Unit-1

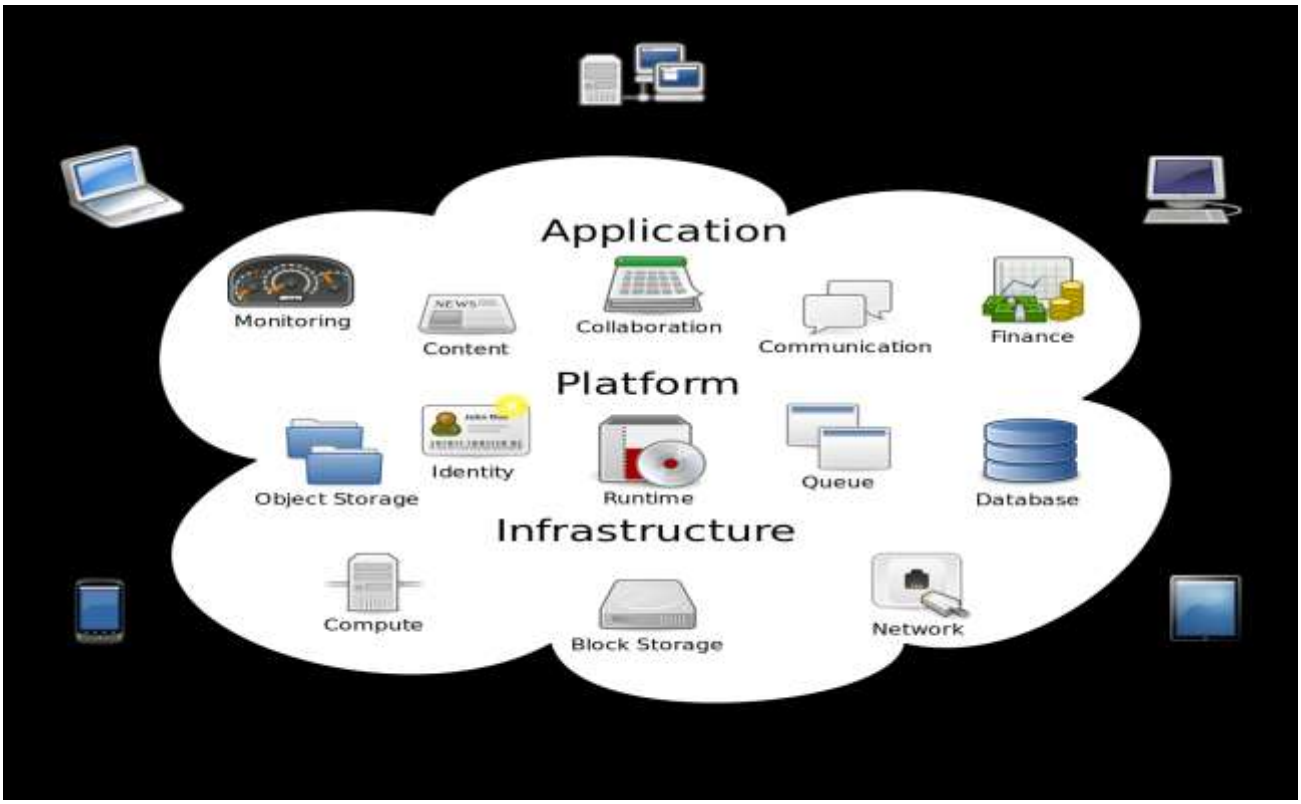
Define cloud computing

- The practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server or a personal computer. **Oxford dictionary**
- Cloud computing is a type of computing that relies on shared computing resources rather than having local servers or personal devices to handle applications. In its most simple description, cloud computing is taking services ("cloud services") and moving them outside an organization's firewall. **Webopedia**
- The term “cloud computing” is the cloud itself. The cloud is the large group of interconnected computers. These computers can be personal computers or network servers; they can be public or private**.** Michael miller

Examples of a Cloud Computing provider are:

- Google's Gmail.
- **Google Drive**: This is a pure **cloud computing** service, with all the storage found online so it can work with the **cloud** apps: **Google Docs**, **Google** Sheets, and **Google** Slides.

- Amazon EC2 — Virtual IT.
- Google App Engine — Application hosting.
- Google Apps and Microsoft Office Online — SaaS.
- Apple iCloud — Network storage.

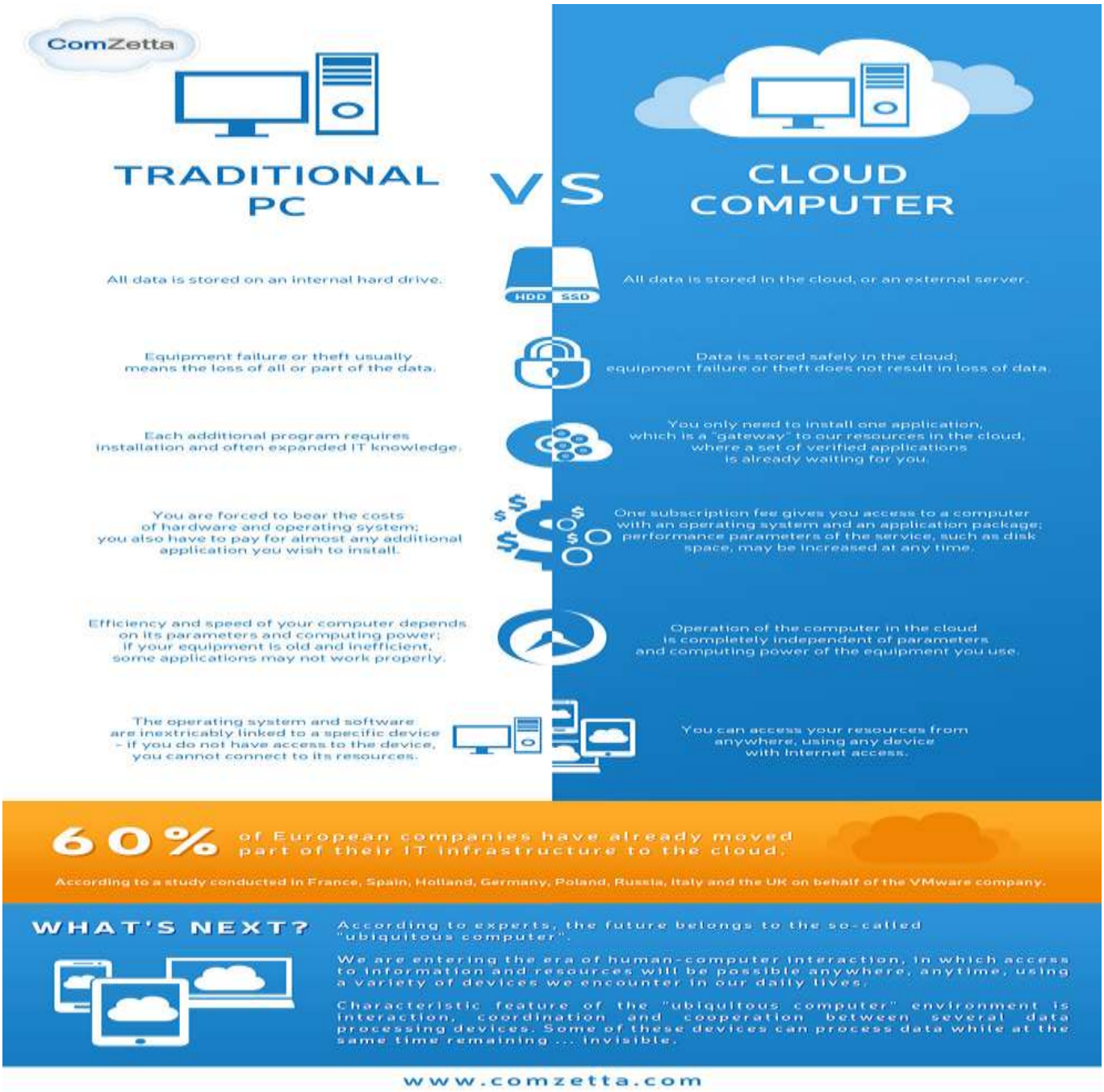


How Virtualisation is often confused with cloud computing?

Virtualisation is often confused with cloud computing, but it is actually the technology that underlies the cloud.

Virtualisation is software that divides up physical IT infrastructures to create virtual environments of dedicated resources. It helps to improve business performance and reduce IT costs by making it possible to run multiple applications and operating systems on the same server, simultaneously. Find out more on our Virtualisation page.

What is the difference between traditional pc and cloud computing?



Components of Cloud Computing

Components in a cloud refers to the platforms, like front end, back end and cloud based delivery and the network that used

i. Front End

The front end is the client part of Cloud Computing which uses as per the requirement of the user. Front-end comprises of the applications and the interfaces which help to access the cloud computing. Example- Browser or an app created by the company itself.

ii. Back End

The back end is a part which manages by the allotted authorities of the company and their back end has large data storage facilities, Virtual machines, security system, and servers. They are also engaged in traffic management along with security management.

The basic components of cloud computing in a simple topology are divided into 3 (three) parts, namely clients, datacenter, and distributed servers. The three basic components have specific goals and roles in running cloud computing operations. The concept of the three components can be described as follows:

- **Clients** on cloud computing architecture are said to be the exact same things that are plain, old, everyday local area networks (LANs). They are, typically, the computers that just sit on your desk. But they might also be laptops, tablet computers, mobile phones, or PDAs - all big drivers for cloud computing because of their mobility. Clients are interacting with to manage their information on the cloud.
- **Datacenter** is collection of servers where the application to which you subscribe is housed. It could be a large room in the basement of your building full of servers on the other side of the world that you access via the Internet. A growing trend in the IT world is virtualizing servers. That is, software can be installed allowing multiple instances of virtual servers to be used. In this way, you can have half a dozen virtual servers running on one physical server.
- **Distributed** Servers is a server placement in a different location. But the servers don't have to be housed in the same location. Often, servers are in geographically disparate locations.

But to you, the cloud subscribers, these servers act as if they're humming away right next to each other.

Another component of cloud computing is Cloud Applications cloud computing in terms of software architecture. So that the user does not need to install and run applications using a computer. Cloud Platform is a service in the form of a computing platform that contains hardware infrastructure and software. Usually have certain business applications and use services PaaS as its business application infrastructure. Cloud Storage involves processes delivering data storage as a service. Cloud Infrastructure is the delivery of computing infrastructure as a service.

Cloud Computing services have several components required, namely:

a. Cloud Clients, a computer or software specifically designed for the use of cloud computing based services.
Example :

- Mobile - Windows Mobile, Symbian
- Thin Client - Windows Terminal Service, CherryPal
- Thick Client - Internet Explorer, FireFox, Chrome

b. Cloud Services, products, services and solutions that are used and delivered real-time via internet media.
Example :

- Identity - OpenID, OAuth, etc.
- Integration - Amazon Simple Queue Service.
- Payments - PayPal, Google Checkout.
- Mapping - Google Maps, Yahoo! Maps.

c. Cloud Applications, applications that use Cloud Computing in software architecture so that users don't need to install but they can use the application using a computer.
Example :

- Peer-to-peer - BitTorrent, SETI, and others.
- Web Application - Facebook.
- SaaS - Google Apps, SalesForce.com, and others

d. Cloud Platform, a service in the form of a computing platform consisting of hardware and infrastructure software. This service is a service in the form of a computing platform which contains infrastructure hardware and software. Usually has an application certain businesses and use PaaS services as application infrastructure his business
Example :

- Web Application Frameworks - Python Django, Rubyon Rails, .NET
- Web Hosting
- Propietary - Force.com

e. Cloud Storage, involves the process of storing data as a service.
Example :

- Database - Google Big Table, Amazon SimpleDB.
- Network Attached Storage - Nirvanix CloudNAS, MobileMe iDisk.

f. Cloud Infrastructure, delivery of computing infrastructure as a service.
Example:

- Grid Computing - Sun Grid.
- Full Virtualization - GoGrid, Skytap.
- Compute - Amazon Elastic Compute Cloud

Shift from distributed computing to cloud computing:

SIX KEY CLOUD COMPUTING PRINCIPLES:

1. The Enablement Principle: Plan for cloud computing as a strategic enabler, rather than as an outsourcing arrangement or technical platform.

2. The Cost/Benefit Principle: Evaluate the benefits of cloud acquisition based on a full understanding of the costs of cloud compared with the costs of other technology platform business solutions.

3. The Enterprise Risk Principle: Take an enterprise risk management (ERM) perspective to manage the adoption and use of cloud.

4. The Capability Principle: Integrate the full extent of capabilities that cloud providers offer with internal resources to provide a comprehensive technical support and delivery solution.

5. The Accountability Principle: Manage accountabilities by clearly defining internal and provider responsibilities.

6. The Trust Principle: Make trust an essential part of cloud solutions, building trust into all business processes that depend on cloud computing.

//////The Five Principles of Cloud Computing Cloud computing is different from your traditional web service because of the principles behind cloud computing. These principles are

•**Resource pooling:**Cloud computing providers harness large economies of scale through resources pooling. They put together a vast network of servers and hard drives and apply the same set of configurations, protection and the works for them.

•**Virtualization:**Users do not have to care about the physical states of their hardware nor worry about hardware compatibility.

•**Elasticity:Addition**: of more hard disk space or server bandwidth can be done with just a few clicks of the mouse on-demand. Geographical scalability is also available in cloud computing - one can choose to replicate data to several data centres around the world.

•**Automatic/easy resource deployment:** The user only needs to choose the types and specifications of the resources he require and the cloud computing provider will configure and set them up automatically.

•**Metered billing:**Users are charged for only what they use.///////

Essential characteristics of Cloud Computing

- **On demand self services**: computer services such as email, applications, network or server service can be provided without requiring human interaction with each service

provider. Cloud service providers providing on demand self services include Amazon Web Services (AWS), Microsoft, Google, IBM and Salesforce.com. New York Times and NASDAQ are examples of companies using AWS (NIST). Gartner describes this characteristic as service based

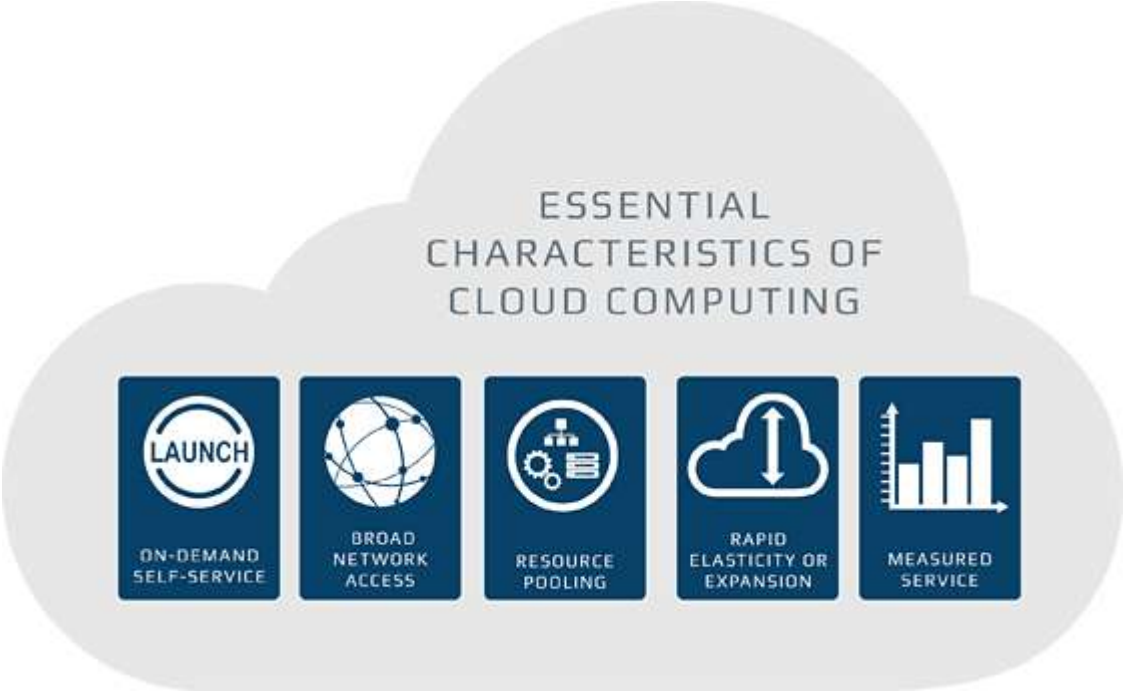
- **Broad network access**: Cloud Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms such as mobile phones, laptops and PDAs.

- **Resource pooling**: The provider's computing resources are pooled together to serve multiple consumers using multiple-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. The resources include among others storage, processing, memory, network bandwidth, virtual machines and email services. The pooling together of the resource builds economies of scale (Gartner).

- **Rapid elasticity**: Cloud services can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time.

- **Measured service**: Cloud computing resource usage can be measured, controlled, and reported providing transparency for both the provider and consumer of the utilised service. Cloud computing services use a metering capability which enables to control and optimise resource use. This implies that just like air time, electricity or municipality water IT services are charged per usage metrics – **pay per use**. The more you utilise the higher the bill. Just as utility companies sell power to subscribers, and telephone companies sell voice and data services, IT services such as network security management, data center hosting or even departmental billing can now be easily delivered as a contractual service.

- **Multi Tenacity**: is the 6th characteristics of cloud computing advocated by the Cloud Security Alliance. It refers to the need for policy-driven enforcement, segmentation, isolation, governance, service levels, and chargeback/billing models for different consumer constituencies. Consumers might utilize a public cloud provider's service offerings or actually be from the same organization, such as different business units rather than distinct organizational entities, but would still share infrastructure.



...../SaaS: Software as a Service

Software as a Service, also known as cloud application services, represents the most commonly utilized option for businesses in the cloud market. SaaS utilizes the internet to deliver applications, which are managed by a third-party vendor, to its users. A majority of SaaS applications run directly through your web browser, which means they do not require any downloads or installations on the client side.

SaaS Delivery

Due to its web delivery model, SaaS eliminates the need to have IT staff download and install applications on each individual computer. With SaaS, vendors manage all potential technical issues, such as data, middleware, servers, and storage, resulting in streamlined maintenance and support for the business.

SaaS Advantages

SaaS provides numerous advantages to employees and companies by greatly reducing the time and money spent on tedious tasks such as installing, managing, and upgrading software. This frees up plenty of time for technical staff to spend on more pressing matters and issues within the organization.

SaaS Characteristics

There are a few ways to help you determine when SaaS is being utilized:

- Managed from a central location
- Hosted on a remote server
- Accessible over the internet
- Users not responsible for hardware or software updates

When to Use SaaS

SaaS may be the most beneficial option in several situations, including:

- Startups or small companies that need to launch ecommerce quickly and don't have time for server issues or software
- Short-term projects that require quick, easy, and affordable collaboration
- Applications that aren't needed too often, such as tax software
- Applications that need both web and mobile access

SaaS Limitations and Concerns

- **Interoperability.** Integration with existing apps and services can be a major concern if the SaaS app is not designed to follow open standards for
- integration. In this case, organizations may need to design their own integration systems or reduce dependencies with SaaS services, which may not always be possible.
- **Vendor lock-in.** Vendors may make it easy to join a service and difficult to get out of it. For instance, the data may not be portable–technically or cost-effectively–across SaaS apps from other vendors without incurring significant cost or inhouse engineering rework. Not every vendor follows standard APIs, protocols, and tools, yet the features could be necessary for certain business tasks.
- **Lack of integration support.** Many organizations require deep integrations with on-premise apps, data, and services. The SaaS vendor may offer limited support in this regard, forcing organizations to invest internal resources in designing and managing integrations. The complexity of integrations can further limit how the SaaS app or other dependent services can be used.
- **Data security.** Large volumes of data may have to be exchanged to the backend data centers of SaaS apps in order to perform the necessary

software functionality. Transferring sensitive business information to public-cloud based SaaS service may result in compromised security and compliance in addition to significant cost for migrating large data workloads.

- **Customization.** SaaS apps offer minimal customization capabilities. Since a one-size-fits-all solution does not exist, users may be limited to specific functionality, performance, and integrations as offered by the vendor. In contrast, on-premise solutions that come with several software development kits (SDKs) offer a high degree of customization options.
- **Lack of control.** SaaS solutions involves handing control over to the third-party service provider. These controls are not limited to the software–in terms of the version, updates, or appearance–but also the data and governance. Customers may therefore need to redefine their data security and governance models to fit the features and functionality of the SaaS service.
- **Feature limitations.** Since SaaS apps often come in a standardized form, the choice of features may be a compromising tradeoff against security, cost, performance, or other organizational policies. Furthermore, vendor lock-in, cost, or security concerns may mean it's not viable to switch vendors or services to serve new feature requirements in the future.
- **Performance and downtime.** Because the vendor controls and manages the SaaS service, your customers now depend on vendors to maintain the service's security and performance. Planned and unplanned maintenance, cyber-attacks, or network issues may impact the performance of the SaaS app despite adequate service level agreement (SLA) protections in place.

Examples of SaaS

These are several popular examples of SaaS, including: Google GSuite (Apps), Dropbox, Salesforce, Cisco WebEx, SAP Concur, and GoToMeeting.

PaaS: Platform as a Service

Cloud platform services, also known as Platform as a Service (PaaS), provide cloud components to certain software while being used mainly for applications. PaaS delivers a framework for developers that they can build upon and use to create customized applications. All servers, storage, and networking can be managed by the enterprise or a third-party provider while the developers can maintain management of the applications.

PaaS Delivery

The delivery model of PaaS is similar to SaaS, except instead of delivering the software over the internet, PaaS provides a platform for software creation. This platform is delivered via the web, giving developers the freedom to concentrate on building the software without having to worry about operating systems, software updates, storage, or infrastructure.

PaaS allows businesses to design and create applications that are built into the PaaS with special software components. These applications, sometimes called middleware, are scalable and highly available as they take on certain cloud characteristics.

PaaS Advantages

No matter the size of your company, using PaaS offers numerous advantages, including:

- Simple, cost-effective development and deployment of apps
- Scalable
- Highly available
- Developers can customize apps without the headache of maintaining the software
- Significant reduction in the amount of coding needed
- Automation of business policy
- Easy migration to the hybrid model

PaaS Characteristics

PaaS has many characteristics that define it as a cloud service, including:

- Builds on virtualization technology, so resources can easily be scaled up or down as your business changes
- Provides a variety of services to assist with the development, testing, and deployment of apps
- Accessible to numerous users via the same development application
- Integrates web services and databases

When to Use PaaS

Utilizing PaaS is beneficial, sometimes even necessary, in several situations. For example, PaaS can streamline workflows when multiple developers are working on the same development project. If other vendors must be included, PaaS can provide great speed and flexibility to the entire process. PaaS is particularly beneficial if you need to create customized applications. This cloud service also can greatly reduce costs and it can simplify some challenges that come up if you are rapidly developing or deploying an app.

PaaS Limitations and Concerns

- **Data security.** Organizations can run their own apps and services using PaaS solutions, but the data residing in third-party, vendor-controlled cloud servers poses security risks and concerns. Your security options may be limited as customers may not be able to deploy services with specific hosting policies.
- **Integrations.** The complexity of connecting the data stored within an onsite data center or off-premise cloud is increased, which may affect which apps and services can be adopted with the PaaS offering. Particularly when not every component of a legacy IT system is built for the cloud, integration with existing services and infrastructure may be a challenge.
- **Vendor lock-in.** Business and technical requirements that drive decisions for a specific PaaS solution may not apply in the future. If the vendor has not provisioned convenient migration policies, switching to alternative PaaS options may not be possible without affecting the business.
- **Customization of legacy systems.** PaaS may not be a plug-and-play solution for existing legacy apps and services. Instead, several customizations and configuration changes may be necessary for legacy systems to work with the PaaS service. The resulting customization can result in a complex IT system that may limit the value of the PaaS investment altogether.
- **Runtime issues.** In addition to limitations associated with specific apps and services, PaaS solutions may not be optimized for the language and frameworks of your choice. Specific framework versions may not be available or perform optimally with the PaaS service. Customers may not be able to develop custom dependencies with the platform.
- **Operational limitation.** Customized cloud operations with management automation workflows may not apply to PaaS solutions, as the platform tends to limit operational capabilities for end users. Although this is intended to reduce the operational burden on end users, the loss of

operational control may affect how PaaS solutions are managed, provisioned, and operated.

Examples of PaaS

Popular examples of PaaS include AWS Elastic Beanstalk, Windows Azure, Heroku, Force.com, Google App Engine, and OpenShift.

IaaS: Infrastructure as a Service

Cloud infrastructure services, known as Infrastructure as a Service (IaaS), are made of highly scalable and automated compute resources. IaaS is fully self-service for accessing and monitoring computers, networking, storage, and other services. IaaS allows businesses to purchase resources on-demand and as-needed instead of having to buy hardware outright.

IaaS Delivery

IaaS delivers cloud computing infrastructure, including servers, network, operating systems, and storage, through virtualization technology. These cloud servers are typically provided to the organization through a dashboard or an API, giving IaaS clients complete control over the entire infrastructure. IaaS provides the same technologies and capabilities as a traditional data center without having to physically maintain or manage all of it. IaaS clients can still access their servers and storage directly, but it is all outsourced through a “virtual data center” in the cloud.

As opposed to SaaS or PaaS, IaaS clients are responsible for managing aspects such as applications, runtime, OSes, middleware, and data. However, providers of the IaaS manage the servers, hard drives, networking, virtualization, and storage. Some providers even offer more services beyond the virtualization layer, such as databases or message queuing.

IaaS Advantages

IaaS offers many advantages, including:

- The most flexible cloud computing model
- Easy to automate deployment of storage, networking, servers, and processing power
- Hardware purchases can be based on consumption
- Clients retain complete control of their infrastructure
- Resources can be purchased as-needed
- Highly scalable

IaaS Characteristics

Characteristics that define IaaS include:

- Resources are available as a service
- Cost varies depending on consumption
- Services are highly scalable
- Multiple users on a single piece of hardware
- Organization retain complete control of the infrastructure
- Dynamic and flexible

When to Use IaaS

Just as with SaaS and PaaS, there are specific situations when IaaS is most advantageous.

Startups and small companies may prefer IaaS to avoid spending time and money on purchasing and creating hardware and software. Larger companies may prefer to retain complete control over their applications and infrastructure, but they want to purchase only what they actually consume or need. Companies experiencing rapid growth like the scalability of IaaS, and they can change out specific hardware and software easily as their needs evolve. Anytime you are unsure of a new application's demands, IaaS offers plenty of flexibility and scalability.

IaaS Limitations and Concerns

Many limitations associated with SaaS and PaaS models – such as data security, cost overruns, vendor lock-in and customization issues – also apply to the IaaS model. Particular limitations to IaaS include:

- **Security.** While the customer is in control of the apps, data, middleware, and the OS platform, security threats can still be sourced from the host or other virtual machines (VMs). Insider threat or system vulnerabilities may expose data communication between the host infrastructure and VMs to unauthorized entities.
- **Legacy systems operating in the cloud.** While customers can run legacy apps in the cloud, the infrastructure may not be designed to deliver specific controls to secure the legacy apps. Minor enhancement to legacy apps may be required before migrating them to the cloud, possibly leading to new security issues unless adequately tested for security and performance in the IaaS systems.
- **Internal resources and training.** Additional resources and training may be required for the workforce to learn how to effectively manage the infrastructure. Customers will be responsible for data security, backup, and business continuity. Due to inadequate control into the infrastructure however, monitoring and management of the resources may be difficult without adequate training and resources available inhouse.
- **Multi-tenant security.** Since the hardware resources are dynamically allocated across users as made available, the vendor is required to ensure that other customers cannot access data deposited to storage assets by previous customers. Similarly, customers must rely on the vendor to ensure that VMs are adequately isolated within the multitenant cloud architecture.

Examples of IaaS

Popular examples of IaaS include DigitalOcean, Linode, Rackspace, Amazon Web Services (AWS), Cisco Metacloud, Microsoft Azure, and Google Compute Engine (GCE).//.....

//Types Of Cloud Computing Services

First let us go through the definition of each Cloud Computing Service type:

SaaS(Software-as-a-Service):

Saas provides clients with ability to use software applications over the internet via subscription basis. Clients can access applications from anywhere via web.

Examples: Google Applications and Salesforce.

PaaS(Platform-as-a-Service):

PaaS provides a platform where the clients can deploy their own applications and host them. The client is free from hassles of setting up infrastructure, managing storage, servers, network etc.

Examples: Amazon Web Services and Rackspace.

IaaS(Infrastructure-as-a-Service):

The IaaS provides just the hardware and network, the clients should install and develop software and applications.

Examples: IBM, Google and Amazon Web Services.

Now that we have gone through the definition, let us go ahead and understand each of these Cloud Computing Services in detail with the help of a **use case**. Consider a scenario where you have made travel plans. And you have decided **car as your mode of transport**. Now based on your requirements you have 4 options to choose from. Those are:

1. Take a taxi (SaaS)
2. Hire a car (PaaS)
3. Lease a car (IaaS)

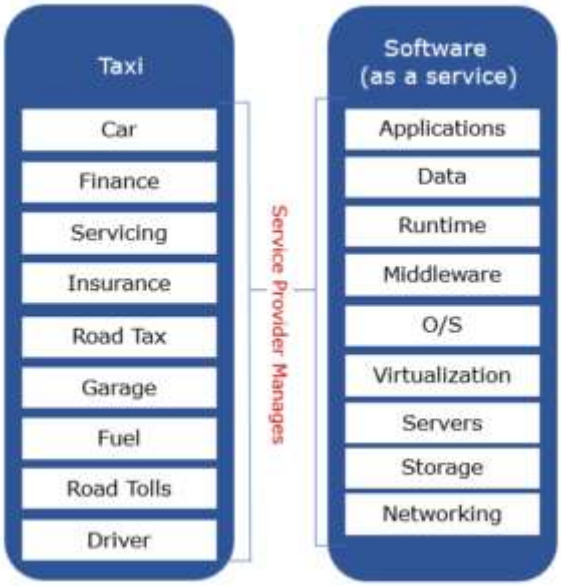
SaaS(Software-as-a-Service)

Use case: Suppose you choose to **take a taxi**, the car agency is responsible for car finance, servicing of the car. Besides that they take care of insurance and road tax. The driver, fuel requirements is taken care as well. You just need to pay for your ride.

Similarly Software-as-a-Service provider delivers software applications over the Internet, on demand and basically on a subscription basis. You just need to pay for the service you are utilizing. Entire software and hardware stack is hosted by the provider and made available to users over the Wide Area Network(WAN) like Internet or other dedicated networks.

SaaS eliminates the need for hardware acquisition, provisioning and maintenance, as well as software licensing, installation and support. Provides scalability, flexible payments and auto updates.

Examples: Google Applications like Gmail, Google Docs.



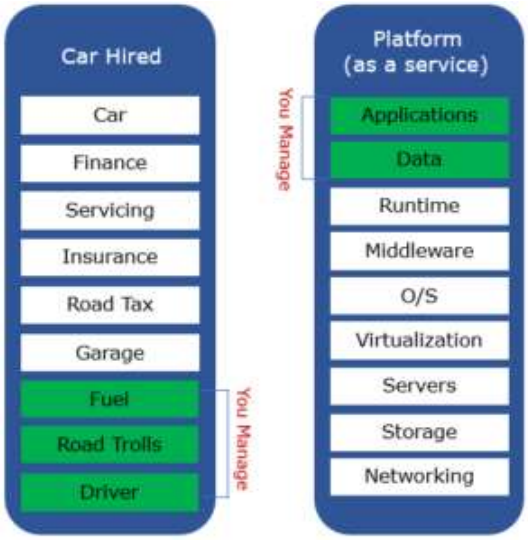
PaaS(Platform-as-a-Service)

Use case: You plan to travel to a nearby place so decided to **rent a car**, then you might have to take care of fuel needs, road tolls and hire a driver as well. Rest of the work like finance of the car, car service, insurance, road tax, garage etc is responsibility of the car renting agency.

Likewise Platform-as-a-Service provider offers core computing services like storage, virtualization and network. In addition, hosts OS, middleware frameworks or other development services such as web services, database management system and SD'kits compatible with various programming languages. The service provider builds and renders a secure and optimized environment on which users can install applications and data sets.

The prime benefits of this type of service include its simplicity and convenience for users–the Platform-as-a-Service users can focus on creating and running applications rather than constructing and maintaining underlying infrastructural stack and services.

Examples: Google app Engine, Microsoft Azure, Salesforce.



IaaS(Infrastructure-as-a-Service)

Use case: You made long travel plans to a far away place so chose to **lease a car.** Here you have to worry about servicing a car, road tax, insurance and garage requirements, pay for the fuel, road tolls and hire a driver. Most of the work is done by you. The car agency takes care of just the finance related to leasing a car.

Similarly Infrastructure-as-a-Service provider offers end users with bare computing resources like storage capacity, virtualization, networking, security and maintenance on a pay-as-you-use basis. The users are no longer concerned with location and purchase costs. Furthermore IaaS provider supplies additional services that complement the above features like load balancing, billing details, data backup, recovery and storage.

IaaS model users handle most of the workload like installing, maintaining and managing software layers.

Example: Amazon AWS, Rackspace, Flexiscale and Google Cloud Platform are some well known IaaS providers.//.....

Developing cloud services: saas, iaas, paas:

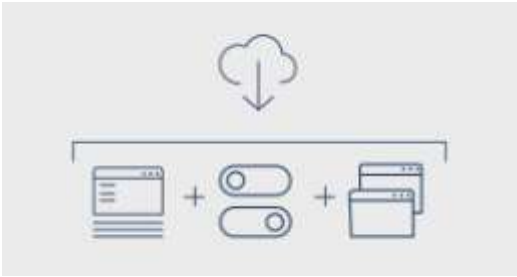
Common Examples of SaaS, PaaS, & IaaS

Platform Type	Common Examples
SaaS	Google Apps, Dropbox, Salesforce, Cisco WebEx, Concur, GoToMeeting
PaaS	AWS Elastic Beanstalk, Windows Azure, Heroku, Force.com, Google App Engine, Apache Stratos, OpenShift
IaaS	DigitalOcean, Linode, Rackspace, Amazon Web Services (AWS), Cisco Metapod, Microsoft Azure, Google Compute Engine (GCE)

DEFINING IAAS, PAAS AND SAAS

Software as a service (SaaS)

A service provider delivers software and applications through the internet. Users subscribe to the software and access it via the web or vendor APIs.



Platform as a service (PaaS)

A service provider offers access to a cloud-based environment in which users can build and deliver applications. The provider supplies underlying infrastructure.

Infrastructure as a service (IaaS)

A vendor provides clients pay-as-you-go access to storage, networking, servers and other computing resources in the cloud.

SaaS

Software as a service (SaaS) is a cloud computing offering that provides users with access to a vendor's cloud-based software. Users do not install applications on their local devices. Instead, the applications reside on a remote cloud network accessed through the web or an API. Through the application, users can store and analyze data and collaborate on projects.

Key features

- SaaS vendors provide users with software and applications via a subscription model.
- Users do not have to manage, install or upgrade software; SaaS providers manage this.
- Data is secure in the cloud; equipment failure does not result in loss of data.
- Use of resources can be scaled depending on service needs.
- Applications are accessible from almost any internet-connected device, from virtually anywhere in the world.

Examples of SaaS

These are several popular examples of SaaS, including: Google GSuite (Apps), Dropbox, Salesforce, Cisco WebEx, SAP Concur, and GoToMeeting.

PaaS

Platform as a service (PaaS) is a cloud computing offering that provides users with a cloud environment in which they can develop, manage and deliver applications. In addition to storage and other computing resources, users are able to use a suite of prebuilt tools to develop, customize and test their own applications.

Key features

- PaaS provides a platform with tools to test, develop and host applications in the same environment.
- Enables organizations to focus on development without having to worry about underlying infrastructure.
- Providers manage security, operating systems, server software and backups.
- Facilitates collaborative work even if teams work remotely.

Examples of PaaS

Popular examples of PaaS include AWS Elastic Beanstalk, Windows Azure, Heroku, Force.com, Google App Engine, and OpenShift.

IaaS

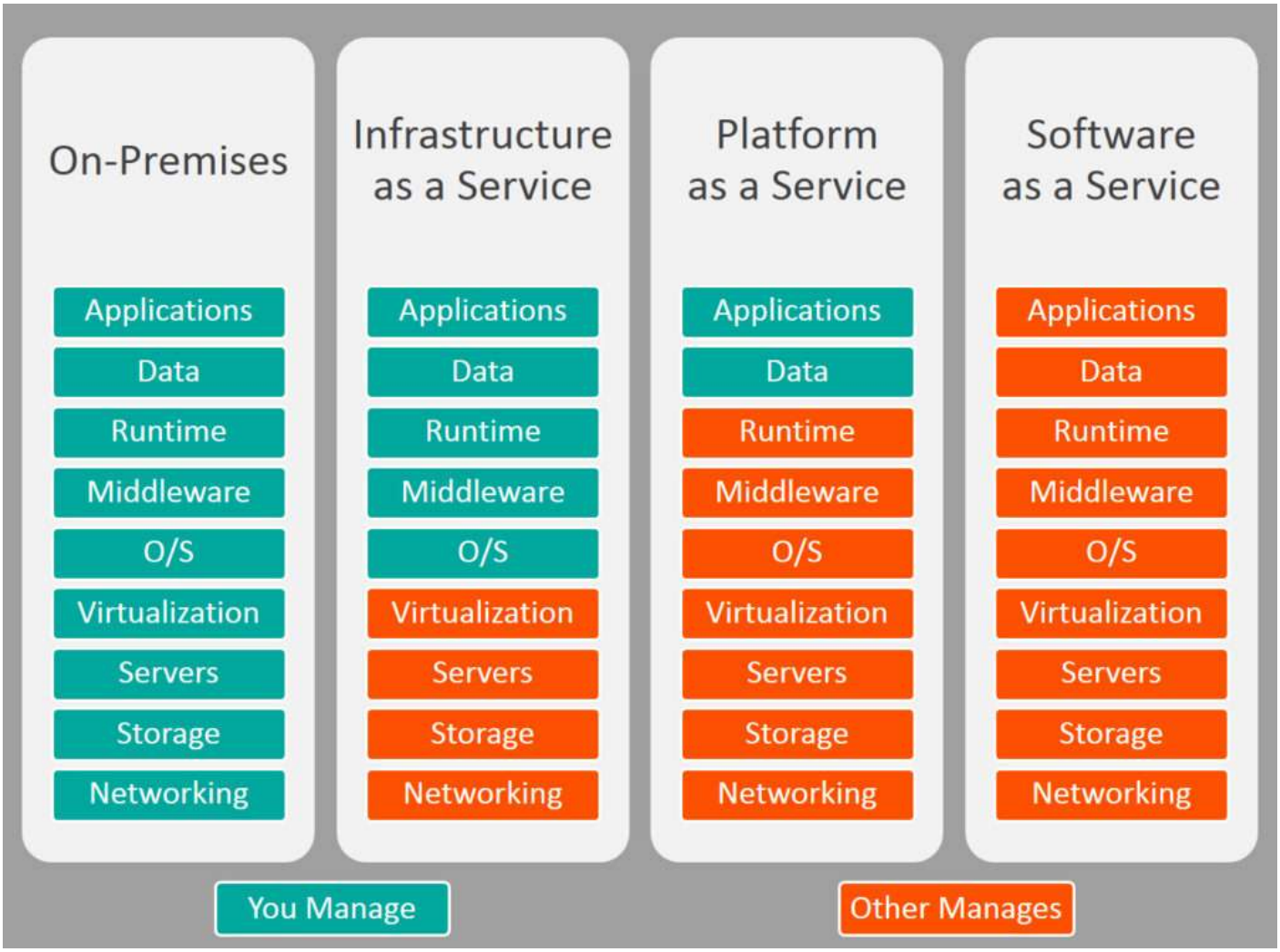
Infrastructure as a service (IaaS) is a cloud computing offering in which a vendor provides users access to computing resources such as servers, storage and networking. Organizations use their own platforms and applications within a service provider's infrastructure.

Key features

- Instead of purchasing hardware outright, users pay for IaaS on demand.
- Infrastructure is scalable depending on processing and storage needs.
- Saves enterprises the costs of buying and maintaining their own hardware.
- Because data is on the cloud, there can be no single point of failure.
- Enables the virtualization of administrative tasks, freeing up time for other work.

Examples of IaaS

Popular examples of IaaS include DigitalOcean, Linode, Rackspace, Amazon Web Services (AWS), Cisco Metacloud, Microsoft Azure, and Google Compute Engine (GCE).

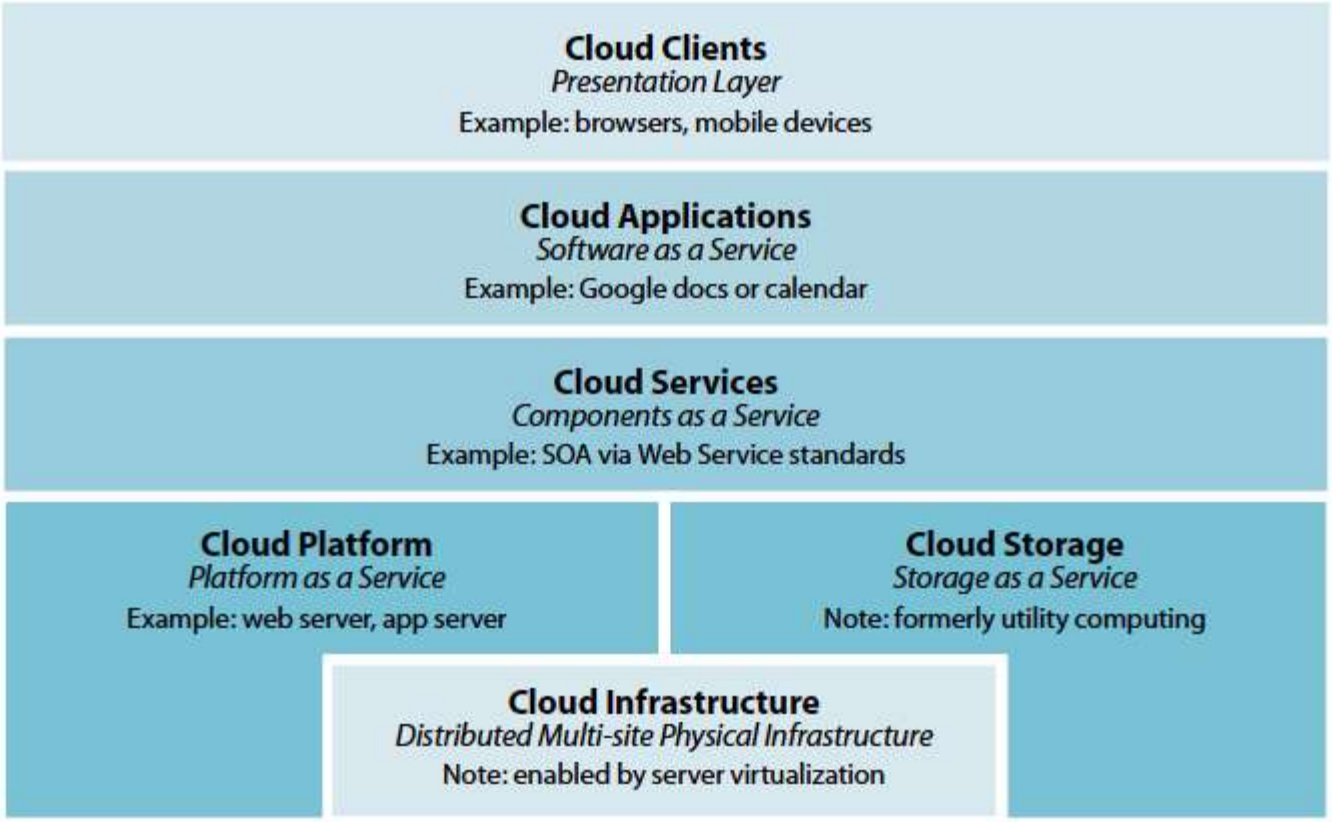


Service Orientated Architecture (SOA) and Cloud Computing

The service-oriented computing (SOC) paradigm and its realization through standardized web service technologies provides a promising solution for the seamless integration of single-function applications to create new large-grained and value-added services. In recent years, SOC, especially service composition, was applied in a lot of domains, e.g., workflow management, finances, e-business, e-science, etc.

What is the relationship between Cloud Computing and Service Orientated Architecture (SOA)?

According to the publication from Mitre, *Cloud Computing and Service Orientated Architecture (SOA)*, cloud computing has many services that can be viewed as a stack of service categories. These service categories include Infrastructure-as-a-Service (IaaS, Platform-as-a-Service (PaaS), Storage-as-a-Service, Components-as-a-Service, Software-as-a-Service (SaaS) and Cloud Clients. The following figure shows the service categories stack as depicted in the Mitre publication:



Mitre's Cloud Stack

SOA is a framework that allows business processes to be highlighted to deliver interoperability and rapid delivery of functionality. It helps system-to-system integration by creating loosely coupled services that can be reused for multiple purposes. The concept of SOA is similar to Object-Orientated Programming where objects are generalized so that they can be reused for multiple purposes.

Now that we have an understanding of the various types of Cloud Computing services and SOA, let's explore how Cloud Computing and SOA are similar and different.

Similarities between Cloud Computing and SOA:

- **Reuse** – Conceptually speaking, the idea of reuse is inherent both in Cloud Computing and SOA.
- **As needed basis** – In Cloud Computing, the services are provided to the users on-demand and as needed. SOA is similar to this since the system-to-system services are on-demand and as needed as well.
- **Network Dependency** – Cloud Computing and SOA both require an available and reliable network. If a network does not exist then the cloud services provided over the Internet would not be possible. Similarly, if a network does not exist then the communications between systems would not be possible. Thus, both Cloud Computing and SOA are dependent on a network.
- **Cloud Contracts** – In Cloud Computing, contracts entail the mutual agreement between an organization and cloud service providers. In cloud contracts, there is a cloud service provider and a cloud service consumer (the organization). In the case of SOA, contracts are important and can be either external (e.g., Yahoo! Pipes) and/or internal (e.g., organizational system integration). In SOA contracts, there are service producer(s) and service consumer(s) that are conceptually similar to cloud contracts.

Differences between Cloud Computing and SOA(1):

Despite the similarities between Cloud Computing and SOA, they are not the same. Following are some of the differences between them:

- **Outcome vs. Technology** – In Cloud Computing, we are paying for the outcome but in SOA we are paying for technology.
- **External vs. External and/or Internal Point-of-View** – In Cloud Computing, the services that organizations get are from external organization but in SOA these services can be either from external organizations (e.g., Yahoo! Pipes)

and/or internally (e.g., system-to-system integration between two or more systems).

- **IaaS, PaaS, SaaS vs. Software Components** – In Cloud Computing, the services provided can go up and down the stack but in SOA the services are software components.

Difference between Cloud computing and SOA(2).

Cloud Computing	SOA
Cloud computing is information on demand.	SOA is service on demand.
Internet based computing.	Technology based computing.
Service is delivered over the internet.	Service is delivered by interacting with various other services or software.
Here the client inheriting this application need a system and internet service to implement cloud computing.	Here the client needs a technology provider along with various services.

Cloud Deployment Models

There are 3 fundamental Deployment Models of cloud computing: Public Cloud, Private Cloud and Hybrid Cloud.

Public cloud

In **Public Cloud model**, services and infrastructure are hosted on premise of cloud provider and are provisioned for open use by general public. The end users can access the services via public network like internet. Public Cloud services are delivered mostly on demand. Popular for hosting everyday apps like email, CRM and other business support apps.

Public Cloud model offers high scalability, automated maintenance but more vulnerable to attacks due to high levels of accessibility.

Common Public Cloud providers include Amazon Web Services and Microsoft Azure.

Private Cloud

Private Cloud model provides cloud services and infrastructure exclusively to a single tenant. The tenant can control and customize it to his need. The cloud infrastructure can be monitored either by cloud provider or the tenant. Many companies are migrating their data centers to Private Cloud to run core business fields like research, manufacturing human resource etc.

The **Private Cloud model** offers great levels of security and control, though cost benefits ought to be sacrificed to some extent.

Common Private Cloud providers include VMware and Openstack.

Hybrid Cloud

As the name suggests Hybrid Cloud is composition of both Public Cloud and Private Cloud infrastructure. The company can use Private Cloud to run mission critical operations and Private Cloud to run non sensitive high demand operations.

The companies using **Hybrid Cloud model** benefit with the security and control aspect of Private Cloud and off-hand management and cost benefits of Public Cloud.

Community Cloud

It is another type of cloud computing in which the setup of the cloud is shared manually among different organizations that belong to the same community or area. Example of such a community is where organizations/firms are there along with the financial institutions/banks. A multi-tenant setup developed using cloud among different organizations that belong to a particular community or group having similar computing concern.

/////////////There are 4 fundamental Deployment Models of cloud computing: Public Cloud, Private Cloud and Hybrid Cloud.

Public Cloud

The name speaks for itself, as public clouds are available to the general public and data are created and stored on third-party servers. As server infrastructure belongs to service providers that manage them and administer pool resources, the need for user companies to buy and maintain their own hardware is eliminated. Provider companies offer resources as a service on a free of charge or pay-per-use basis via the Internet connection. Users can scale them when required.

At the same time, relying on a third party in running their infrastructure deprives users of knowing where their information is kept and who has access to it. Often enough, public clouds experience outages and malfunction, as in the case of the Salesforce CRM disruption in 2016 that caused a 10-hour storage collapse.

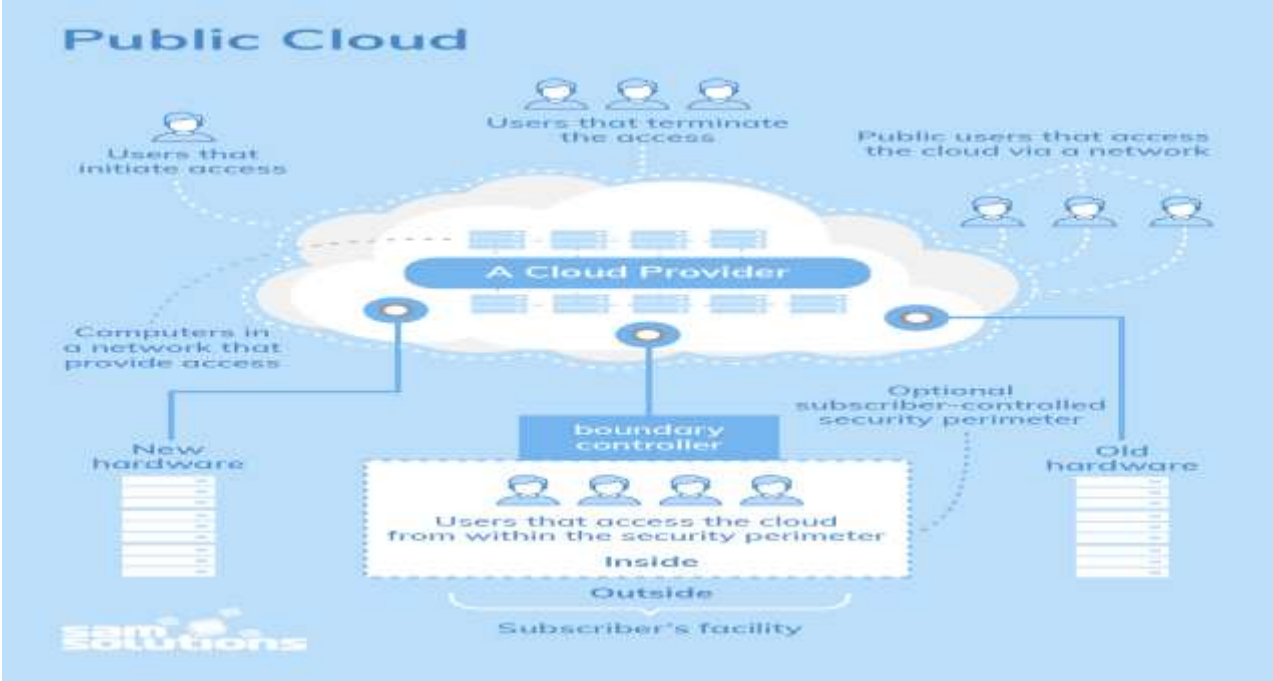
The pros of a public cloud are:

- Unsophisticated setup and use
- Easy access to data
- Flexibility to add and reduce capacity
- Cost-effectiveness
- Continuous operation time
- 24/7 upkeep
- Scalability
- Eliminated need for software

The cons of a public model:

- Data security and privacy
- Compromised reliability
- The lack of individual approach

The public cloud deployment model is the first choice of businesses that operate within the industries with low privacy concerns. When it comes to popular cloud deployment models, examples are Amazon Elastic Compute, Google AppEngine, IBM's Blue, Microsoft Azure, Salesforce Heroku and others.



Private Cloud

There is little to no difference between public and private clouds from the technical point of view, as their designs are very similar. However, unlike in the public one, only one specific company owns a private cloud, which is why it is also called internal or corporate. Because these data center architectures reside within the firewall, they provide enhanced security. Even though one organization runs its workloads on a private basis, a third party can also manage it, and the server can be hosted externally or on-premises of the user company.

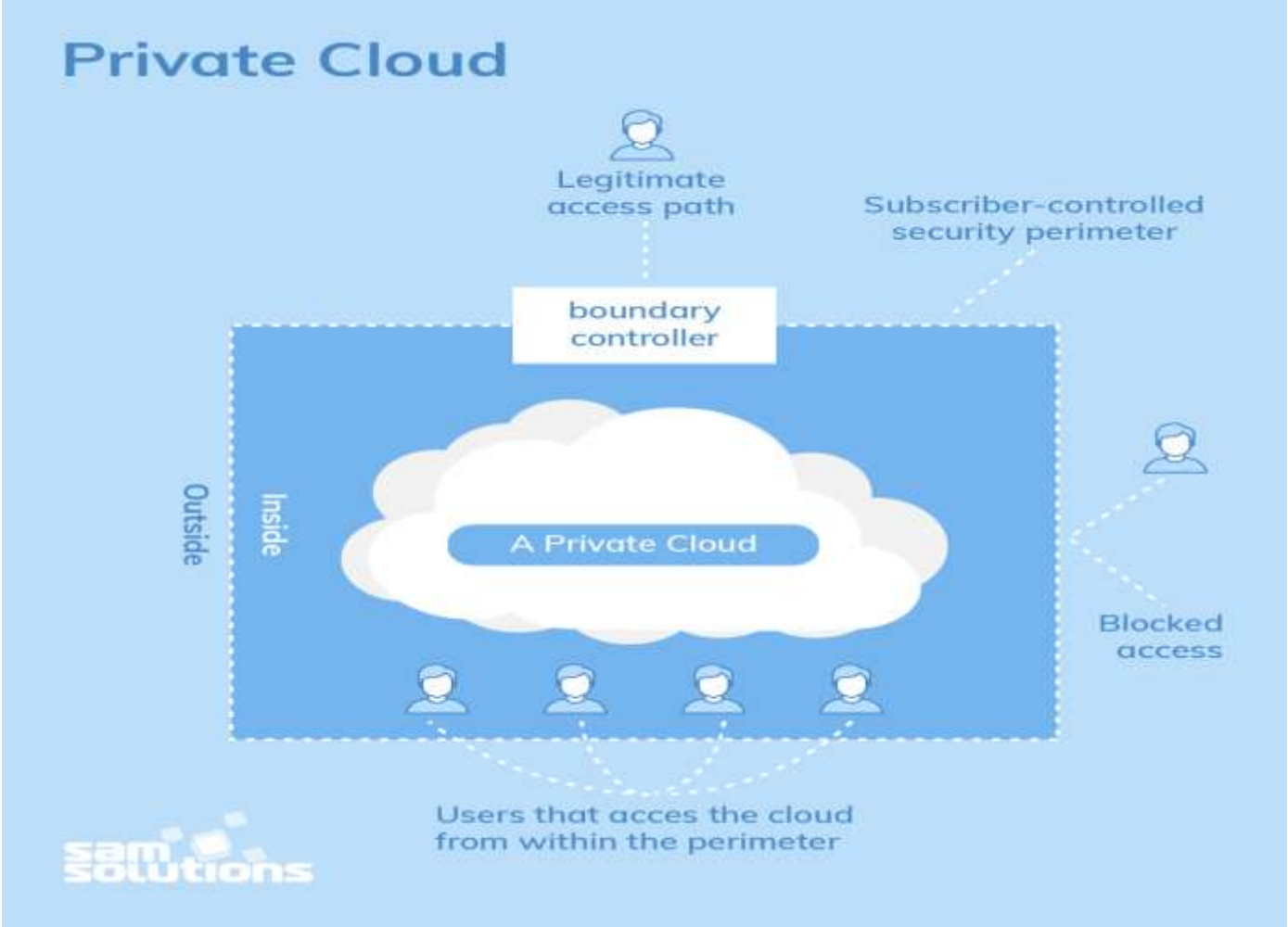
Only a clearly defined scope of persons have access to the information kept in a private repository, preventing the general public from using it. In light of numerous breaches, a growing number of large corporations decided on a closed private type as it is expected to be less risky.

The advantages of a private model:

- Individual development
- Storage and network components are customizable
- High control over the corporate information
- High security, privacy and reliability

The major disadvantage of the private cloud deployment model is its cost intensiveness, as it entails considerable expenses on hardware, software and staff training. That is why this secure flexible computing deployment model is not a choice of small to medium companies. Also, it is especially suitable for companies that seek to safeguard their mission-critical operations or for businesses with changing requirements.

Multiple service providers – including Amazon, IBM, Cisco, Dell and Red Hat – also build private solutions. We at SaM Solutions have created our proprietary cloud solution — CloudBOX. It is a ready-to-use Platform as a Service that facilitates projects by their quick and easy launching.



Community Cloud

A community cloud deployment model resembles a private one to a large extent; the only difference is the set of users. While a private type implies that only one company owns the server, in the case of a community one, several organizations with similar backgrounds share the infrastructure and related resources.

As the organizations have uniform security, privacy and performance requirements, this multi-tenant data center architecture helps companies achieve their business-specific objectives. That is why a community model is particularly suited for organizations that work on joint projects. In that case, a centralized cloud facilitates project development, management and implementation. Also, the costs are shared across all users.

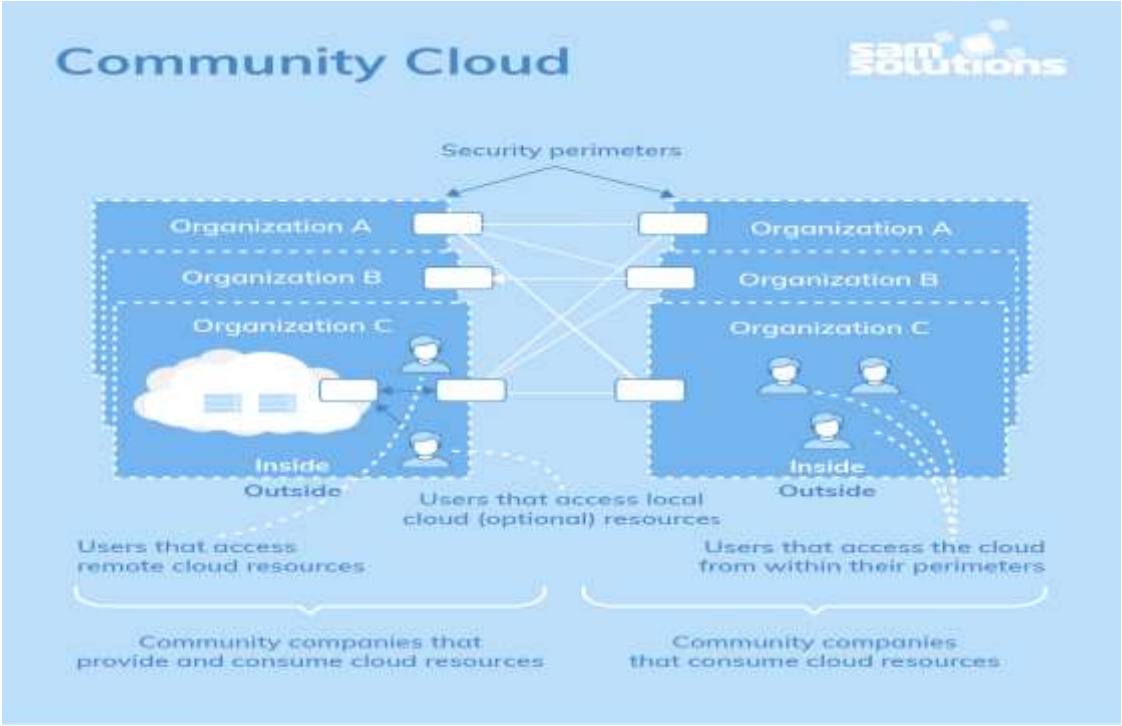
The strengths of a community computing type include the following:

- Cost reduction
- Improved security, privacy and reliability
- Ease of data sharing and collaboration

The shortcomings are:

- Higher cost than that of a public one
- Sharing of fixed storage and bandwidth capacity
- It is not widespread so far

Companies can decide on community solutions that Google, Red Hat, IBM, Microsoft or others provide.



Hybrid Cloud

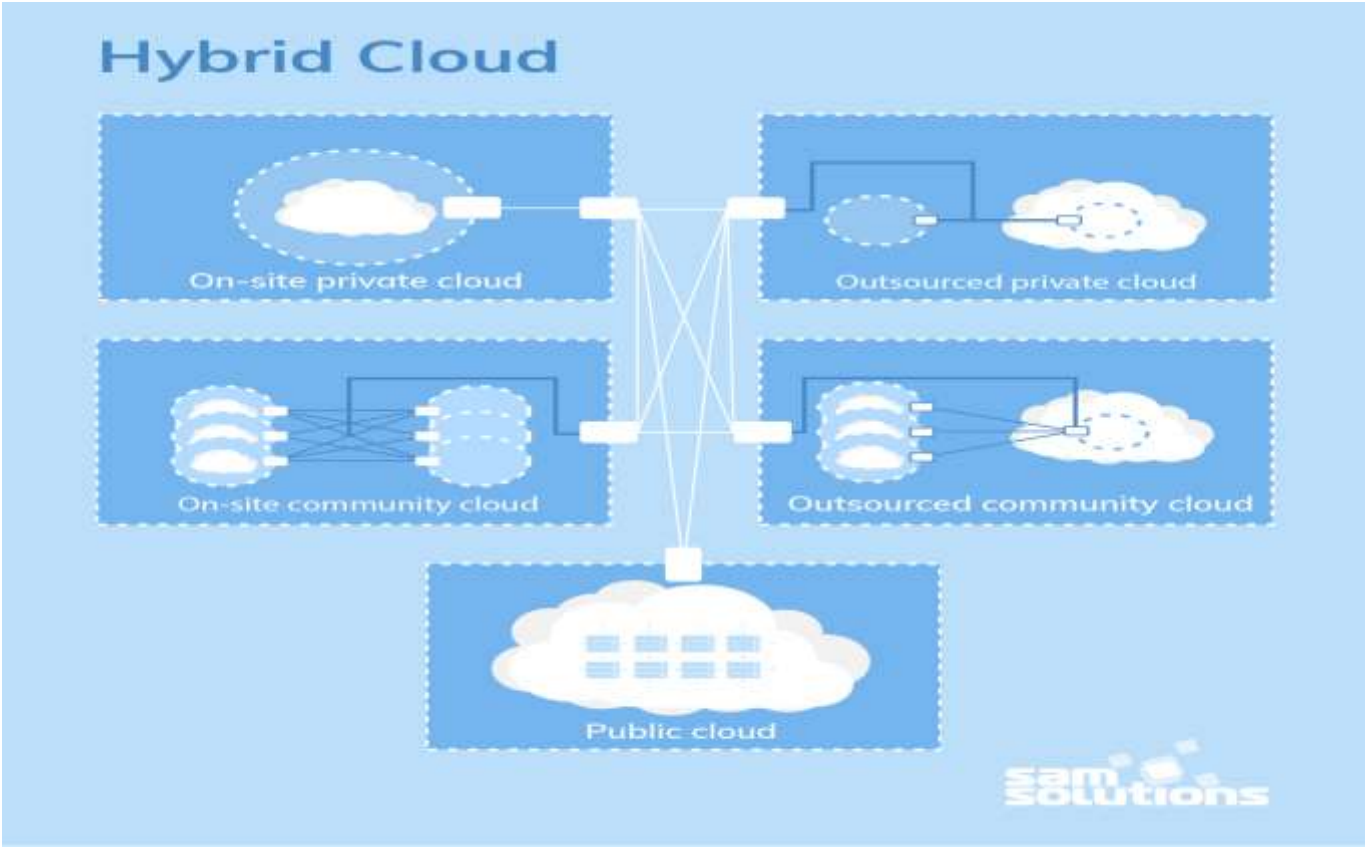
As it is usually the case with any hybrid phenomenon, a hybrid cloud encompasses the best features of the above-mentioned cloud computing deployment models – a public, private and community ones. It allows companies to mix and match the facets of all three types that best suit their requirements.

As an example, a company can balance its load by locating mission-critical workloads on a secure private cloud and deploying less sensitive ones to a public one. It not only safeguards and controls strategically important assets but does so in the most cost- and resource-effective way possible for each specific case. Also, this approach facilitates data and application portability.

The benefits of a hybrid model are:

- Improved security and privacy
- Enhanced scalability and flexibility
- Reasonable price

However, the hybrid cloud deployment model only makes sense if companies can split their data into mission-critical and non-sensitive.



Types of Cloud Deployment Models: The Comparison

To facilitate the choice of the appropriate deployment models of cloud computing by opting for the ones with the most business-critical features, we have created a comparative table that provides an overall view of the specificity of each type.

The comparative analysis of the best cloud deployment models

	Public	**Private**	**Community**	**Hybrid**
Ease of setup and use	Easy	Requires IT proficiency	Requires IT proficiency	Requires IT proficiency
Data security and privacy	Low	High	Comparatively high	High
Data control	Little to none	High	Comparatively high	Comparatively high
Reliability	Vulnerable	High	Comparatively high	High
Scalability and flexibility	High	High	Fixed capacity	High
Cost-effectiveness	The cheapest one	Cost-intensive, the most expensive one	Cost is shared among community members	Cheaper than a private model but more costly than a public one
Demand for in-house hardware	No	Depends	Depends	Depends