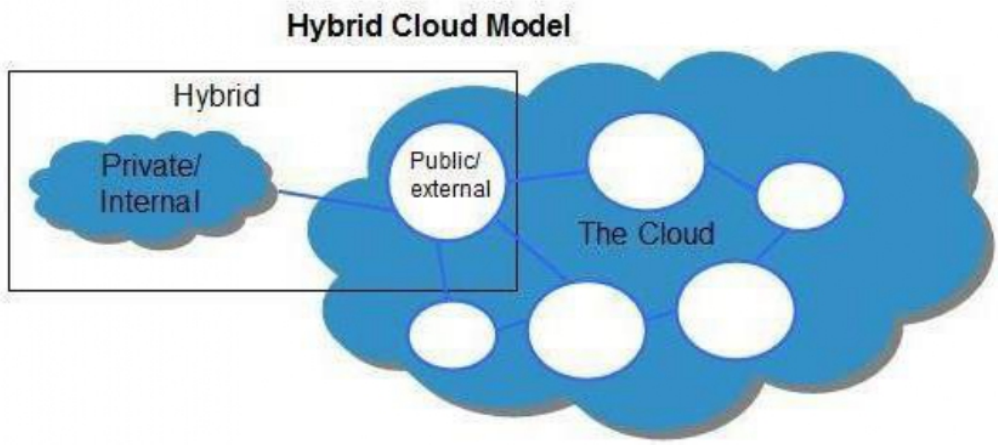
  Private clouds have more control on its resources and hardware than public cloud because it is accessed only within an organization.
- ✓ **Cost and energy efficiency**

  Private cloud resources are not as cost effective as public clouds but they offer more efficiency than public cloud.

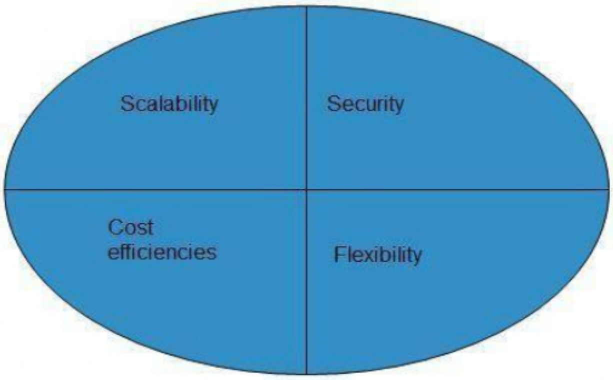
- **Disadvantages**
- Here are the disadvantages of using private cloud model:
- ✓ **Restricted area**
  Private cloud is only accessible locally and is very difficult to deploy globally.
- ✓ **Inflexible pricing**
  In order to fulfill demand, purchasing new hardware is very costly.
- ✓ **Limited scalability**
  Private cloud can be scaled only within capacity of internal hosted resources.
- ✓ **Additional skills**
  In order to maintain cloud deployment, organization requires more skilled and expertise.

2.3.3 Hybrid cloud

This can be a combination of private and public clouds that support the requirement to retain some data in an organization, and also the need to offer services in the cloud. A company may use internal resources in a private cloud and maintain total control over its proprietary data. It can then use a public cloud storage provider for backing up less sensitive information.



- **Benefits**
- There are many benefits of deploying cloud as hybrid cloud model. The following diagram shows some of those benefits:

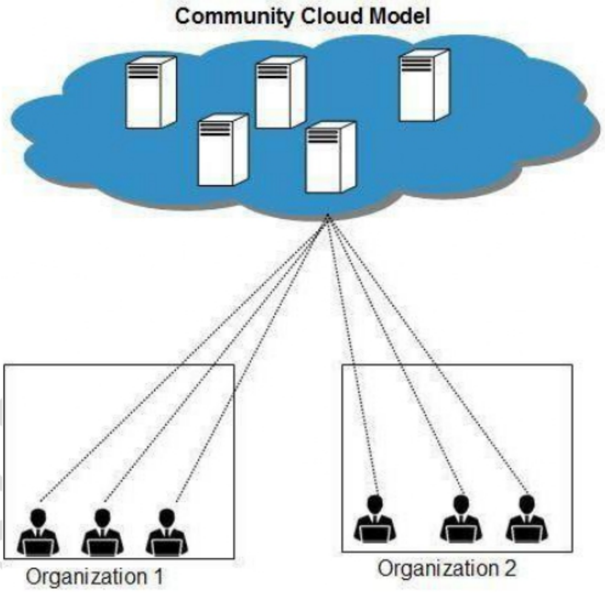


- ✓ **Scalability**
  It offers both features of public cloud scalability and private cloud scalability.
- ✓ **Flexibility**
  It offers both secure resources and scalable public resources

- ✓ **Cost efficiencies**
  Public cloud are more cost effective than private, therefore hybrid cloud can have this saving.
- ✓ **Security**
  Private cloud in hybrid cloud ensures higher degree of security.
- **Disadvantages**
- ✓ **Networking issues**
  Networking becomes complex due to presence of private and public cloud.
- ✓ **Security compliance**
  It is necessary to ensure that cloud services are compliant with organization's security policies.
- ✓ **Infrastructural dependency**
  The hybrid cloud model is dependent on internal it infrastructure, therefore it is necessary to ensure redundancy across data centers.

2.3.4 Community cloud:

- The community cloud allows system and services to be accessible by group of organizations. It shares the infrastructure between several organizations from a specific community. It may be managed internally or by the third-party.



- **Benefits**
  There are many benefits of deploying cloud as community cloud model. The following diagram shows some of those benefits:
- ✓ **Cost effective**
  Community cloud offers same advantage as that of public\ cloud at low cost. Sharing between organizations community cloud provides an infrastructure to share cloud resources and capabilities among several organizations.
- ✓ **Security**
  Community cloud is comparatively more secure than the public cloud.
- ✓ **Issues**
  - since all data is housed at one location, one must be careful in storing data in community cloud because it might be accessible by others.
  - It is also challenging to allocate responsibilities of governance, security and cost.

2.4 Service Oriented Architecture (SOA)

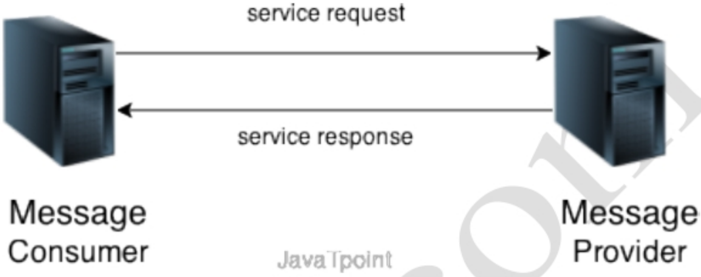
- A Service-Oriented Architecture or SOA is a design pattern which is designed to build distributed systems that deliver services to other applications through the protocol. It is only a concept and not limited to any programming language or platform.

Service

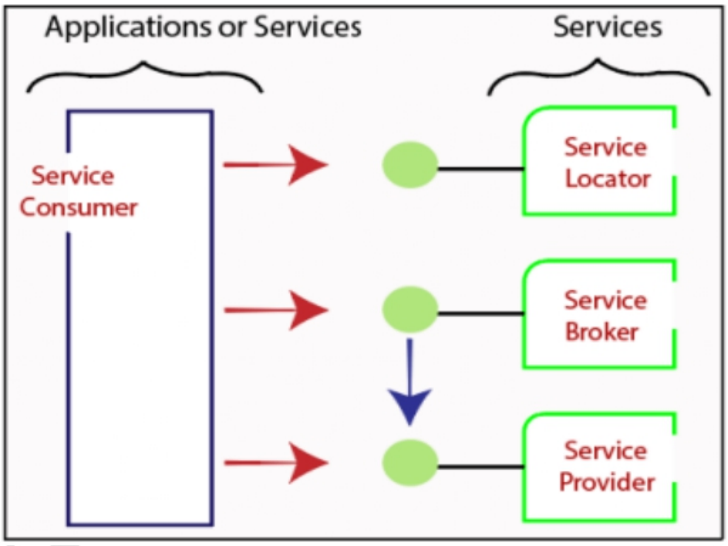
- A service is a well-defined, self-contained function that represents a unit of functionality. A service can exchange information from another service. It is not dependent on the state of another service. It uses a loosely coupled, message-based communication model to communicate with applications and other services.

Service Connections

- The figure given below illustrates the service-oriented architecture. Service consumer sends a service request to the service provider, and the service provider sends the service response to the service consumer. The service connection is understandable to both the service consumer and service provider.



Service-Oriented Terminologies



Services - The services are the logical entities defined by one or more published interfaces.

Service provider - It is a software entity that implements a service specification.

Service consumer - It can be called as a requestor or client that calls a service provider. A service consumer can be another service or an end-user application.

Service locator - It is a service provider that acts as a registry. It is responsible for examining service provider interfaces and service locations.

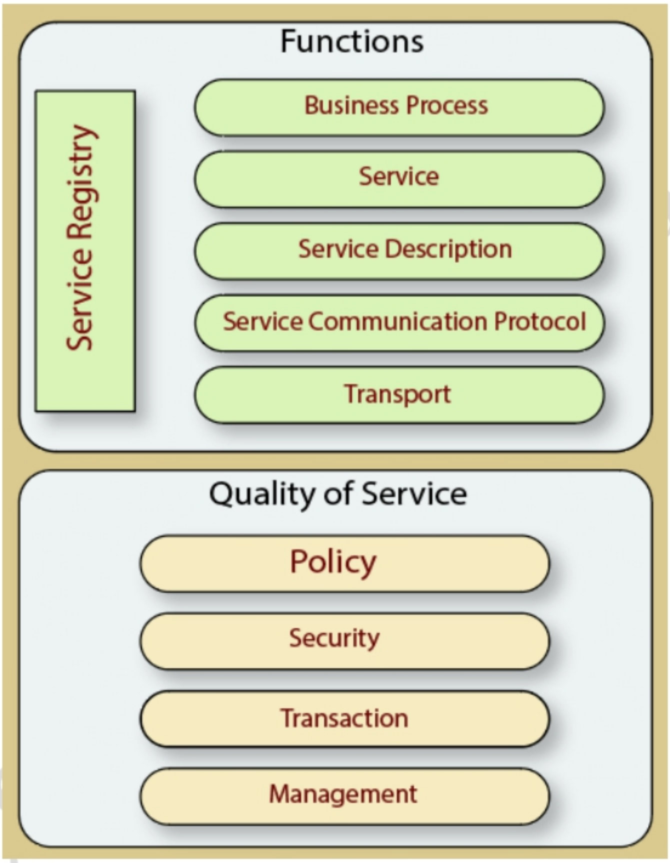
Service broker - It is a service provider that pass service requests to one or more additional service providers.

Characteristics of SOA

- The services have the following characteristics:
  - ✓ They are loosely coupled.
  - ✓ They support interoperability.
  - ✓ They are location-transparent
  - ✓ They are self-contained.

Components of service-oriented architecture

The service-oriented architecture stack can be categorized into two parts - functional aspects and quality of service aspects.



Functional aspects

The functional aspect contains:

- **Transport** - It transports the service requests from the service consumer to the service provider and service responses from the service provider to the service consumer.
- **Service Communication Protocol** - It allows the service provider and the service consumer to communicate with each other.
- **Service Description** - It describes the service and data required to invoke it.
- **Service** - It is an actual service.
- **Business Process** - It represents the group of services called in a particular sequence associated with the particular rules to meet the business requirements.
- **Service Registry** - It contains the description of data which is used by service providers

to publish their services.

- **Quality of Service aspects**

The quality-of-service aspects contains:

- **Policy** - It represents the set of protocols according to which a service provider makes and provide the services to consumers.
- **Security** - It represents the set of protocols required for identification and authorization.
- **Transaction** - It provides the surety of consistent result. This means, if we use the group of services to complete a business function, either all must complete or none of the complete.
- **Management** - It defines the set of attributes used to manage the services.

- **Advantages of SOA**

SOA has the following advantages:

- **Easy to integrate** - In a service-oriented architecture, the integration is a service specification that provides implementation transparency.
- **Manage Complexity** - Due to service specification, the complexities get isolated, and integration becomes more manageable.
- **Platform Independence** - The services are platform-independent as they can communicate with other applications through a common language.
- **Loose coupling** - It facilitates to implement services without impacting other applications or services.
- **Parallel Development** - As SOA follows layer-based architecture, it provides parallel development.
- **Available** - The SOA services are easily available to any requester.
- **Reliable** - As services are small in size, it is easier to test and debug them.

- **Practical applications of SOA:**

SOA is used in many ways around us whether it is mentioned or not.

- SOA infrastructure is used by many armies and air force to deploy situational awareness systems.
- SOA is used to improve the healthcare delivery.
- Nowadays many apps are games and they use inbuilt functions to run. For example, an app might need GPS so it uses inbuilt GPS functions of the device. This is SOA in mobile solutions.
- SOA helps maintain museums a virtualized storage pool for their information and content.

2.5 Security, trust, and privacy

Security, trust, and privacy issues are major obstacles for massive adoption of cloud computing. The traditional cryptographic technologies are used to prevent data tampering and access to sensitive information. The massive use of virtualization technologies exposes the existing system to new threats, which previously were not considered applicable.

For example, it might be possible that applications hosted in the cloud can process sensitive information; such information can be stored within a cloud storage facility using the most advanced technology in cryptography to protect data and then be considered safe from any attempt to access it without the required permissions. Although these data are processed in memory, they must necessarily be decrypted by the legitimate application, but since the application is hosted in a managed virtual environment it becomes accessible to the virtual machine manager that by program is designed to access the memory pages of such an application. In this case, what is experienced is a lack of control over the environment in which the application is executed, which is made possible by leveraging the cloud. It then happens that a new way of using existing technologies creates new opportunities for additional threats to the security of applications. The lack of control over their own data and processes also poses severe problems for the trust we give to the cloud service provider and the level of privacy we want to have for our data.

On one side we need to decide whether to trust the provider itself; on the other side, specific regulations can simply prevail over the agreement the provider is willing to establish with us concerning the privacy of the information managed on our behalf. Moreover, cloud services delivered to the end user can be the result of a complex stack of services that are obtained by third parties via the primary cloud service provider. In this case there is a chain of responsibilities in terms of service delivery that can introduce more vulnerability for the secure management of data, the enforcement of privacy rules, and the trust given to the service provider. In particular, when a violation of privacy or illegal access to sensitive information is detected, it could become difficult to identify who is liable for such violations. The challenges in this area are, then, mostly concerned with devising secure and trustable systems from different perspectives: technical, social, and legal.