**19IINFCX1: CLOUD COMPUTING**

**Unit-1**

**Computing Basics:** Cloud computing definition- Characteristics- Benefit-Challenges- Distributed Systems- Virtualization-Service-oriented computing- Utility-oriented computing- Building Cloud Computing environments- computing platforms & technologies - Cloud Models – Cloud Service Examples - Cloud Based Services & Applications - Cloud concepts and Technologies.

**CLOUD COMPUTING DEFINITION**

Cloud computing is the delivery of computing services—including servers, storage, databases, networking, software, analytics, and intelligence—over the internet (“the cloud”) to offer faster innovation, flexible resources, and economies of scale. You typically pay only for cloud services you use, helping you lower your operating costs, run your infrastructure more efficiently, and scale as your business needs change.

## Types of cloud services: IaaS, PaaS, serverless, and SaaS

Most cloud computing services fall into four broad categories: infrastructure as a service (IaaS), platform as a service (PaaS), serverless, and software as a service (SaaS).

### IaaS

The most basic category of cloud computing services. With infrastructure as a service (IaaS), you rent IT infrastructure—servers and virtual machines (VMs), storage, networks, operating systems—from a cloud provider on a pay-as-you-go basis.

### PaaS

Platform as a service (PaaS) refers to cloud computing services that supply an on-demand environment for developing, testing, delivering, and managing software applications. PaaS is designed to make it easier for developers to quickly create web or mobile apps, without worrying about setting up or managing the underlying infrastructure of servers, storage, network, and databases needed for development.

### SaaS

Software as a service (SaaS) is a method for delivering software applications over the internet, on demand and typically on a subscription basis. With SaaS, cloud providers host and manage the software application and underlying infrastructure, and handle any maintenance, like software upgrades and security patching. Users connect to the application over the internet, usually with a web browser on their phone, tablet, or PC.

### Server less computing

Overlapping with PaaS, serverless computing focuses on building app functionality without spending time continually managing the servers and infrastructure required to do so. The cloud provider handles the setup, capacity planning, and server management for you. Serverless architectures are highly scalable and event-driven, only using resources when a specific function or trigger occurs.

**BENEFITS OF CLOUD COMPUTING**

**Accessibility anywhere, with any device**

Each branch or office across various states or countries. The improved accessibility doesn’t just impact employees; clients and customers can also log in to an account and access their information as well. This ensures everyone has up-to-date information whether they’re at the office or on the go.

**Ability to get rid of most or all hardware and software**

With cloud computing, you’re no longer required to have your own server, cables, network switches, backup generators, redundant routers, and so on. Depending on the cloud provider you choose, they can manage all of this for a monthly fee. Reducing expenses is essential in any business model and every cloud-based platform benefits from this factor alone.

**Centralized data security**

When you use cloud computing, data backups are centralized in the cloud providers' data centers, removing the need for individual users or teams to maintain their own backups onsite or offsite. This lowers the risk of data loss should any one backup fail or be destroyed by a disaster. Cloud providers can restore the data from another copy maintained in their cloud storage, which is continuously updated with every piece of data added.

**Higher performance and availability**

By using cloud computing resources together simultaneously, you reap greater performance gains than by having your own dedicated server hardware. Cloud computing increases input/output operations per second (IOPS).

**Quick application deployment**

Unpredictable business needs often require cloud computing resources on short notice. You can improve your cloud application development by quickly deploying cloud applications because they are readily available without the need to procure additional hardware or wait for IT staff to set up servers.

**Instant business insights**

Cloud-based platforms provide a unique opportunity to access data as soon as it’s collected. This facilitates better decision-making as well as insight into what the future may hold for your organization based on predictions from historical data.

**Business continuity**

In the event of disaster or unforeseen circumstances, do you have an effective backup plan? If not, relying on cloud computing services can benefit your organization. Cloud computing uses infinite data storage space and systems that can be activated remotely if necessary to ensure business continuity.

**Price-performance and cost savings**

Although an initial financial investment is required to implement a cloud strategy, organizations save substantial amounts in the long run because they don’t have to maintain expensive hardware or local data centers. Also, since there are no upfront costs to use cloud-based systems, businesses can test them out before investing in them at their own pace. Oracle provides price-performance and flexible sizing.

**Virtualized computing**

Cloud computing is perfect for virtualized computer environments because cloud resources can be allocated instantly to support significant increases in demand so you never experience downtime again. With cloud computing, your business can expand its capabilities almost effortlessly to meet growing demands without increasing staff or capital expenditures.

**Cloud computing is greener**

Cloud computing is a greener technology than traditional IT solutions. By moving to the cloud, businesses can reduce their energy consumption and carbon footprint by up to 90%. Rather than having in-house servers and software, businesses can use cloud-based services to access the same applications and data from any computer or device with an internet connection. This eliminates the need for businesses to purchase and maintain their own IT infrastructure.

**CHALLENGES OF CLOUD COMPUTING**

The most common challenges that are faced when dealing with cloud computing, let’s have a look at them one by one:

**1. Data Security and Privacy**

Data security is a major concern when switching to cloud computing. User or organizational data stored in the cloud is critical and private. Even if the cloud service provider assures data integrity, it is your responsibility to carry out user authentication and authorization, identity management, data encryption, and access control. Security issues on the cloud include identity theft, data breaches, malware infections, and a lot more which eventually decrease the trust amongst the users of your applications. This can in turn lead to potential loss in revenue alongside reputation and stature. Also, dealing with cloud computing requires sending and receiving huge amounts of data at high speed, and therefore is susceptible to data leaks.

**2. Cost Management**

Even as almost all cloud service providers have a “Pay As You Go” model, which reduces the overall cost of the resources being used, there are times when there are huge costs incurred to the enterprise using cloud computing. When there is under optimization of the resources, let’s say that the servers are not being used to their full potential, add up to the hidden costs. If there is a degraded application performance or sudden spikes or overages in the usage, it adds up to the overall cost. Unused resources are one of the other main reasons why the costs go up. If you turn on the services or an instance of cloud and forget to turn it off during the weekend or when there is no current use of it, it will increase the cost without even using the resources.

**3. Multi-Cloud Environments**

Due to an increase in the options available to the companies, enterprises not only use a single cloud but depend on multiple cloud service providers. Most of these companies use hybrid cloud tactics and close to 84% are dependent on multiple clouds. This often ends up being hindered and difficult to manage for the infrastructure team. The process most of the time ends up being highly complex for the IT team due to the differences between multiple cloud providers.

**4. Performance Challenges**

Performance is an important factor while considering cloud-based solutions. If the performance of the cloud is not satisfactory, it can drive away users and decrease profits. Even a little latency while loading an app or a web page can result in a huge drop in the percentage of users. This latency can be a product of inefficient load balancing, which means that the server cannot efficiently split the incoming traffic so as to provide the best user experience. Challenges also arise in the case of fault tolerance, which means the operations continue as required even when one or more of the components fail.

**5. Interoperability and Flexibility**

 When an organization uses a specific cloud service provider and wants to switch to another cloud-based solution, it often turns up to be a tedious procedure since applications written for one cloud with the application stack are required to be re-written for the other cloud. There is a lack of flexibility from switching from one cloud to another due to the complexities involved. Handling data movement, setting up the security from scratch and network also add up to the issues encountered when changing cloud solutions, thereby reducing flexibility.

**6. High Dependence on Network**

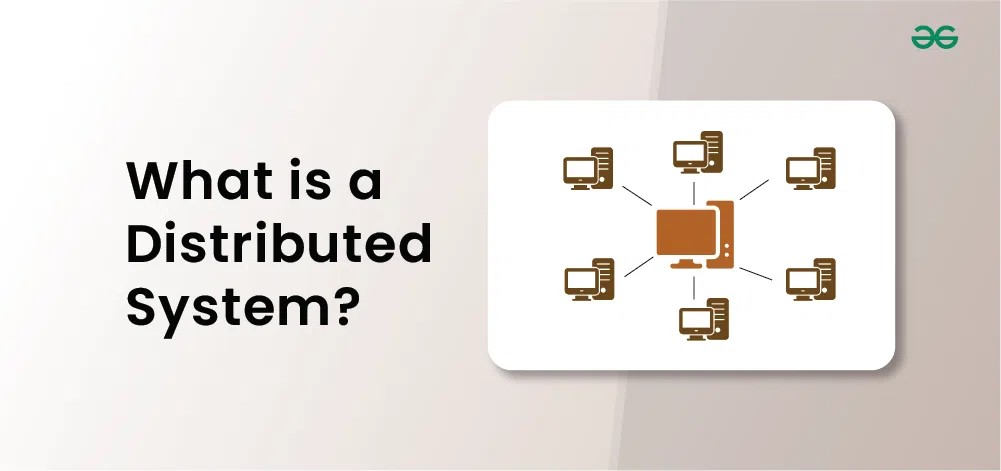
Since cloud computing deals with provisioning resources in real-time, it deals with enormous amounts of data transfer to and from the servers. This is only made possible due to the availability of the high-speed network. Although these data and resources are exchanged over the network, this can prove to be highly vulnerable in case of limited bandwidth or cases when there is a sudden outage. Even when the enterprises can cut their hardware costs, they need to ensure that the internet bandwidth is high as well there are zero network outages, or else it can result in a potential business loss. It is therefore a major challenge for smaller enterprises that have to maintain network bandwidth that comes with a high cost.

**7. Lack of Knowledge and Expertise**

Due to the complex nature and the high demand for research working with the cloud often ends up being a highly tedious task. It requires immense knowledge and wide expertise on the subject. Although there are a lot of professionals in the field they need to constantly update themselves. Cloud computing is a highly paid job due to the extensive gap between demand and supply. There are a lot of vacancies but very few talented cloud engineers, developers, and professionals. Therefore, there is a need for upskilling so these professionals can actively understand, manage and develop cloud-based applications with minimum issues and maximum reliability.

**DISTRIBUTED SYSTEMS**

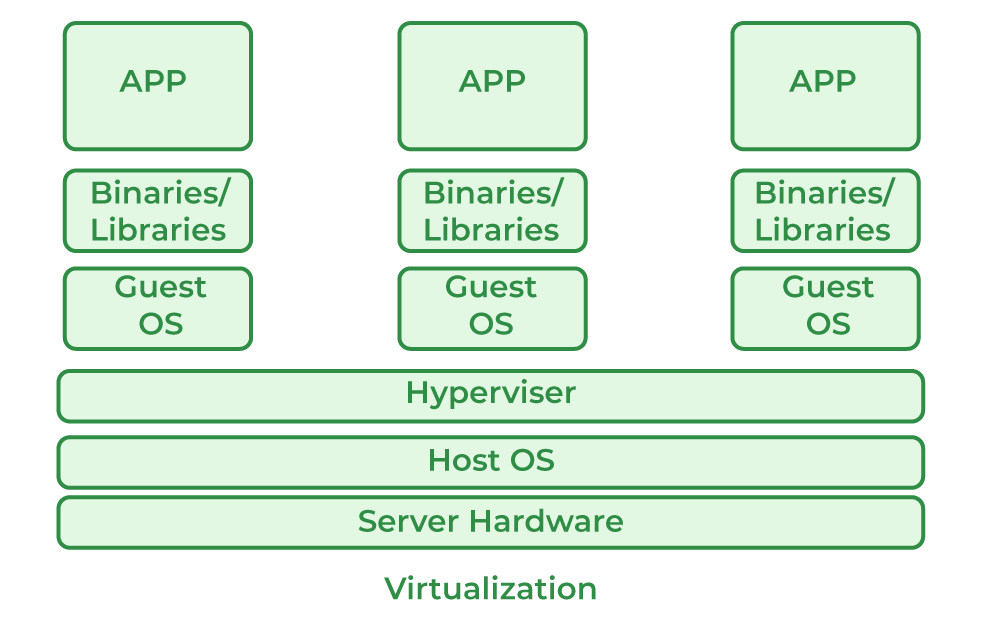
A distributed system is a collection of independent computers that appear to the users of the system as a single coherent system. These computers or nodes work together, communicate over a network, and coordinate their activities to achieve a common goal by sharing resources, data, and tasks.



**VIRTUALIZATION**

Virtualization is used to create a virtual version of an underlying service With the help of Virtualization, multiple operating systems and applications can run on the same machine and its same hardware at the same time, increasing the utilization and flexibility of hardware. It was initially developed during the mainframe era.

It is one of the main cost-effective, hardware-reducing, and energy-saving techniques used by cloud providers. Virtualization allows sharing of a single physical instance of a resource or an application among multiple customers and organizations at one time. It does this by assigning a logical name to physical storage and providing a pointer to that physical resource on demand. The term virtualization is often synonymous with hardware virtualization, which plays a fundamental role in efficiently delivering Infrastructure-as-a-Service (IaaS) solutions for cloud computing. Moreover, virtualization technologies provide a virtual environment for not only executing applications but also for storage, memory, and networking.



* Host Machine: The machine on which the virtual machine is going to be built is known as Host Machine.
* Guest Machine: The virtual machine is referred to as a Guest Machine.

**Work of Virtualization in Cloud Computing**

Virtualization has a prominent impact on Cloud Computing. In the case of cloud computing, users store data in the cloud, but with the help of Virtualization, users have the extra benefit of sharing the infrastructure. Cloud Vendors take care of the required physical resources, but these cloud providers charge a huge amount for these services which impacts every user or organization. Virtualization helps Users or Organisations in maintaining those services which are required by a company through external (third-party) people, which helps in reducing costs to the company. This is the way through which Virtualization works in Cloud Computing.

**Benefits of Virtualization**

* More flexible and efficient allocation of resources.
* Enhance development productivity.
* It lowers the cost of IT infrastructure.
* Remote access and rapid scalability.
* High availability and disaster recovery.
* Pay peruse of the IT infrastructure on demand.
* Enables running multiple operating systems.

**Drawback of Virtualization**

* **High Initial Investment:**Clouds have a very high initial investment, but it is also true that it will help in reducing the cost of companies.
* **Learning New Infrastructure:** As the companies shifted from Servers to Cloud, it requires highly skilled staff who have skills to work with the cloud easily, and for this, you have to hire new staff or provide training to current staff.
* **Risk of Data:**Hosting data on third-party resources can lead to putting the data at risk, it has the chance of getting attacked by any hacker or cracker very easily.

**Characteristics of Virtualization**

* **Increased Security:** The ability to control the execution of a guest program in a completely transparent manner opens new possibilities for delivering a secure, controlled execution environment. All the operations of the guest programs are generally performed against the virtual machine, which then translates and applies them to the host programs.
* **Managed Execution:** In particular, sharing, aggregation, emulation, and isolation are the most relevant features.
* **Sharing:** Virtualization allows the creation of a separate computing environment within the same host.
* **Aggregation:** It is possible to share physical resources among several guests, but virtualization also allows aggregation, which is the opposite process.

**SERVICE-ORIENTED COMPUTING**

Service Oriented Computing is a model that aims to provide standardized models and protocols to make services easily accessible and interoperable among distributed application components. It allows users to use services on-demand, relieving them from the need to build and maintain a complete system in-house.

# 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.

## What is 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.

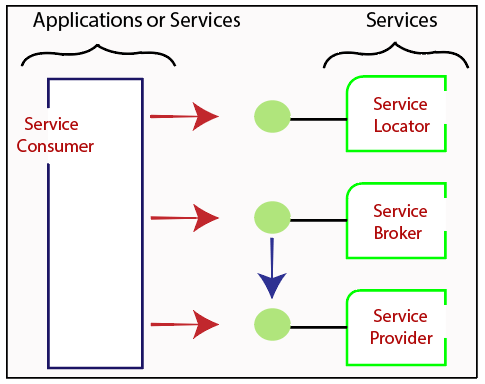
## Service Connections

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**

Let's see some important 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.

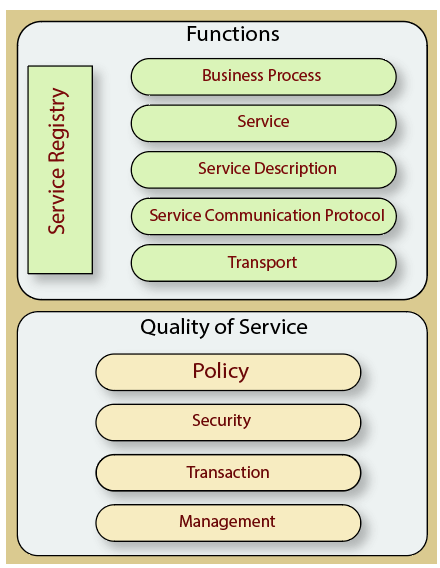
**Characteristics of SOA**

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 make 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.

**UTILITY-ORIENTED COMPUTING**

Utility computing is a groundbreaking concept in cloud computing that has transformed how organizations access and manage their computing resources. In this blog, we will explore the fundamentals of utility computing, its key benefits, examples, and significant impact on modern technology and business operations.

**What is Utility Computing?**

Utility computing is a service model where computing resources are provided on demand, similar to how you pay for electricity or water. Instead of buying and maintaining your own servers or storage, you rent them from a service provider. You pay only for what you use, which can lead to cost savings. This model makes it easier to scale up or down as your needs change.

**How Does Utility Computing in Cloud Computing Work?**

Utility computing works on a “pay-as-you-go” model. Below is a comprehensive explanation of how utility computing in cloud computing operates:

**Service Providers:** Companies that offer utility computing services maintain vast data centers with servers, storage, and networking capabilities.

**Customer Needs:** Businesses or individual users assess their computing requirements, such as server space, processing power, or storage capacity.

**On-Demand Access:** Customers can then access these resources online, usually through a web-based interface or application programming interface (API).

**Resource Allocation:** The utility computing provider dynamically allocates resources based on the customer’s needs. This can happen almost instantly, making it easy to scale up or down.

**Usage Monitoring:** The provider tracks the resources you use, often in real time. This could be measured in data storage, processing power, bandwidth, or a combination.

**Billing:** At the end of a billing cycle, the customer is charged based on their actual usage. This avoids the cost of purchasing and maintaining in-house hardware.

**Flexibility:** As your needs change, you can easily adjust your usage, adding more resources during peak times or scaling down when less is needed.

**Maintenance:** The service provider takes care of all the backend maintenance tasks, such as software updates, security patches, and hardware upkeep, freeing you to focus on your core business.

**Benefits of Utility Computing in Cloud Computing**

Below, we have highlighted some of the benefits of utility computing in cloud computing:

**Cost Efficiency:** Utility computing allows users to pay only for the computing resources they actually use. This removes the necessity for large upfront investments in hardware and infrastructure, reducing capital expenses. Operating costs are optimized as users can scale resources up or down as needed, avoiding over-provisioning.

**Scalability:**Utility computing offers unparalleled scalability. Businesses can easily adapt to changing demands by adding or reducing resources on the fly. This flexibility ensures that systems can handle increased workloads during peak periods without the need for significant manual intervention.

**Accessibility and Convenience:** With utility computing, resources and applications are accessible from virtually anywhere with internet connectivity. This accessibility facilitates remote work, collaboration, and disaster recovery planning. It ensures that critical data and applications are always within reach.

**Reliability and Redundancy:** Leading utility computing providers offer high levels of reliability and redundancy. Data is often stored across multiple data centers with robust backup systems. This reduces the risk of data loss and downtime due to hardware failures.

**Improved Resource Utilization:** Utility computing optimizes resource utilization. Virtualization technology allows for efficient use of physical hardware, reducing idle resources and increasing overall efficiency. This, in turn, benefits both the environment and the bottom line.

**Rapid Deployment:** Deploying new applications or services is expedited in utility computing environments. Infrastructure and development tools are readily available, reducing the time required to bring new ideas to market.

**Security:** Leading utility computing providers invest heavily in security measures. They employ advanced encryption, access controls, and monitoring to safeguard user data and applications. These providers often have dedicated security teams offering expertise in protecting against cyber threats.

**Global Reach:** Utility computing providers have data centers located across the globe. This global reach ensures low-latency access to resources and allows businesses to expand their operations internationally with ease.

**Examples of Utility Computing**

Here are some examples of utility computing in cloud computing that collectively demonstrate the flexibility, ease of use, and cost-effectiveness of utility computing.

* **Amazon Web Services (AWS)** **for Web Hosting:** This example illustrates how utility computing can eliminate the need to own and maintain physical servers. AWS provides various services on-demand, allowing you to easily scale resources up or down depending on your website’s traffic.
* **Microsoft Azure for Machine Learning:** Here, in this example, the focus is on specialized computing needs. Azure offers machine learning services that let engineers develop and deploy models without worrying about the underlying hardware or software. This means you can focus solely on the machine learning algorithms and data.
* **Google Cloud Platform (GCP)** **for Data Storage and Backup**: This example shows the benefits of utility computing for data storage. Just like a ride-sharing service charges you based on the distance you travel, in utility computing, you’re billed according to the computing resources you consume. This makes it a cost-effective solution, especially for businesses that need to maintain large amounts of data but may not need to access it all the time.

**BUILDING CLOUD COMPUTING ENVIRONMENTS**

Cloud computing is the on-demand delivery of computing services, like storage, software, analytics, and databases over the Internet to offer flexible resources and economies of scale. It is a paradigm shift from the traditional way businesses think about computing and IT resources.

**Types of Cloud Computing Environments**

**Public cloud:** The third-party cloud service providers own the public cloud and the supporting infrastructure like hardware and software. They deliver computing resources to organizations on the basis of a subscription fee. Users can get access to these services and manage their profiles using web browsers. This cloud deployment model is completely virtualized, which allows users to share resources while maintaining the security and privacy of each user’s data.

**Private cloud:** In this internal or corporate cloud environment, computing services are offered via a private network and used exclusively by a single customer or business organization. Companies can maintain their private cloud on their on-premise datacenters or subscribe to third-party service providers to host their private cloud. Private cloud not only offers several benefits of cloud computing, like flexibility and scalability but also enhances security and resource optimization of on-premise infrastructure.

**Hybrid cloud:** It is a combination of both public and private cloud, or like an on-premise data center and a public cloud, where data and applications can be shared securely between them. Most organizations do not depend entirely on the public cloud. A hybrid cloud environment offers greater flexibility, and more deployment options, and helps in maintaining the privacy and regulatory compliance.

**Difference between Multi-cloud and Hybrid Cloud**

Both hybrid and multi-cloud environments refer to cloud deployments with more than one cloud. The difference lies in the cloud infrastructure they incorporate.

While a multicloud environment is an integration of different clouds of the same nature, for instance, a combination of various public clouds (e.g. Amazon Web Services and Azure), hybrid cloud environments blend two or more varying types of clouds, like a combination of public and private cloud.

In a multi-cloud architecture, the solutions are offered by multiple third-party service providers. Hence, it operates in silos, whereas, in a hybrid cloud environment, the processes and data tend to interconnect. Multi-cloud deployment depends on third-party cloud utilization and cost control, while in a hybrid cloud environment, the native cloud utilization and cost control are emphasized.

**Types of Cloud Services**

**Software as a Service (SaaS):**It is a software licensing and delivery model where the cloud service providers host software applications, and infrastructure, and manage software upgrades and patching. Users can connect to the software applications over the Internet on a pay-as-you-go basis, using their devices like laptops and phones. The service provider ensures the security and availability of the applications and data privacy and manages the hardware and software with suitable service agreements.

**Platform as a Service (PaaS):** This cloud computing service works as an on-demand development and deployment environment for developing, managing, and testing mobile and web applications. PaaS helps developers quickly create applications without the added tension of setting up infrastructure, servers, storage, and databases and managing software licenses, which are crucial for app development. PaaS comprises the primary cloud features, like scalability, availability, and multi-tenant capability, and it allows developers to spend much lesser time on coding.

**Infrastructure as a Service (IaaS):** This cloud computing service, which was first introduced by Oracle in 2012, offers on-demand storage and networking resources in all types of cloud environments, public, private and hybrid clouds. IaaS replaces the complexity of managing physical servers and datacenter infrastructure and thus helps users to save expenses on hardware and on-premise datacenters. A cloud service provider manages the infrastructure and enables the users to purchase and configure their own software and operating systems.

**SaaS v/s On-premise**

The major difference between on-premise and SaaS is that SaaS solutions are hosted by third-party cloud service providers, whereas the on-premise solutions are hosted and managed in-house i.e. by business organizations.

When comparing these two solutions, you need to consider several factors:

* Cost: The pricing model of SaaS is flexible and the budget invested in upgrades is lower than on-premise solutions. SaaS also reduces the cost related to internal resources. The entry cost of single-tenant and multi-tenant SaaS solutions is lower than on-premise solutions, whose entry and operations costs are relatively higher.
* Scalability: SaaS solutions are highly scalable and flexible, and can be tailored as per business needs. On-premise solutions require long-term planning to scale and they are not as efficient as SaaS solutions as IT teams have to focus on continuous software upgrades from time to time. SaaS upgrades are much easier, and they require lesser intervention from the internal IT staff.
* Maintenance: For application support and maintenance, the SaaS solutions require no extra effort, as the cloud service providers take care of these concerns, and they are responsible for the availability, security, and performance. Businesses that use SaaS solutions have to trust the vendors when it comes to the security of their critical corporate data. On the other hand, on-premise solutions are owned, managed, and maintained by the organization itself, and it holds the sole responsibility for ensuring data security.
* Implementation: Implementing a single tenancy is expensive and not sustainable in the long run.  Implementing multi-tenant SaaS solutions is lesser time-consuming than the on-premise solutions. Users need to subscribe to the solutions provided by the cloud service providers and get started with their accounts. But, for on-premise solutions, human resources and monetary investments are needed, and the software and hardware upgrades are performed by the IT teams of the organization.
* Security and compliance: High-end SaaS providers, like Google and Microsoft, offer top-notch security and regulatory compliance, and organizations who subscribe to these SaaS solutions need not worry. The service providers offer baseline validation and enforce compliance. In the case of on-premise solutions, the security risks are higher and the internal IT teams are responsible for meeting the standards required for compliance.

**COMPUTING PLATFORMS & TECHNOLOGIES**

Cloud Computing applications develops by leveraging platforms and frameworks. Various types of services are provided from the bare metal infrastructure to customize-able applications serving specific purposes.



**Amazon Web Services (**AWS**)**

AWS provides different wide-ranging clouds IaaS services, which ranges from virtual compute, storage, and networking to complete computing stacks. AWS is well known for its storage and compute on demand services, named as Elastic Compute Cloud (EC2) and Simple Storage Service (S3). EC2 offers customizable virtual hardware to the end user which can be utilize as the base infrastructure for deploying computing systems on the cloud. It is likely to choose from a large variety of virtual hardware configurations including GPU and cluster instances. Either the AWS console, which is a wide-ranged Web portal for retrieving AWS services, or the web services API available for several programming language is used to deploy the EC2 instances. EC2 also offers the capability of saving an explicit running instance as image, thus allowing users to create their own templates for deploying system. S3 stores these templates and delivers persistent storage on demand. S3 is well ordered into buckets which contains objects that are stored in binary form and can be grow with attributes. End users can store objects of any size, from basic file to full disk images and have them retrieval from anywhere. In addition, EC2 and S3, a wide range of services can be leveraged to build virtual computing system including: networking support, caching system, DNS, database support, and others.

**Google AppEngine**

Google AppEngine is a scalable runtime environment frequently dedicated to executing web applications. These utilize benefits of the large computing infrastructure of Google to dynamically scale as per the demand. AppEngine offers both a secure execution environment and a collection of which simplifies the development if scalable and high-performance Web applications. These services include: in-memory caching, scalable data store, job queues, messaging, and corn tasks. Developers and Engineers can build and test applications on their own systems by using the AppEngine SDK, which replicates the production runtime environment, and helps test and profile applications. On completion of development, Developers can easily move their applications to AppEngine, set quotas to containing the cost generated, and make it available to the world. Currently, the supported programming languages are Python, Java, and Go.

**Microsoft Azure**

Microsoft Azure is a Cloud operating system and a platform in which user can develop the applications in the cloud. Generally, a scalable runtime environment for web applications and distributed applications is provided. Application in Azure are organized around the fact of roles, which identify a distribution unit for applications and express the application’s logic. Azure provides a set of additional services that complement application execution such as support for storage, networking, caching, content delivery, and others.

**Hadoop**

Apache Hadoop is an open source framework that is appropriate for processing large data sets on commodity hardware. Hadoop is an implementation of MapReduce, an application programming model which is developed by Google. This model provides two fundamental operations for data processing: map and reduce. Yahoo! Is the sponsor of the Apache Hadoop project, and has put considerable effort in transforming the project to an enterprise-ready cloud computing platform for data processing. Hadoop is an integral part of the Yahoo! Cloud infrastructure and it supports many business processes of the corporates. Currently, Yahoo! Manges the world’s largest Hadoop cluster, which is also available to academic institutions.

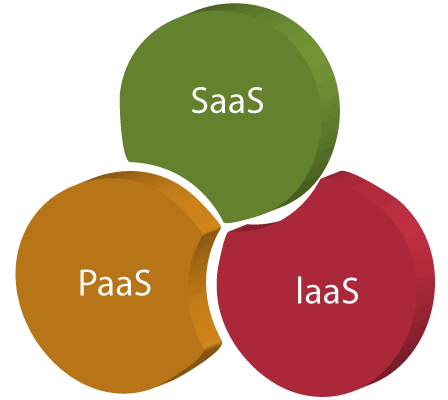
**Force.com and Salesforce.com**

Force.com is a Cloud computing platform at which user can develop social enterprise applications. The platform is the basis of SalesForce.com – a Software-as-a-Service solution for customer relationship management. Force.com allows creating applications by composing ready-to-use blocks: a complete set of components supporting all the activities of an enterprise are available. From the design of the data layout to the definition of business rules and user interface is provided by Force.com as a support. This platform is completely hostel in the Cloud, and provides complete access to its functionalities, and those implemented in the hosted applications through Web services technologies.

**CLOUD MODELS**

There are the following three types of cloud service models -

1. [Infrastructure as a Service (IaaS)](https://www.javatpoint.com/cloud-service-models#IaaS)
2. [Platform as a Service (PaaS)](https://www.javatpoint.com/cloud-service-models#PaaS)
3. [Software as a Service (SaaS)](https://www.javatpoint.com/cloud-service-models#SaaS)



**Infrastructure as a Service (IaaS)**

IaaS is also known as **Hardware as a Service (HaaS)**. It is a computing infrastructure managed over the internet. The main advantage of using IaaS is that it helps users to avoid the cost and complexity of purchasing and managing the physical servers.

Characteristics of IaaS

There are the following characteristics of IaaS

* Resources are available as a service
* Services are highly scalable
* Dynamic and flexible
* GUI and API-based access
* Automated administrative tasks

**Example:** DigitalOcean, Linode, Amazon Web Services (AWS), Microsoft Azure, Google Compute Engine (GCE), Rackspace, and Cisco Metacloud.

**Platform as a Service (PaaS)**

PaaS cloud computing platform is created for the programmer to develop, test, run, and manage the applications.

Characteristics of PaaS

There are the following characteristics of PaaS -

* Accessible to various users via the same development application.
* Integrates with web services and databases.
* Builds on virtualization technology, so resources can easily be scaled up or down as per the organization's need.
* Support multiple languages and frameworks.
* Provides an ability to "**Auto-scale**".

**Example:** AWS Elastic Beanstalk, Windows Azure, Heroku, Force.com, Google App Engine, Apache Stratos, Magento Commerce Cloud, and OpenShift.

**Software as a Service (SaaS)**

SaaS is also known as "**on-demand software**". It is a software in which the applications are hosted by a cloud service provider. Users can access these applications with the help of internet connection and web browser.

Characteristics of SaaS

There are the following characteristics of SaaS

* Managed from a central location
* Hosted on a remote server
* Accessible over the internet
* Users are not responsible for hardware and software updates. Updates are applied automatically.
* The services are purchased on the pay-as-per-use basis

**Example:** BigCommerce, Google Apps, Salesforce, Dropbox, ZenDesk, Cisco WebEx, ZenDesk, Slack, and GoToMeeting.

**Difference between IaaS, PaaS, and SaaS**

The below table shows the difference between IaaS, PaaS, and SaaS -

|  |  |  |
| --- | --- | --- |
| **IaaS** | **Paas** | **SaaS** |
| It provides a virtual data center to store information and create platforms for app development, testing, and deployment. | It provides virtual platforms and tools to create, test, and deploy apps. | It provides web software and apps to complete business tasks. |
| It provides access to resources such as virtual machines, virtual storage, etc. | It provides runtime environments and deployment tools for applications. | It provides software as a service to the end-users. |
| It is used by network architects. | It is used by developers. | It is used by end users. |
| IaaS provides only Infrastructure. | PaaS provides Infrastructure+Platform. | SaaS provides Infrastructure+Platform +Software. |

**Advantages of Cloud Service Models**

**Cost Efficiency:** Cloud providers provide a pricing model that permits customers to pay only for the sources they consume. This gets rid of the need for advanced infrastructure investments and allows price efficiency as businesses scale resources based totally on need.

**Scalability:** Cloud services provide the potential to scale sources up or down speedily and respond to changing workloads and commercial organization requirements. This flexibility ensures that agencies can correctly manipulate fluctuating needs without over-provisioning.

**Accessibility and Flexibility:** Cloud computing allows one to get access to applications and facts remotely from everywhere with an internet connection. This fosters collaboration among geographically dispersed groups and allows users to work flexibly.

**Rapid Deployment:** Cloud provider models facilitate rapid deployment of programs. Users can provision sources and deploy programs quickly, decreasing time-to-market and allowing faster innovation.

**Managed Services:** Cloud providers offer more than a few managed offerings, managing duties together with safety, tracking, and safety. This helps agencies dump operational obligations, pay attention to relevant skills, and experience the records of cloud carriers.

**Automatic Updates and Patch Management:** Cloud providers manipulate software application updates, patches, and protection functions robotically. This ensures that clients always have to get proper entry to the required abilities and protection upgrades without the need for guide intervention.

**Disadvantages of Cloud Service Models**

**Security Concerns:** Security remains a top concern for companies moving to the cloud. Storing information and programs on out-of-door servers will increase questions on statistics' privateness, regulatory compliance, and the functionality of unauthorized access.

**Dependency on Internet Connectivity:** Cloud services require a reliable internet connection. Downtime or disruptions in internet connectivity can impact the right to access essential applications and information, affecting business operations.

**Limited Customization in SaaS:** While SaaS offers convenience, it is able to lack the extent of customization that a few organizations require. Users depend on the capabilities and configurations supplied by the useful resources of the SaaS company, restricting flexibility.

**Data Transfer Costs:** Moving huge volumes of records from the cloud can require extra charges. Organizations need to cautiously recollect and manipulate facts and switch fees, in particular at the same time as dealing with enormous amounts of records.

**Vendor Lock-In:** Adopting certain cloud providers can also result in provider lock-in, wherein it becomes hard to migrate packages and statistics to a different employer or again to on-premises surroundings. This can limit flexibility and cause lengthy periods of dependence on a specific cloud organization.

**Potential for Downtime:** Cloud company companies may also experience outages or downtime, impacting the supply of services. While respectable businesses try for immoderate availability, occasional disruptions can occur, affecting users who get proper entry to agency continuity.

**CLOUD SERVICE EXAMPLES**

**Examples of Cloud Storage**

**Ex: Dropbox, Gmail, Facebook**

The number of cloud storage providers online seems to grow every day. Each competing over the amount of storage they can provide to clients.

Right now, Dropbox is the clear leader in streamlined cloud storage allowing users to **access files on any device through its application or website with up to 1 terabyte of free storage.**

Google’s email service provider Gmail, on the other hand, provides unlimited storage on the cloud. Gmail has revolutionized the way we send emails and largely responsible for the increased usage of email worldwide.

Facebook is a mix of the two, in that it can store an infinite amount of information, images, and videos on your profile. They can then be easily accessed on multiple devices. Facebook goes a step further with their Messenger app, which allows for profiles to exchange data.

**Examples of Marketing Cloud Platforms**

**Ex: Maropost for Marketing, Hubspot, Adobe Marketing Cloud**

A marketing cloud is an end-to-end digital marketing platform for clients to manage contacts and target leads. **Maropost Marketing Cloud**combines easy-to-use marketing automation and hyper-targeting of leads. At the same time, ensuring emails actually arrive in the inbox, thanks to its advanced email deliverability capabilities.

In general, marketing clouds fulfill a need for personalization. This is important in a market that demands messaging be “more human.” That’s why communicating that your brand is here to help, will make all the difference in closing.

**Examples of Cloud Computing in Education**

**Ex: SlideRocket, Ratatype, Amazon Web Servies**

Education is increasingly adopting advanced technology because students already are. So, in an effort to modernize classrooms, educators have introduced e-learning software like SlideRocket.

SlideRocket is a platform that students can use to **build presentations and submit them. Students can even present them through web conferencing all on the cloud.** Another tool teachers use is Ratatype, which helps students learn to type faster and offers online typing tests to track their progress.

For school administration, Amazon’s AWS Cloud for K12 and Primary Education features a virtual desktop infrastructure (VDI) solution. Through the cloud, allows instructors and students to access teaching and learning software on multiple devices.

**Examples of Cloud Computing in Healthcare**

**Ex: CleanDATA Dell’s Secure Healthcare Cloud, IBM Cloud**

Cloud computing lets nurses, physicians, and administrators share information quickly from anywhere. It also saves on costs by allowing large data files to be shared instantly for maximum convenience. This is a major boost for efficiency.

Ultimately, **cloud technology ensures patients receive the best possible care without unnecessary delay**. The patient’s condition can also be updated in seconds through remote conferencing.

However, many modern hospitals have yet to implement cloud computing but are forecasted to do so in the near future.

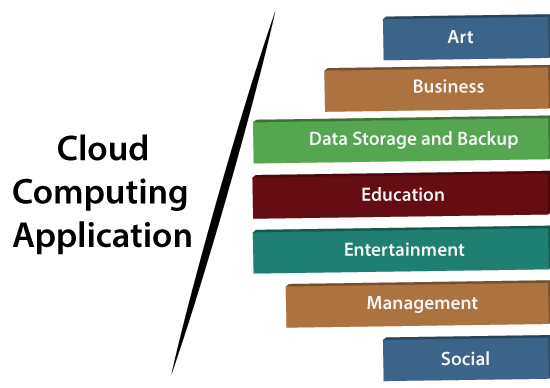
**Examples of Cloud Computing for Government**

**Uses: IT consolidation, shared services, citizen services**

The **U.S. government and military were early adopters of cloud computing.** The U.S. Federal Cloud Computing Strategy, introduced under the Obama administration, was instituted to accelerate cloud adoption in all departments.

**CLOUD BASED SERVICES & APPLICATIONS**

Cloud service providers provide various applications in the field of art, business, data storage and backup services, education, entertainment, management, social networking, etc. The most widely used cloud computing applications are given below.

**1. Art Applications -** Cloud computing offers various art applications for quickly and easily design attractive cards, booklets, and images. Some most commonly used cloud art applications are given below:

**(i) Moo**

Moo is one of the best cloud art applications. It is used for designing and printing business cards, postcards, and mini cards.

**(ii). Vistaprint**

Vistaprint allows us to easily design various printed marketing products such as business cards, Postcards, Booklets, and wedding invitations cards.

**(iiI), Adobe Creative Cloud**

Adobe creative cloud is made for designers, artists, filmmakers, and other creative professionals. It is a suite of apps which includes PhotoShop image editing programming, Illustrator, InDesign, TypeKit, Dreamweaver, XD, and Audition.

**2. Business Applications**

Business applications are based on cloud service providers. Today, every organization requires the cloud business application to grow their business. It also ensures that business applications are 24\*7 available to users.

There are the following business applications of cloud computing -

**(i). MailChimp**

MailChimp is an **email publishing platform** which provides various options to **design, send,** and **save** templates for emails.

**(ii). Salesforce**

Salesforce platform provides tools for sales, service, marketing, e-commerce, and more. It also provides a cloud development platform.

**(iii) Chatter**

Chatter helps us to **share important information** about the organization in real time.

**(iv). Bitrix24**

Bitrix24 is a **collaboration** platform which provides communication, management, and social collaboration toos.

**(v). Paypal**

Paypal offers the simplest and easiest **online payment** mode using a secure internet account. Paypal accepts the payment through debit cards, credit cards, and also from Paypal account holders.

**(vi). Slack**

Slack stands for **Searchable Log of all Conversation and Knowledge**. It provides a **user-friendly** interface that helps us to create public and private channels for communication.

**(viii). Quickbooks**

Quickbooks works on the terminology "**Run Enterprise anytime, anywhere, on any device**." It provides online accounting solutions for the business. It allows more than 20 users to work simultaneously on the same system.

**4. Data Storage and Backup Applications**

Cloud computing allows us to store information (data, files, images, audios, and videos) on the cloud and access this information using an internet connection. As the cloud provider is responsible for providing security, so they offer various backup recovery application for retrieving the lost data. A list of data storage and backup applications in the cloud are given below –

**(i). Box.com**

Box provides an online environment for **secure content management, workflow,** and **collaboration**. It allows us to store different files such as Excel, Word, PDF, and images on the cloud. The main advantage of using box is that it provides drag & drop service for files and easily integrates with Office 365, G Suite, Salesforce, and more than 1400 tools.

**(ii)**. **Mozy**

Mozy provides powerful **online backup solutions** for our personal and business data. It schedules automatically back up for each day at a specific time.

**(iii). Joukuu**

Joukuu provides the simplest way to **share** and **track cloud-based backup files**. Many users use joukuu to search files, folders, and collaborate on documents.

**(iv). Google G Suite**

Google G Suite is one of the best **cloud storage** and **backup** application. It includes Google Calendar, Docs, Forms, Google+, Hangouts, as well as cloud storage and tools for managing cloud apps. The most popular app in the Google G Suite is Gmail. Gmail offers free email services to users.

**4. Education Applications**

Cloud computing in the education sector becomes very popular. It offers various online distance learning platforms and student information portals to the students. The advantage of using cloud in the field of education is that it offers strong virtual classroom environments, Ease of accessibility, secure data storage, scalability, greater reach for the students, and minimal hardware requirements for the applications. There are the following education applications offered by the cloud.

**(i). Google Apps for Education**

Google Apps for Education is the most widely used platform for free web-based email, calendar, documents, and collaborative study.

**(ii). Chromebooks for Education**

Chromebook for Education is one of the most important Google's projects. It is designed for the purpose that it enhances education innovation.

(**iii). Tablets with Google Play for Education**

It allows educators to quickly implement the latest technology solutions into the classroom and make it available to their students.

(**iv). AWS in Education**

AWS cloud provides an education-friendly environment to universities, community colleges, and schools.

**5. Entertainment Applications**

Entertainment industries use a multi-cloud strategy to interact with the target audience. Cloud computing offers various entertainment applications such as online games and video conferencing.

**(i).** o**nline games**

Today, cloud gaming becomes one of the most important entertainment media. It offers various online games that run remotely from the cloud. The best cloud gaming services are Shaow, GeForce Now, Vortex, Project xCloud, and PlayStation Now.

(**ii). Video Conferencing Apps**

Video conferencing apps provides a simple and instant connected experience. It allows us to communicate with our business partners, friends, and relatives using a cloud-based video conferencing. The benefits of using video conferencing are that it reduces cost, increases efficiency, and removes interoperability.

**6. Management Applications**

Cloud computing offers various cloud management tools which help admins to manage all types of cloud activities, such as resource deployment, data integration, and disaster recovery. These management tools also provide administrative control over the platforms, applications, and infrastructure. Some important management applications are:

**(i). Toggl**

Toggl helps users to track allocated time period for a particular project.

**(ii). Evernote**

Evernote allows you to sync and save your recorded notes, typed notes, and other notes in one convenient place. It is available for both free as well as a paid version. It uses platforms like Windows, macOS, Android, iOS, Browser, and Unix.

**(iii). Outright**

Outright is used by management users for the purpose of accounts. It helps to track income, expenses, profits, and losses in real-time environment.

**(iv). GoToMeeting**

GoToMeeting provides **Video Conferencing** and **online meeting apps**, which allows you to start a meeting with your business partners from anytime, anywhere using mobile phones or tablets. Using GoToMeeting app, you can perform the tasks related to the management such as join meetings in seconds, view presentations on the shared screen, get alerts for upcoming meetings, etc.

**7. Social Applications**

Social cloud applications allow a large number of users to connect with each other using social networking applications such as **Facebook, Twitter, Linkedln,** etc. There are the following cloud based social applications.

(**i). Facebook**

Facebook is a **social networking website** which allows active users to share files, photos, videos, status, more to their friends, relatives, and business partners using the cloud storage system. On Facebook, we will always get notifications when our friends like and comment on the posts.

**(ii). Twitter**

Twitter is a **social networking** site. It is a **microblogging** system. It allows users to follow high profile celebrities, friends, relatives, and receive news. It sends and receives short posts called tweets.

(**iii). Yammer**

Yammer is the **best team collaboration** tool that allows a team of employees to chat, share images, documents, and videos.

(**iv). LinkedIn**

LinkedIn is a **social network** for students, freshers, and professionals.

**CLOUD CONCEPTS AND TECHNOLOGIES**

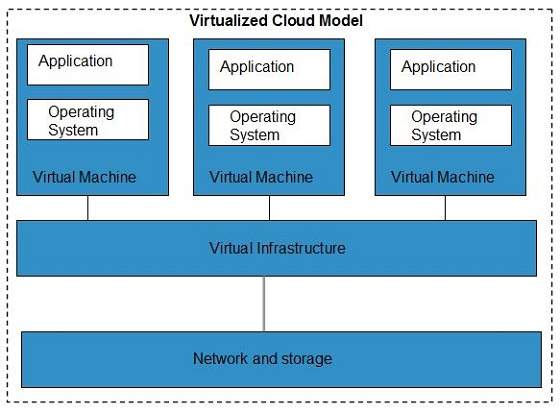
There are certain technologies working behind the cloud computing platforms making cloud computing flexible, reliable, and usable. These technologies are listed below:

* Virtualization
* Service-Oriented Architecture (SOA)
* Grid Computing
* Utility Computing

**Virtualization**

Virtualization is a technique, which allows to share single physical instance of an application or resource among multiple organizations or tenants (customers). It does this by assigning a logical name to a physical resource and providing a pointer to that physical resource when demanded.

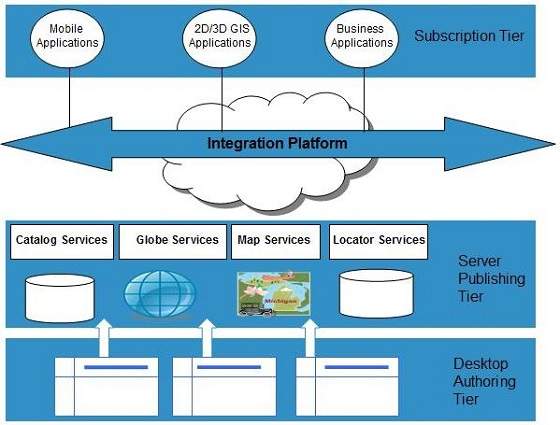
The Multitenant architecture offers virtual isolation among the multiple tenants. Hence, the organizations can use and customize their application as though they each have their instances running.



**Service-Oriented Architecture (SOA)**

Service-Oriented Architecture helps to use applications as a service for other applications regardless the type of vendor, product or technology. Therefore, it is possible to exchange the data between applications of different vendors without additional programming or making changes to services.

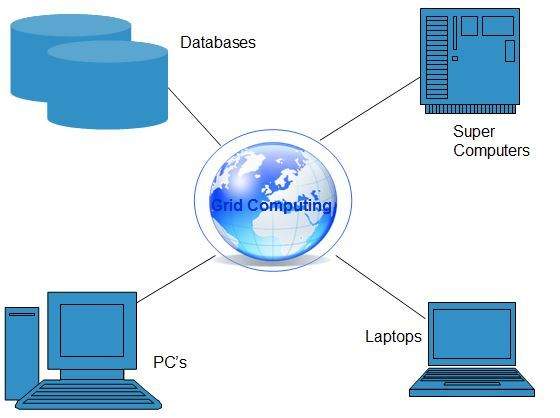
The cloud computing service oriented architecture is shown in the diagram below.



**Grid Computing**

**Grid Computing** refers to distributed computing, in which a group of computers from multiple locations are connected with each other to achieve a common objective. These computer resources are heterogeneous and geographically dispersed.

Grid Computing breaks complex task into smaller pieces, which are distributed to CPUs that reside within the grid.



Ezoic

**Utility Computing**

Utility computing is based on **Pay-per-Use model.** It offers computational resources on demand as a metered service. Cloud computing, grid computing, and managed IT services are based on the concept of utility computing.

**Big Data and Analytics**

An important component of cloud computing is big data and analytics. Here are some details on it:

**Big Data** is the term used to describe big and complicated data collections that are difficult to process or analyze using conventional data processing techniques. **Analytics describes** the procedure of studying data and drawing conclusions from it.



**Features:**

* **Volume:** Big Data is characterized by its enormous volume, which is frequently measured in terabytes or petabytes.
* **Veracity:** Big Data is frequently characterized by its incompleteness and uncertainty.
* **Velocity:** Big Data must be processed in real-time or very close to real-time because it is produced quickly.
* **Variety:** There are many different types of big data, including structured, semi-structured, and unstructured data.

A scalable and affordable platform for Big Data processing and analytics is provided by cloud computing.

**Benefits:**

* **Cost-Effectiveness:** Using cloud computing, businesses only pay for the resources they really use, which lowers the cost of processing and analyzing Big Data.
* **Scalability:** The infrastructure offered by cloud computing is scalable and can meet the needs for processing and storing Big Data.
* **Simple Integration:** Big Data technologies and frameworks like Hadoop and Spark are simple to integrate with cloud computing systems.
* **Real-time Analytics:** Thanks to cloud computing, businesses can perform near-real-time analytics and act on data fast.

**Serverless Computing:**

A cloud computing architecture known as "serverless computing," or "Function-as-a-Service" (FaaS), relies on the cloud provider to manage the infrastructure required to run and scale applications while the user only needs to concentrate on creating and distributing code. In serverless computing, the user creates actions that are triggered by specific occurrences, such as HTTP requests or database changes, and the cloud provider automatically sets up and scales the necessary computing resources to carry out those actions.

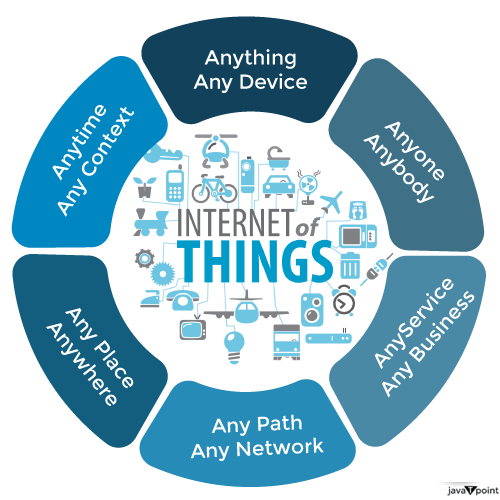
Features:

* **No Management of Infrastructure:** Users may concentrate on writing and delivering code because the cloud provider manages the network, servers, and operating systems that make up the underlying infrastructure.
* **Reduced Time-to-Market:** Serverless computing enables developers to deploy and scale their applications quickly without worrying about infrastructure setup and management, which can reduce the time it takes to market new features and applications.
* **Event-Driven Architecture:** Serverless computing is made to react to particular occurrences like user requests, database changes, or predetermined events.
* **Pay-per-use Pricing:** Serverless computing allows customers to only pay for the resources they really use to carry out their duties, which can result in cost savings.
* **Scalability and High Availability:** Serverless computing platforms can dynamically assign computer resources based on the demand for processing, making them extremely scalable.

**Benefits of Serverless Computing**

* **Faster Time to Market:** Because serverless computing does not require infrastructure management, developers can concentrate on building code and releasing their apps more quickly.
* **Better Scalability:** Serverless computing automatically scales based on the volume of incoming requests, ensuring that your application can handle sudden increases in traffic without requiring any manual intervention.
* **Reduced Operational Overhead:** Since server provisioning, configuration, and maintenance are handled entirely by the cloud provider with serverless computing, developers don't have to worry about it. Developers can concentrate on writing code as a result of the decreased operational overhead.
* **Savings:** Because serverless computing uses a pay-per-invocation business model, you only pay for the resources your function really consumes. Especially for applications with intermittent or unexpected workloads, this might result in significant cost savings.

Internet Of Things (IoT)



**Cloud-Based IoT Platforms and Services**

The management and processing of data produced by connected devices in the Internet of Things (IoT) ecosystem is the goal of cloud-based IoT platforms and services. With the help of these services, IoT device data may be stored, processed, and analyzed in a secure and scalable architecture.

Features:

**Processing and Analysing Data from IoT Devices:** Cloud-based IoT systems and services offer strong analytics tools, such as real-time data stream processing, batch processing, and machine learning.

**Security:** Cloud-based IoT systems and services include strong security capabilities, such as encryption, access control, and threat detection, to shield data and devices from cyber threats.

**Device Management:** Tools for managing a lot of devices, including provisioning, monitoring, and firmware updates, are offered by cloud-based IoT platforms and services.

**Data Intake and Storage:** To handle the enormous amounts of data created by IoT devices, cloud-based IoT platforms, and services offer scalable and adaptable storage options.

**The Following are Some Advantages of Employing Cloud-Based IoT Platforms and Services:**

* **Scalability:** To accommodate the rising volume of data produced by IoT devices, cloud-based IoT systems, and services can easily scale up or down.
* **Speed to Market:** IoT platforms and services that are cloud-based provide a quick and simple way to develop and manage IoT applications, enabling businesses to swiftly launch new goods and services.
* **Flexibility:** To accommodate many use cases, including real-time analytics, predictive maintenance, and anomaly detection, cloud-based IoT platforms and services provide a wide range of storage and processing possibilities.
* **Cost-Effectiveness:** Pay-as-you-go pricing models are available on cloud-based IoT platforms and services, enabling businesses to only pay for the resources they actually use rather than making a large upfront investment in infrastructure and hardware.

The platforms and services offered by AWS IoT, Microsoft Azure IoT, Google Cloud IoT, and IBM Watson IoT are a few examples of cloud-based IoT solutions.

**Numerous Advantages of Cloud-Based IoT Solutions Include:**

* **Cost Savings:** Organisations no longer need to invest in pricey infrastructure to support their IoT deployments thanks to cloud-based IoT solutions. Instead, companies can make use of the IoT platform provider's cloud infrastructure.
* **Scalability:** Depending on the amount of data provided by IoT devices, cloud-based IoT platforms can scale up or down. This enables businesses to manage massive amounts of data and scale up their IoT implementations as necessary.
* **Enhanced Security Features**, including encryption, multi-factor authentication, and access control, are available on cloud-based IoT solutions. This aids in shielding sensitive IoT data from online dangers.
* **Real-Time Data Analysis:** Organisations may swiftly make educated decisions with the help of real-time data analysis offered by cloud-based IoT technologies. Real-time insights can assist in preventing equipment failures and downtime in applications like predictive maintenance, where they are very helpful.
* **Increased Adaptability:** Organisations may tailor their IoT installations to suit their unique requirements thanks to the great degree of adaptability offered by cloud-based IoT solutions. This includes the option to customize the data processing and analytics tools as well as a range of IoT sensors and devices.

**19IINFCX1: CLOUD COMPUTING**

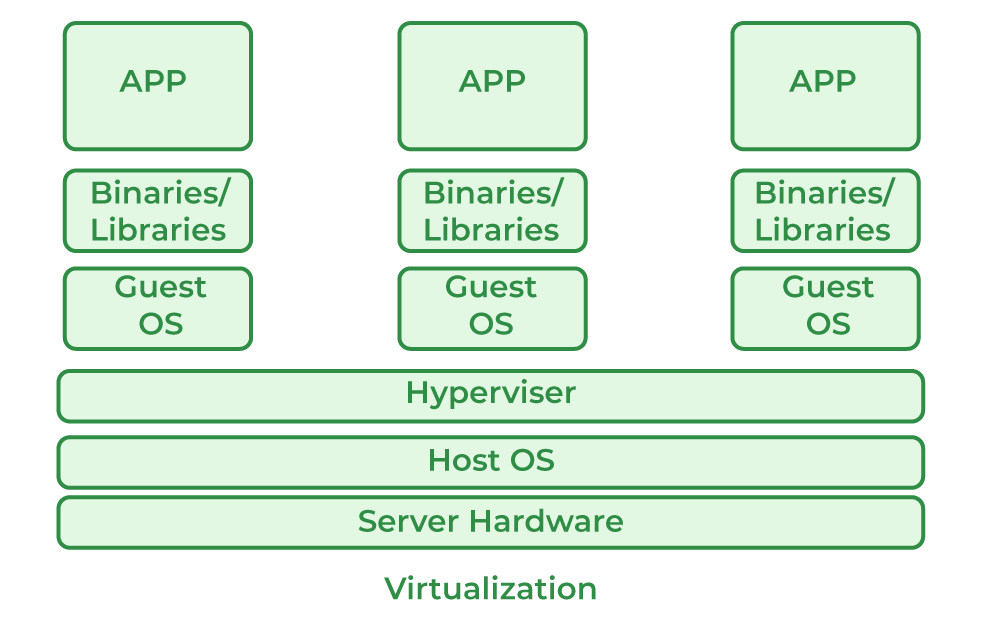
**Unit-2**

**Virtualization, Cloud Services and Platforms:** Virtualization:Virtualization- Characteristics taxonomy- types- Pros and Cons- Examples Architecture: Reference model- types of clouds- Compute Service - Storage Services - Cloud Database Services - Application Services - Content Delivery Services - Analytics Services - Deployment And Management Service - Identity And Access Management Services - Open Source Private Cloud Software.

**VIRTUALIZATION**

**Virtualization**is used to create a virtual version of an underlying service With the help of Virtualization, multiple operating systems and applications can run on the same machine and its same hardware at the same time, increasing the utilization and flexibility of hardware. It was initially developed during the mainframe era.

It is one of the main cost-effective, hardware-reducing, and energy-saving techniques used by cloud providers. Virtualization allows sharing of a single physical instance of a resource or an application among multiple customers and organizations at one time. It does this by assigning a logical name to physical storage and providing a pointer to that physical resource on demand. The term virtualization is often synonymous with hardware virtualization, which plays a fundamental role in efficiently delivering Infrastructure-as-a-Service (IaaS) solutions for cloud computing. Moreover, virtualization technologies provide a virtual environment for not only executing applications but also for storage, memory, and networking.



* Host Machine: The machine on which the virtual machine is going to be built is known as Host Machine.
* Guest Machine: The virtual machine is referred to as a Guest Machine.

Virtualization has a prominent impact on Cloud Computing. In the case of cloud computing, users store data in the cloud, but with the help of Virtualization, users have the extra benefit of sharing the infrastructure. Cloud Vendors take care of the required physical resources, but these cloud providers charge a huge amount for these services which impacts every user or organization. Virtualization helps Users or Organisations in maintaining those services which are required by a company through external (third-party) people, which helps in reducing costs to the company. This is the way through which Virtualization works in Cloud Computing.

**Benefits of Virtualization**

* More flexible and efficient allocation of resources.
* Enhance development productivity.
* It lowers the cost of IT infrastructure.
* Remote access and rapid scalability.
* High availability and disaster recovery.
* Pay peruse of the IT infrastructure on demand.
* Enables running multiple operating systems.

**Drawback of Virtualization**

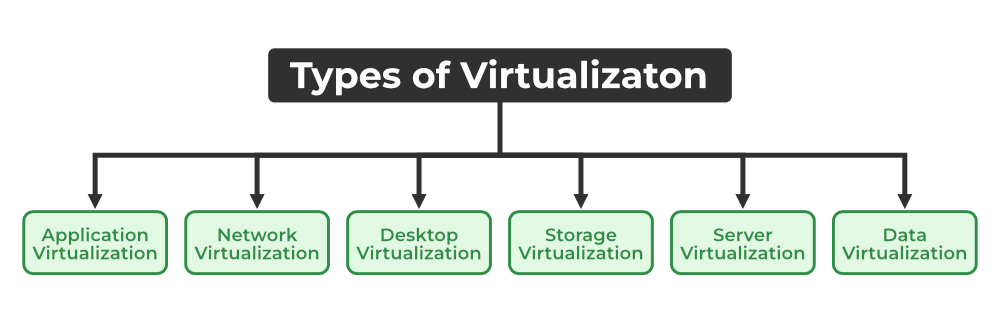
* **High Initial Investment:**Clouds have a very high initial investment, but it is also true that it will help in reducing the cost of companies.
* **Learning New Infrastructure:** As the companies shifted from Servers to Cloud, it requires highly skilled staff who have skills to work with the cloud easily, and for this, you have to hire new staff or provide training to current staff.
* **Risk of Data:**Hosting data on third-party resources can lead to putting the data at risk, it has the chance of getting attacked by any hacker or cracker very easily.

**CHARACTERISTICS OF VIRTUALIZATION**

* **Increased Security:** The ability to control the execution of a guest program in a completely transparent manner opens new possibilities for delivering a secure, controlled execution environment. All the operations of the guest programs are generally performed against the virtual machine, which then translates and applies them to the host programs.
* **Managed Execution:** In particular, sharing, aggregation, emulation, and isolation are the most relevant features.
* **Sharing:** Virtualization allows the creation of a separate computing environment within the same host.
* **Aggregation:** It is possible to share physical resources among several guests, but virtualization also allows aggregation, which is the opposite process.
* **Isolation**
* **Encapsulation**
* **Hardware Independence**
* **Resource Sharing**
* **Snapshot and Cloning**
* **Dynamic Resource Allocation:**
* **High Availability and Fault Tolerance**
* **Scalability**

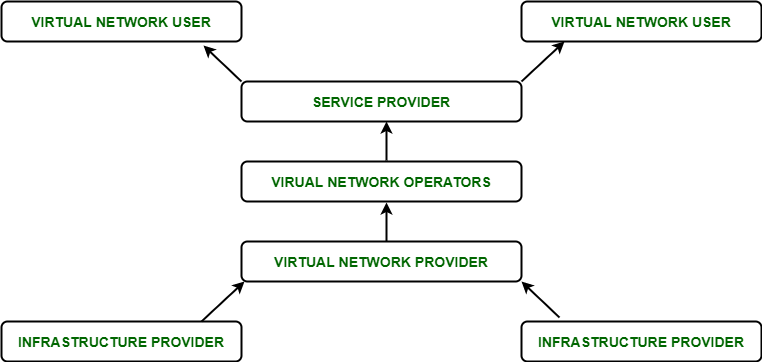
**TYPES OF VIRTUALIZATION**

1. Application Virtualization
2. Network Virtualization
3. Desktop Virtualization
4. Storage Virtualization
5. Server Virtualization
6. Data virtualization



**1.** **Application Virtualization:** Application virtualization helps a user to have remote access to an application from a server. The server stores all personal information and other characteristics of the application but can still run on a local workstation through the internet. An example of this would be a user who needs to run two different versions of the same software. Technologies that use application virtualization are hosted applications and packaged applications.

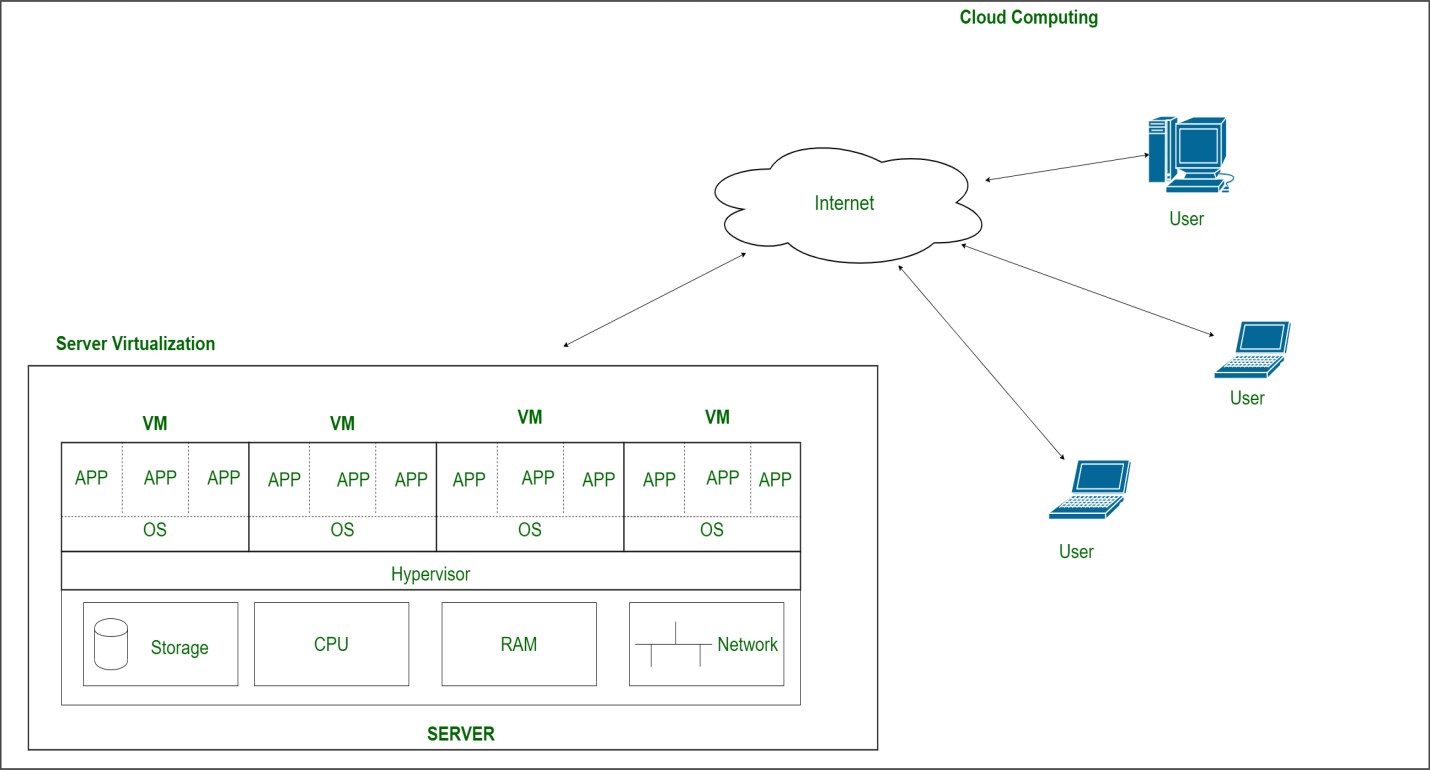
**2.** **Network Virtualization:** The ability to run multiple virtual networks with each having a separate control and data plan. It co-exists together on top of one physical network. It can be managed by individual parties that are potentially confidential to each other. Network virtualization provides a facility to create and provision virtual networks, logical switches, routers, firewalls, load balancers, Virtual Private Networks (VPN), and workload security within days or even weeks.



**3.** **Desktop Virtualization:**Desktop virtualization allows the users’ OS to be remotely stored on a server in the data center. It allows the user to access their desktop virtually, from any location by a different machine. Users who want specific operating systems other than Windows Server will need to have a virtual desktop. The main benefits of desktop virtualization are user mobility, portability, and easy management of software installation, updates, and patches.

**4.** **Storage Virtualization:** Storage virtualization is an array of servers that are managed by a virtual storage system. The servers aren’t aware of exactly where their data is stored and instead function more like worker bees in a hive. It makes managing storage from multiple sources be managed and utilized as a single repository. storage virtualization software maintains smooth operations, consistent performance, and a continuous suite of advanced functions despite changes, breaks down, and differences in the underlying equipment.

**5.** **Server Virtualization:**This is a kind of virtualization in which the masking of server resources takes place. Here, the central server (physical server) is divided into multiple different virtual servers by changing the identity number, and processors. So, each system can operate its operating systems in an isolated manner. Where each sub-server knows the identity of the central server. It causes an increase in performance and reduces the operating cost by the deployment of main server resources into a sub-server resource. It’s beneficial in virtual migration, reducing energy consumption, reducing infrastructural costs, etc.



**6.** **Data Virtualization:**This is the kind of virtualization in which the data is collected from various sources and managed at a single place without knowing more about the technical information like how data is collected, stored & formatted then arranged that data logically so that its virtual view can be accessed by its interested people and stakeholders, and users through the various cloud services remotely. Many big giant companies are providing their services like Oracle, IBM, At scale, Cdata, etc.

**Uses of Virtualization**

* Data-integration
* Business-integration
* Service-oriented architecture data-services
* Searching organizational data

**PROS AND CONS OF VIRTUALIZATION**

**Pros of Virtualization**

* **Utilization of Hardware Efficiently**: With the help of Virtualization Hardware is Efficiently used by user as well as Cloud Service Provider. In this the need of Physical Hardware System for the User is decreases and this results in less costly.
* **High Availability**: One of the main benefit of Virtualization is that it provides advance features which allow virtual instances to be available all the times.
* **Disaster Recovery** is efficient and easy: With the help of virtualization Data Recovery, Backup, Duplication becomes very easy. In traditional method , if somehow due to some disaster if Server system Damaged then the surety of Data Recovery is very less. But with the tools of Virtualization real time data backup recovery and mirroring become easy task and provide surety of zero percent data loss.
* **Virtualization saves Energy:** Virtualization will help to save Energy because while moving from physical Servers to Virtual Server’s, the number of Server’s decreases due to this monthly power and cooling cost decreases which will Save Money as well.
* **Quick and Easy Set up:** In traditional methods Setting up physical system and servers are very time-consuming. Firstly Purchase them in bulk after that wait for shipment. When Shipment is done then wait for Setting up and after that again spend time in installing required software etc. Which will consume very time. But with the help of virtualization the entire process is done in very less time which results in productive setup.
* **Cloud Migration becomes easy**: Most of the companies those who already have spent a lot in the server have a doubt of Shifting to[Cloud.](https://www.geeksforgeeks.org/cloud-based-services/)But it is more cost-effective to shift to cloud services because all the data that is present in their server’s can be easily migrated into the cloud server and save something from maintenance charge, power consumption, cooling cost, cost to Server Maintenance Engineer etc.
* **Resource Optimization**: Virtualization allows efficient utilization of physical hardware by running multiple virtual machines (VMs) on a single physical server. This consolidation leads to cost savings in terms of hardware, power, cooling, and space

**Cons of Virtualization**

* **High Initial Investment:** While virtualization reduces costs in the long run, the initial setup costs for storage and servers can be higher than a traditional setup.
* **Complexity:** Managing virtualized environments can be complex, especially as the number of VMs increases.
* **Security Risks:** Virtualization introduces additional layers, which may pose security risks if not properly configured and monitored.
* **Learning New Infrastructure:** As Organization shifted from Servers to Cloud. They required skilled staff who can work with cloud easily. Either they hire new IT staff with relevant skill or provide training on that skill which increase the cost of company.
* **Data can be at Risk:** Working on virtual instances on shared resources means that our data is hosted on third party resource which put’s our data in vulnerable condition. Any hacker can attack on our data or try to perform unauthorized access. Without Security solution our data is in threaten situation.

**CLOUD COMPUTING AND VIRTUALIZATION ARCHITECTURE**

Virtualization in cloud computing helps create virtual versions of hardware such as desktop computers with a virtual ecosystem of operating systems, storage, memory, and networking services. Virtualization architecture uses the same hardware to run multiple operating systems on the same machine and optimize their performance.

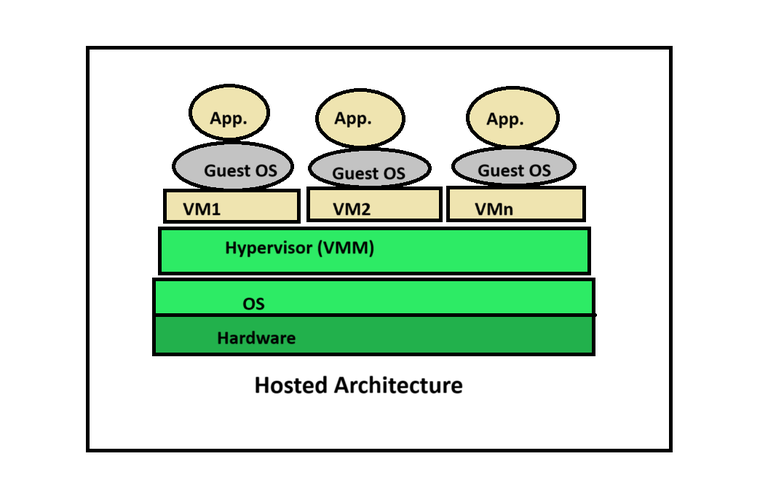
Virtualization and virtualization architecture are important concepts in cloud computing. In fact, since the definition of cloud computing also includes virtual ecosystems, these terms are often used interchangeably. Whether the ecosystem is private (i.e., cloud) or public (public cloud), virtualization reduces the need for organizations to maintain physical (on-premises) infrastructure for their computing needs. With cloud computing and virtualization architecture, applications can be shared with many active users. With a public cloud like Amazon Web Services (AWS) or Microsoft Azure, these can be shared with multiple businesses.

**Types of Virtualization Architectures**

There are two main types of virtualization architectures: hosted and bare metal.

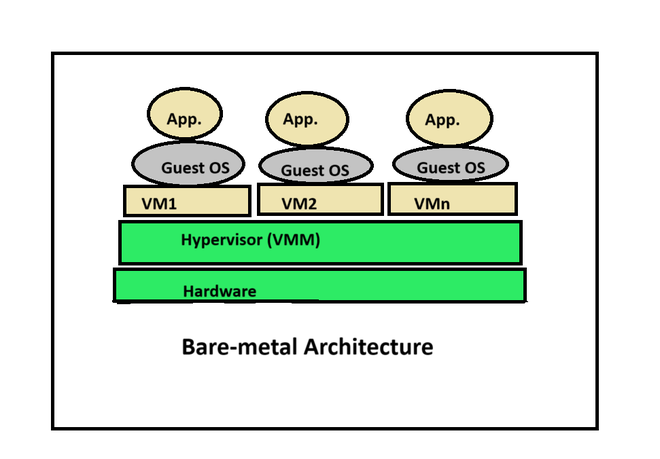
**Hosted Architecture**

In this type of configuration, first, the host operating system is installed on the hardware, then the software is installed. The software is a hypervisor or virtual machine (VM) that requires many guest operating systems or VMs to be installed on the hardware to set up the virtualization architecture. Once the hypervisor is in place, applications can be installed and run on the virtual machine as if they were installed on the physical machine.



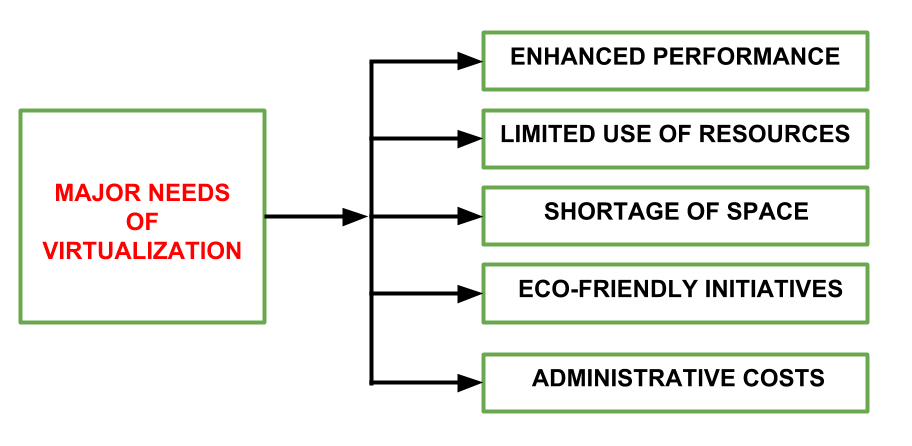
**Bare Metal Architecture**

In this architecture, the hypervisor is installed directly on the hardware, not on top of the operating system. Hypervisors and virtual machines are configured the same way as infrastructure. Bare metal virtualization architecture is designed for applications that provide real-time access or perform some form of data processing.

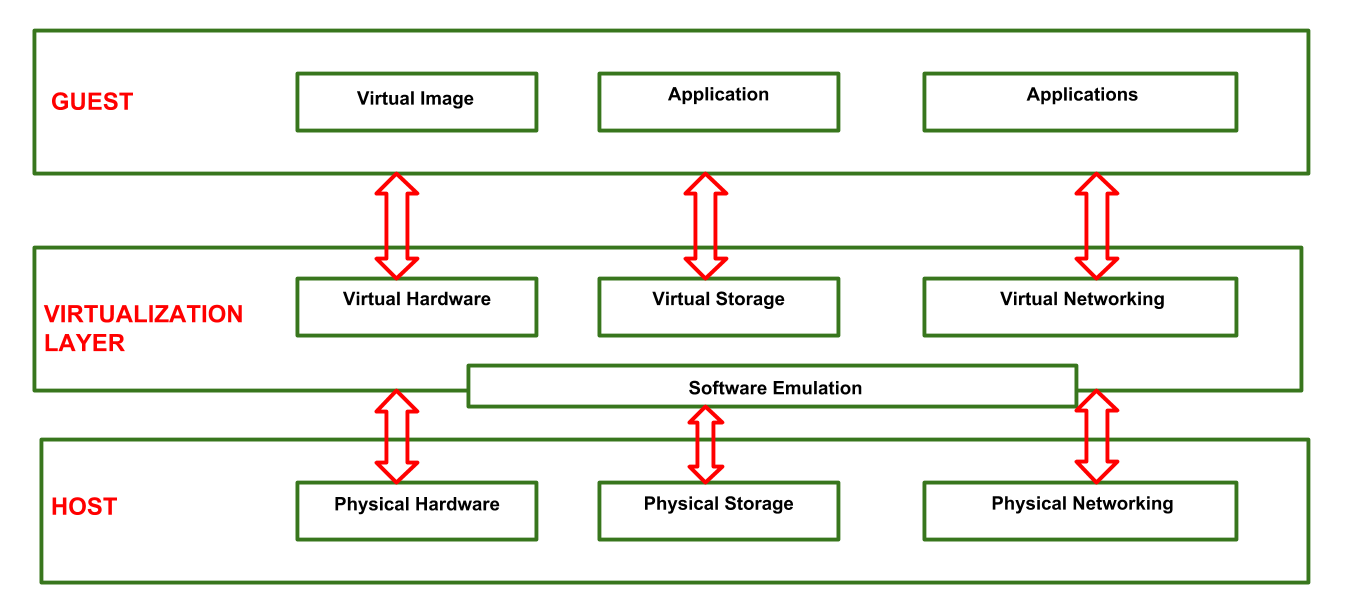


**REFERENCE MODEL FOR VIRTUALIZATION**

There are five major needs of virtualization which are described below:



**VIRTUALIZATION REFERENCE MODEL**



Three major Components falls under this category in a virtualized environment:

**1. Guest**

The guest represents the system component that interacts with the virtualization layer rather than with the host, as would normally happen. Guests usually consist of one or more virtual disk files, and a VM definition file. Virtual Machines are centrally managed by a host application that sees and manages each virtual machine as a different application.

**2. Host**  
The host represents the original environment where the guest is supposed to be managed. Each guest runs on the host using shared resources donated to it by the host. The operating system, works as the host and manages the physical resource management, and the device support.

**3. Virtualization Layer**

The virtualization layer is responsible for recreating the same or a different environment where the guest will operate. It is an additional abstraction layer between a network and storage hardware, computing, and the application running on it. Usually it helps to run a single operating system per machine which can be very inflexible compared to the usage of virtualization.

**COMPUTE AS A SERVICE (CAAS)**

Compute as a Service (CaaS) is a consumption-based (pay-per-use) infrastructure model that provides on-demand processing resources for general and specific workloads. CaaS lets enterprises simplify and scale compute operations to eliminate over provisioning and add flexibility for new or unexpected demands.

**How does Compute as a Service work?**

CaaS is a cloud-based solution that relies on virtual and physical processing power. Physical processing takes place in private, on-premises servers, while virtual processing occurs in the cloud. Compute resources can include general, high-speed graphics processing (GPU) for machine learning and artificial intelligence or high-performance computing (HPC) for raw processing power. The exact infrastructure configuration will vary from enterprise to enterprise, depending on their precise needs, and this infrastructure can scale up or down over time. Providers may offer this service as a flat-rate subscription or on a fluid consumption-based model so customers are charged only for the compute they use.

**Benefits of Compute as a Service**

CaaS can be a game-changer for enterprises looking to accelerate their digital transformation, offering a solution that’s more cost efficient, flexible, and streamlined.

Compared to building your cloud from scratch, which can be labor and capital intensive, CaaS doesn’t require as high an upfront investment in hardware, cloud resources, and man-hours. Instead, CaaS delivers workload-optimized systems at a faster rate to your data center or edge location—and at a fraction of the cost of a self-managed or legacy solution.

CaaS solutions can be scaled over time. Private, on-premises IT infrastructure is often overprovisioned, meaning it’s fixed to accommodate a wide range of workloads and spikes in demand. The problem? Those resources aren’t always used, and any required expansion can result in constrained resources or extended downtime. CaaS mitigates those concerns with on-demand configuration allocation that can be scaled up or down in response to new opportunities and unexpected challenges, helping maintain compute bandwidth and the teams that rely on it.

No matter the requirements, CaaS can be provisioned for virtually any workload before it’s needed—general-purpose compute, composable infrastructure, mission-critical applications, data analytics, and more. These preconfigured solutions can be deployed across several tiers and scales. And since CaaS is typically a managed solution covering installation to maintenance and support, enterprises can refocus their teams to refocus on higher-level tasks and innovation.

**Some examples of Compute as a Service**

While the name implies pure processing power, CaaS has a multitude of applications, ranging from basic compute and cloud computing needs to Big Data and compute security. By far the most common is cloud computing, which delivers software and applications accessible to end users outside the server via an Internet connection. In some cases, configurations can be optimized for specific workloads. These workloads can be put into public cloud, which is ideal for shared resources and collaboration, or protected behind private cloud for optimal security and compliance.

CaaS can also help enterprises get more from Big Data by deepening your data analytics infrastructure, transforming your data using rules and models and unlocking new insights faster from data-collecting devices. These insights can be gleaned in real time from the data center, colocations, and at the edge.

But CaaS can do more than crunch numbers; it can also protect invaluable IT infrastructure. Compute can provide security features like zero-trust provisioning, cryptographic certificates, and zero-touch onboarding, including automated protection that detects malware and other threats before they cause harm or recover a compromised server. Security may also be applied to the supply chain, extending from manufacturing to installation.

**Underlying Technologies and Components of CaaS**

Here are the underlying components of CaaS:

* CaaS platforms use virtualization and hypervisor technologies to create and manage virtual machines (VMs) for hosting containers, improving resource usage and isolation.
* Containerization technologies like Docker are essential for CaaS, providing lightweight and isolated environments for running applications. Container orchestration platforms such as Kubernetes automate container management, deployment, and scaling.
* CaaS abstracts hardware details, allowing users to focus on their applications. Resource allocation mechanisms ensure containers have the computing resources (CPU, memory, storage) they need to run effectively.

These technologies and components work together to provide a scalable and efficient environment for deploying and managing containerized applications in a CaaS model.

**STORAGE SERVICES**

Cloud Storage is a mode of computer data storage in which digital data is stored on servers in off-site locations. The servers are maintained by a third-party provider who is responsible for hosting, managing, and securing data stored on its infrastructure. The provider ensures that data on its servers is always accessible via public or private internet connections.

**How does Cloud Storage work?**

Cloud Storage uses remote servers to save data, such as files, business data, videos, or images. Users upload data to servers via an internet connection, where it is saved on a virtual machine on a physical server. To maintain availability and provide redundancy, cloud providers will often spread data to multiple virtual machines in data centers located across the world. If storage needs increase, the cloud provider will spin up more virtual machines to handle the load. Users can access data in Cloud Storage through an internet connection and software such as web portal, browser, or mobile app via an application programming interface (API).

Cloud Storage is available in four different models:

**Public**

Public Cloud Storage is a model where an organization stores data in a service provider’s data centers that are also utilized by other companies. Data in public Cloud Storage is spread across multiple regions and is often offered on a subscription or pay-as-you-go basis. Public Cloud Storage is considered to be “elastic” which means that the data stored can be scaled up or down depending on the needs of the organization. Public cloud providers typically make data available from any device such as a smartphone or web portal.

**Private**

Private Cloud Storage is a model where an organization utilizes its own servers and data centers to store data within their own network. Alternatively, organizations can deal with cloud service providers to provide dedicated servers and private connections that are not shared by any other organization. Private clouds are typically utilized by organizations that require more control over their data and have stringent compliance and security requirements.

**Hybrid**

A hybrid cloud model is a mix of private and public cloud storage models. A hybrid cloud storage model allows organizations to decide which data it wants to store in which cloud. Sensitive data and data that must meet strict compliance requirements may be stored in a private cloud while less sensitive data is stored in the public cloud. A hybrid cloud storage model typically has a layer of orchestration to integrate between the two clouds. A hybrid cloud offers flexibility and allows organizations to still scale up with the public cloud if need arises.

**Multicloud**

A multicloud storage model is when an organization sets up more than one cloud model from more than one cloud service provider (public or private). Organizations might choose a multicloud model if one cloud vendor offers certain proprietary apps, an organization requires data to be stored in a specific country, various teams are trained on different clouds, or the organization needs to serve different requirements that are not stated in the servicers’ Service Level Agreements. A multicloud model offers organizations flexibility and redundancy.

## Advantages of Cloud Storage

### Total cost of ownership

Cloud Storage enables organizations to move from a capital expenditure to an operational expenditure model, allowing them to adjust budgets and resources quickly.

### Elasticity

Cloud Storage is elastic and scalable, meaning that it can be scaled up (more storage added) or down (less storage needed) depending on the organization’s needs.

### Flexibility

Cloud Storage offers organizations flexibility on how to store and access data, deploy and budget resources, and architect their IT infrastructure.

### Security

Most cloud providers offer robust security, including physical security at data centers and cutting edge security at the software and application levels. The best cloud providers offer zero trust architecture, identity and access management, and encryption.

### Sustainability

One of the greatest costs when operating on-premises data centers is the overhead of energy consumption. The best cloud providers operate on sustainable energy through renewable resources.

### Redundancy

Redundancy (replicating data on multiple servers in different locations) is an inherent trait in public clouds, allowing organizations to recover from disasters while maintaining business continuity.

## Disadvantages of Cloud Storage

### Compliance

Certain industries such as finance and healthcare have stringent requirements about how data is stored and accessed. Some public cloud providers offer tools to maintain compliance with applicable rules and regulations.

### Latency

Traffic to and from the cloud can be delayed because of network traffic congestion or slow internet connections.

### Control

Storing data in public clouds relinquishes some control over access and management of that data, entrusting that the cloud service provider will always be able to make that data available and maintain its systems and security.

### Outages

While public cloud providers aim to ensure continuous availability, outages sometimes do occur, making stored data unavailable.

**How to use Cloud Storage**

Cloud Storage provides several use cases that can benefit individuals and organizations. Whether a person is storing their family budget on a spreadsheet, or a massive organization is saving years of financial data in a highly secure database, Cloud Storage can be used for saving digital data of all kinds for as long as needed.

**Backup**

Data backup is one of the simplest and most prominent uses of Cloud Storage. Production data can be separated from backup data, creating a gap between the two that protects organizations in the case of a cyber threat such as ransomware. Data backup through Cloud Storage can be as simple as saving files to a digital folder such as Google Drive or using block storage to maintain gigabytes or more of important business data.

**Archiving**

The ability to archive old data has become an important aspect of Cloud Storage, as organizations move to digitize decades of old records, as well as hold on to records for governance and compliance purposes. Google Cloud offers several tiers of storage for archiving data, including coldline storage and archival storage, that can be accessed whenever an organization needs them.

**Disaster recovery**

A disaster—natural or otherwise— that wipes out a data center or old physical records needs not be the business-crippling event that it was in the past. Cloud Storage allows for disaster recovery so that organizations can continue with their business, even when times are tough.

**Data processing**

As Cloud Storage makes digital data immediately available, data becomes much more useful on an ongoing basis. Data processing, such as analyzing data for business intelligence or applying machine learning and artificial intelligence to large datasets, is possible because of Cloud Storage.

**Content delivery**

With the ability to save copies of media data, such as large audio and video files, on servers dispersed across the globe, media and entertainment companies can serve their audience low-latency, always available content from wherever they reside.

**CLOUD DATABASE SERVICES**

A cloud database is a database service built and accessed through a cloud computing platform. It serves many of the same functions as a traditional database with the added flexibility of cloud computing. Users install software on a cloud infrastructure to implement the database.

Managing engagement and application data for massive networks of mobile users or remote devices can be a scalability and availability challenge. The problem is that most databases require updates to occur in a central “master” database. This can result in performance bottlenecks and prevent applications from running if the connection to the master database is unavailable.

A cloud database enables organizations to push database access to the farthest edge of the network for mobile devices, remote facilities, sensors and internet-enabled goods. This helps to improve scalability and enable applications to continue running while offline.

**How a cloud database works**

Cloud databases collect, deliver, replicate and push to the edge all of an organization’s data by using the hybrid cloud concept. Users no longer have to deploy the dependent middleware to deliver database requests anywhere in the world. They can connect applications directly to their database.

Hybrid databases create a distributed hybrid data cloud for increased performance, reach, uptime mobility and cost savings so organizations can:

* start small and grow big;
* elastically scale on demand;
* span clusters across multiple data centers;
* manage cloud independently, or let a provider manage it for them; and
* mix and match cloud providers to optimize geographic reach, service level agreements (SLAs), pricing and regulatory requirements.

For example, financial organizations are embracing the hybrid concept by using the database as a central repository for all their disparate data sources, and then delivering this financial data in JSON format. This data is then distributed to the database as a service and replicated to geographic regions across the world.

**Features of a cloud database**

* A database service built and accessed through a cloud platform
* Enables enterprise users to host databases without buying dedicated hardware
* Can be managed by the user or offered as a service and managed by a provider
* Can support relational databases (including MySQL and PostgreSQL) and NoSQL databases (including [MongoDB](https://www.ibm.com/topics/mongodb) and Apache CouchDB)
* Accessed through a web interface or vendor-provided application programming interface (API)

**Benefits of a cloud database**

#### Ease of access

Users can access cloud databases from anywhere, using a vendor’s API or web interface.

#### Scalability

Cloud databases can expand their storage capacities on run-time to accommodate changing needs. Organizations pay only for what they use.

#### Disaster recovery

In the event of a natural disaster, equipment failure or power outage, data is kept secure through backups on remote servers.

**Advantages of cloud database**

A cloud database can accommodate growing data management needs. Organizations can continuously optimize the data layer for cost, performance, security and reach. They can break up their data, distribute it and move it closer to their users. Considerations for a cloud database include:

**Control options:** Users can opt for a virtual machine image managed like a traditional database or a provider’s DBaaS.

**Database technology:** SQL databases are difficult to scale but common. NoSQL databases scale more easily but do not work with some applications.

**Security:** Most cloud database providers encrypt data and provide other security measures.

**Maintenance:** When using a virtual machine image, IT staffers should understand how to maintain the underlying infrastructure.

**APPLICATION SERVICES**

An application server extends the capabilities of a web server by supporting dynamic content generation, application logic, and integration with various resources. It provides a runtime environment where you can run application code and interact with other software components, like messaging systems and databases.

Systems and applications refer to software and computing capabilities that perform specific functions in support of business processes. They vary in technical architecture, operating systems, programming languages, and intended function, posing a challenge to defining and auditing them uniformly.

**1. Art related application**

Cloud computing applications offer a variety of art-related applications for various designing purposes, assisting in the development of eye-catching designs for cards, books, and other visuals. Examples of cloud computing applications for the arts include Moo, Adobe Creative Cloud, and Vistaprint. These programs facilitate quick card creation, printing, and design. Additionally, programs like Adobe Creative Cloud, a cloud-based program that offers first-rate professional editing services, are available.

**2. Image-editing application**

There are several applications available today that offer free image editing. These cloud computing services offer a variety of functions, such as graphic user interfaces and image editing, resizing, cropping, and special effects (GUI). Additionally, these programs include customizable brightness and contrast options. Additionally, they offer highly sophisticated functions that are simple to use. Famous examples include Fotor and Adobe Creative Cloud.

### **3. Data Storage**

Computer programs for data storage are another choice for cloud computing applications. It is also one among the many cloud applications that let you save data, files, photos, and other types of material in the cloud. It facilitates using the applications of the cloud to obtain information. These cloud computing applications are designed for security and to guarantee that data is safely backed up. Word, Excel, PDF, Excel, and other file types can all be used to restore and convert data. Applications like Box, Mozy, Jouks, and Google Suite are excellent examples of cloud storage.

**4. Business applications**

Applications for businesses rely on cloud-based service providers. Today, every corporation needs a cloud-based business application to expand. Additionally, it makes sure users can access corporate applications around-the-clock.

* **Salesforce:** The Salesforce platform offers solutions for e-commerce, marketing, sales, and other functions. Additionally, a cloud development platform is offered.
* **MailChimp:**A platform for publishing emails called MailChimp offers a number of tools for creating, sending, and storing email templates.
* **Quickbooks:**The phrase “Run Enterprise anywhere, anytime on any device” is how Quickbooks operates. The firm, it offers online accounting solutions. More than 20 persons can collaborate on the very same system at once.

**5. Education based applications**

The use of cloud computing in education is growing rapidly. It provides students with a variety of online distance learning platforms and student information websites. Strong virtual classrooms, easy accessibility, safe data storage, scalability, better reach for the students, and minimum hardware requirements for the apps are all benefits of using the cloud in the field of education.

**6. Big Data Analysis**

The use of cloud computing for in-depth data analysis is one of its most significant uses. Big data cannot be stored using conventional data management solutions due to its enormous volume. Big data can now be stored and analysed by enterprises to produce priceless business insights thanks to the cloud’s limitless storage capacity.

**7. Testing**

Applications for cloud computing offer the simplest method for product testing and development. Such environment would’ve been time-consuming, costly to set up in terms of IT resources as well as infrastructure, and labor-intensive using conventional approaches. Organizations may employ scalable and adaptable cloud services for product creation, testing, and deployment thanks to cloud computing.

**8. Entertainment Application**

The entertainment sector employs a multi-cloud approach to engage the target audience. V and video conference and Online games are just a couple of the leisure applications available with cloud computing.

* **Online gaming**

Cloud gaming is one of the most significant forms of entertainment nowadays. It provides a variety of online games that are remote-controlled by the cloud. GeForce Now, Vortex, Project xCloud, Shaow, and PlayStation Now are the top cloud gaming providers.

* **Video Conference**

Apps for video conferencing offer a quick and easy way to connect. We can use cloud-based video conferencing to speak with our friends, family, and business colleagues. The advantages of adopting video conferencing include cost savings, increased productivity, and elimination of interoperability.

**9. E-commerce Applications**

Ecommerce customers and e-businesses can react swiftly to possibilities as they arise thanks to cloud-based ecommerce solutions. It offers business leaders a fresh method for getting things done quickly and efficiently. They manage product data, customer data, as well as other operational systems in cloud settings.

**10. Social Media Applications**

Many users can connect with one another via social networking services like Facebook, Twitter, LinkedIn, etc. thanks to social cloud applications.

**11. Antivirus Application**

There are also numerous antivirus programmes accessible for support. These cloud application services guarantee the system’s efficient operation. They provide numerous advantages to consumers, including system cleaning, malware detection, and virus removal. This antivirus is free of charge and is considered the best antivirus for your personal computer. By sending the information to the cloud data centre and repairing it, this application’s main purpose is to detect malware. Some of the most widely used cloud antivirus programmes include Kaspersky Endpoint Security and Sophos Endpoint Protection.

**12. URL conversion application**

There are a number of social media programmes, one of which is connected to Twitter and aids in the reduction of long URLs to shorter ones. The function of a programme like bitly is to shorten lengthy URLs, which then refers users back to the initial website. In addition to protecting the application from malware and hacking activity, it facilitates microblogging.

**13. Presentation Application**

For presentation services, there is software available that enables the importation of PowerPoint Presentations by making slides. One such programme that assists the user in making professional presentations is called “Sliderocket.” Anywhere in the planet can access these cloud computing applications. It offers both a free and a paid version of the programme.

**14. GPS application**

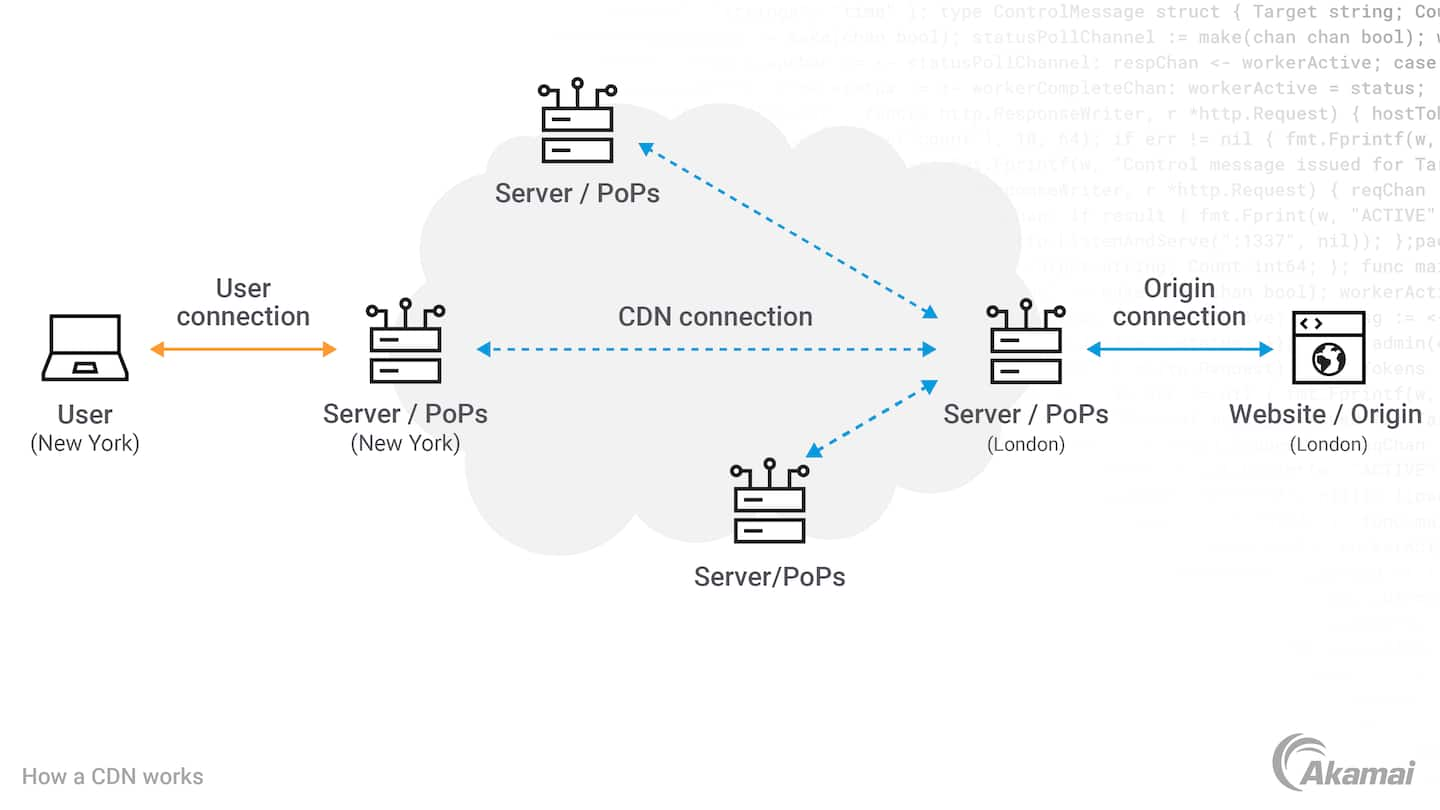
GPS, another innovation of cloud computing and its uses, is made available to users. These programmes assist users in discovering destinations online and in following directions from a map. Websites like Google Maps, Yahoo Maps, and others are examples of such cloud computing applications. Millions of users utilise these applications, which are also available for free.

**15. Accounting application**

One of the real-time cloud computing apps that supports management in the accounting area of the firm is accounting software. One such tool utilised by larger businesses to provide real-time daily accounting services is Outright. You may keep track of spending, earnings, and losses in real time. Other cloud accounting software examples are Kash Flow and Zoho Books.

**CONTENT DELIVERY SERVICES**

A content delivery network (CDN) is a group of interconnected servers distributed around the world that accelerate the delivery of web content by serving it from locations closer to users. By storing copies of files — a process called caching — CDNs minimize the time it takes for content like web pages, images, and video to reach a user’s device. This prevents users from having to wait for content to load as they watch movies, download software, make purchases, or perform other tasks online. A cloud CDN is a CDN that uses distributed cloud-based compute, storage, and networking resources. Many cloud services providers offer CDNs as part of their platforms.



**Components of a cloud CDN**

* **Points of presence** (PoPs) are data centers distributed around the world that serve content to nearby users. PoPs reduce the time required to serve content to a user’s device.
* **Servers** with large amounts of storage and RAM store and deliver cached files to users, accelerating the load times of websites and reducing bandwidth consumption.
* **Cloud load balancing** technology seamlessly directs traffic to the PoPs that can serve content to a user fastest, evenly distributing the load across cloud resources and achieving greater cloud optimization.
* **Solid-state drives (SSD), hard disk drives (HDD), and random-access memory (RAM)** within cloud CDN servers store files to be served, with the most requested files being stored on the fastest media.

**Benefits of a CDN**

* **High performance**. CDNs improve the performance of web content, high-quality video, audio streams, software downloads, and other online experiences.
* **Greater availability**. Superior cloud CDNs can absorb massive amounts of traffic from users while still serving content at an optimal rate.
* **Better security**. Distributed denial-of-service (DDoS) attacks and web-based exploits not only threaten organizations but the online experience for users as well. CDN systems have made DDoS protection and cloud-based security a core competency, providing cloud-based security solutions to protect content providers and users by mitigating a wide array of attacks that could compromise delivery and availability.
* **Lower cost**. Serving content from traditional clouds can incur significant data egress charges. CDNs reduce cost by reducing bandwidth consumption and the significant expense associated with purchasing bandwidth.

**Use cases of the cloud CDN**

* **Delivering content at high speeds**. Using a cloud CDN, content providers and media companies can deliver content to users much more quickly, providing high-quality, low-latency online experiences.
* **Improving the quality of streaming media**. Companies that deliver audio and video streaming rely on CDNs to reduce delivery time, minimize bandwidth costs, and scale up to meet user demand as needed. Streaming video services especially benefit from cloud CDN technology that enables them to meet spikes in demand when certain video content becomes quite popular.
* **Scaling up for multiple users**. From gaming companies to SaaS and business productivity platforms, the scalability of CDNs enable millions of users to access a website or platform simultaneously without experiencing degraded performance.

## Cloud Computing vs. cloud CDN?

Cloud computing provides on-demand access to computing, storage, and networking resources with a pay-as-you-go pricing model. Clouds rely on hundreds of data centers around the world with virtualized machines running on servers that allow multiple users to access computer resources as needed. A cloud CDN is a collection of servers that distribute content from an origin server throughout the world, caching content close to where each end user will access it. By caching content physically close to the users that request it, CDNs help to significantly reduce latency.

**Cloud CDN Vs Cloud Storage**

A cloud CDN is a network of servers that distributes content from an origin server throughout the world, storing or caching content close to where each user will access it. Cloud storage is a cloud computing service model where organizations can store content in remote cloud servers managed by cloud providers to reduce costs, minimize IT burden, enhance backup and disaster recovery efforts, simplify compliance, and streamline archiving and record retention.

**ANALYTICS SERVICES**

Cloud analytics is a process whereby business data is stored in the cloud for further analysis. With this process, the data is harnessed and used to gain valuable information to help grow or develop a business. It deals with business data analysis similarly to an on-premise solution. Algorithms of cloud analytics are deployed to assess and analyze large data sets. This is used to identify possible patterns and trends further. They can predict future events while producing a wide range of valuable information for an organization's decision-makers.

It is usually hosted as a service whereby businesses are meant to purchase as much as they need. This means that cloud-hosted analytics allows business users to pay as they use the solution. Another option is on-premises analytics.

**Cloud Computing: The basis of analytics**

Firstly, to understand the technique behind cloud analytics, it is essential to understand the basis of cloud computing. Cloud computing is a process whereby computer/digital services are delivered through the internet. Cloud is a word used to describe the many groups of computers that form the background of the internet infrastructure.

It is a solution that is hosted through the internet. Many cloud analytics solutions are provided by providers who have sophisticated data centers. These providers usually invest in first-rate solutions with the needed storage space and processing power for assessing large amounts of data.

It helps to collect and store data securely using cloud-hosted solutions. This way, the data can be accessed easily from any device by any authorized user. Also. It usually maintains the integrity of the data by performing some routine tasks. It does this by −

* Clean up the data
* Organize the data
* Process the data
* Analyze the data

All of these tasks are carried out by AI systems that may direct drivers and ensure their safety by the cloud analytics solution using proprietary algorithms. There are many vendor-specific algorithms that, as soon as the data analytics is completed, the solution usually uses algorithms to evaluate and present the data insights through clear visualizations and other presentable visual formats. When it comes to commercial solutions for it, users can expect to get varying features. But all cloud-hosted analytics solutions have common components. Some of these components that are common to solutions of it include −

* Data storage and sharing
* Analytic models
* Computing power
* Processing applications
* Data models
* Data sources

## Benefits of Cloud Analytics

Cloud analytics for businesses means a lot of benefits for subscribers. There are so many benefits of this solution that can significantly impact a business. They are −

**Data consolidation** − It combines data from different non-compatible sources to present them in a single format. Generated and featured in a single visual display format, it becomes immediately accessible to all users.

**Security** − Providers of cloud analytics solutions deploy robust data encryption technology for guaranteed protection whenever it is transmitted over systems. The essential security benefit is the offsite storage of data. A recent report showed that 34% of data breaches occurred due to insider threats. This includes current and past employees who share confidential data as they leave the organization.

**Cost Reduction**− It offers superior cost reduction since on-premise data analytics usually involves investing in different hardware systems. With on-premise solutions, there is a constant need for migrations and upgrades - which generally causes long periods of system downtime that affect business continuity. It does not require any investment in additional hardware. Also, you can leverage the in-house team's expertise from the service providers.

**Scalability**− It supports scalability as businesses can upgrade their subscription to include more features and support more in-house users. On-premise solutions mean you must buy and set up new system hardware to cater to any demand spike.

**Collaboration and sharing**− It can integrate all the data acquired from different parts of an organization. This leads to superior decision-making and accessible communication.

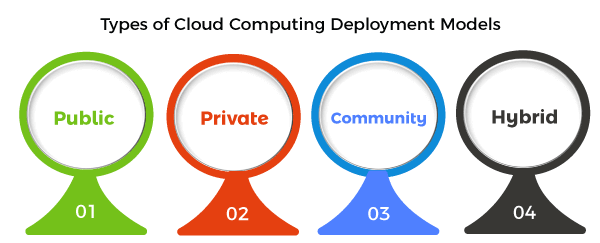
**Deployment And Management Service**

**What Is A Cloud Deployment Model?**

It works as your virtual computing environment with a choice of deployment model depending on how much data you want to store and who has access to the Infrastructure.

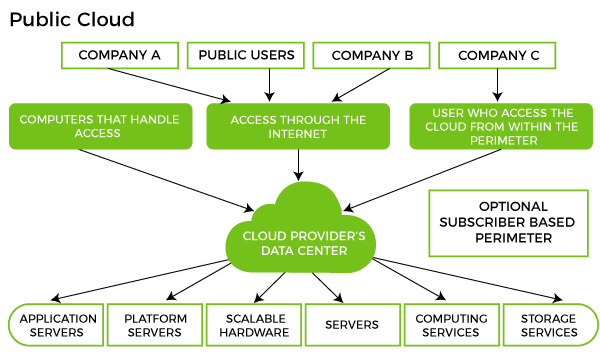
**Different Types Of Cloud Computing Deployment Models**

Most cloud hubs have tens of thousands of servers and storage devices to enable fast loading. It is often possible to choose a geographic area to put the data "closer" to users. Thus, deployment models for cloud computing are categorized based on their location. To know which model would best fit the requirements of your organization, let us first learn about the various types.



**Public Cloud**

The name says it all. It is accessible to the public. Public deployment models in the cloud are perfect for organizations with growing and fluctuating demands. It also makes a great choice for companies with low-security concerns. Thus, you pay a cloud service provider for networking services, compute virtualization & storage available on the public internet. It is also a great delivery model for the teams with development and testing. Its configuration and deployment are quick and easy, making it an ideal choice for test environments.



**Benefits of Public Cloud**

* Minimal Investment - As a pay-per-use service, there is no large upfront cost and is ideal for businesses who need quick access to resources
* No Hardware Setup - The cloud service providers fully fund the entire Infrastructure
* No Infrastructure Management - This does not require an in-house team to utilize the public cloud.

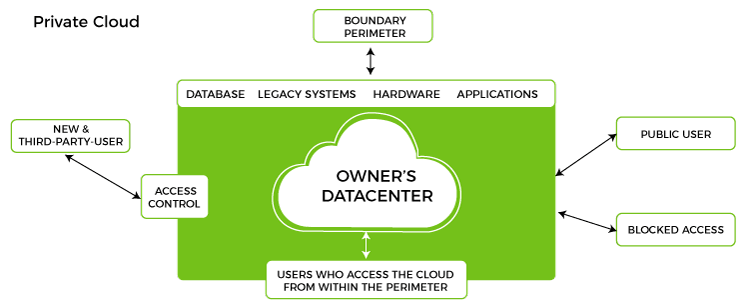
**Limitations of Public Cloud**

* Data Security and Privacy Concerns - Since it is accessible to all, it does not fully protect against cyber-attacks and could lead to vulnerabilities.
* Reliability Issues - Since the same server network is open to a wide range of users, it can lead to malfunction and outages
* Service/License Limitation - While there are many resources you can exchange with tenants, there is a usage cap

### Private Cloud

Now that you understand what the public cloud could offer you, of course, you are keen to know what a private cloud can do. Companies that look for cost efficiency and greater control over data & resources will find the private cloud a more suitable choice.

It means that it will be integrated with your data center and managed by your IT team. Alternatively, you can also choose to host it externally. The private cloud offers bigger opportunities that help meet specific organizations' requirements when it comes to customization. It's also a wise choice for mission-critical processes that may have frequently changing requirements.

. 

**Benefits of Private Cloud**

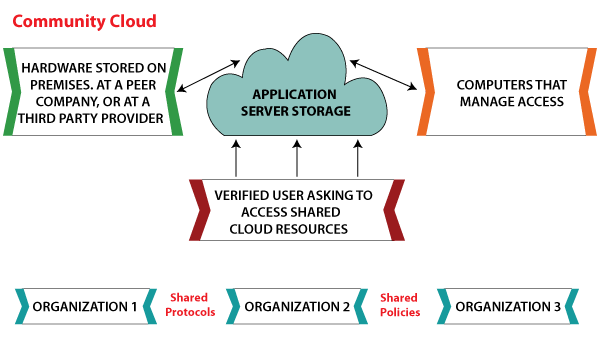
* Data Privacy - It is ideal for storing corporate data where only authorized personnel gets access
* Security - Segmentation of resources within the same Infrastructure can help with better access and higher levels of security.
* Supports Legacy Systems - This model supports legacy systems that cannot access the public cloud.

**Limitations of Private Cloud**

* Higher Cost - With the benefits you get, the investment will also be larger than the public cloud. Here, you will pay for software, hardware, and resources for staff and training.
* Fixed Scalability - The hardware you choose will accordingly help you scale in a certain direction
* High Maintenance - Since it is managed in-house, the maintenance costs also increase.

**Community Cloud**

The community cloud operates in a way that is similar to the public cloud. There's just one difference - it allows access to only a specific set of users who share common objectives and use cases. This type of deployment model of cloud computing is managed and hosted internally or by a third-party vendor. However, you can also choose a combination of all three.



**Benefits of Community Cloud**

* Smaller Investment - A community cloud is much cheaper than the private & public cloud and provides great performance
* Setup Benefits - The protocols and configuration of a community cloud must align with industry standards, allowing customers to work much more efficiently.

**Limitations of Community Cloud**

* Shared Resources - Due to restricted bandwidth and storage capacity, community resources often pose challenges.
* Not as Popular - Since this is a recently introduced model, it is not that popular or available across industries

**Hybrid Cloud**

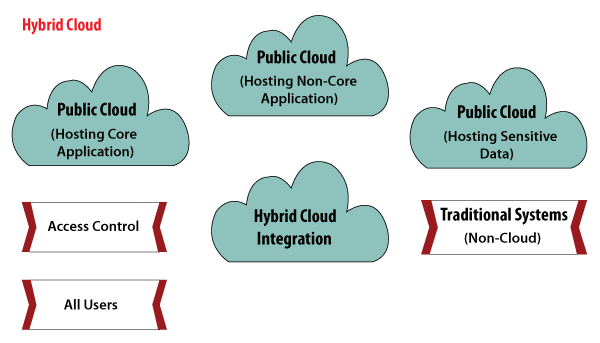
As the name suggests, a hybrid cloud is a combination of two or more cloud architectures. While each model in the hybrid cloud functions differently, it is all part of the same architecture. Further, as part of this deployment of the cloud computing model, the internal or external providers can offer resources.

**Benefits of Hybrid Cloud**

* Cost-Effectiveness - The overall cost of a hybrid solution decreases since it majorly uses the public cloud to store data.
* Security - Since data is properly segmented, the chances of data theft from attackers are significantly reduced.
* Flexibility - With higher levels of flexibility, businesses can create custom solutions that fit their exact requirements

**Limitations of Hybrid Cloud**

* Complexity - It is complex setting up a hybrid cloud since it needs to integrate two or more cloud architectures
* Specific Use Case - This model makes more sense for organizations that have multiple use cases or need to separate critical and sensitive data



**IDENTITY AND ACCESS MANAGEMENT SERVICES**

Identity Access Management is used by the root user (administrator) of the organization. The users represent one person within the organization, and the users can be grouped in that all the users will have the same privileges to the services.

## **The Role of IAM in Cloud Computing**

In cloud computing, IAM is critical for managing access to resources while maintaining operational efficiency and security. The framework enables organizations to verify identities and grant access based on permissions, ensuring that only authenticated users can access sensitive resources. As cloud usage grows, IAM becomes even more integral in protecting digital assets and establishing a resilient, secure infrastructure. With eMudhra’s advanced IAM solutions, organizations can strategically secure their cloud environments and enforce effective identity governance across multiple cloud platforms.

## **Key Components of IAM Architecture in Cloud Computing**

IAM in cloud computing comprises several components, each supporting different aspects of identity and access management. eMudhra’s IAM architecture integrates these components seamlessly, offering a unified approach to cloud security:

**1. Identity Management**

At the core of IAM is managing digital identities—whether for users, devices, or applications. Identity management encompasses creating, maintaining, and deleting identities to ensure secure access to resources. eMudhra’s identity management solution is built on strict access controls that facilitate secure identity lifecycle management, from onboarding to offboarding, thereby enforcing the principle of least privilege.

**2. Authentication**

Authentication verifies the identity of users and entities seeking access to the cloud. eMudhra’s IAM solutions incorporate advanced authentication mechanisms, including multi-factor authentication (MFA), providing an additional layer of security to prevent unauthorized access. By implementing robust authentication methods, organizations can control access efficiently and securely.

**3. Authorization**

Authorization determines what actions an authenticated entity can perform. eMudhra’s IAM solutions define permissions, roles, and policies that control access and ensure each user has only the access they need to perform their tasks. This least privilege approach minimizes security risks and ensures that sensitive resources are only accessible to authorized personnel.

**4. Access Policies**

Access policies in IAM define which resources an identity can access and the actions it can perform. eMudhra’s IAM architecture allows organizations to create granular policies, tailoring access controls to meet security requirements and operational needs. This capability ensures resources are safeguarded while enabling efficient access for legitimate users.

**5. Audit and Monitoring**

Effective IAM systems monitor and audit user activities, providing visibility into cloud operations. eMudhra’s IAM architecture offers comprehensive tracking and logging capabilities, which detect anomalies, investigate incidents, and support proactive risk mitigation. Auditing also helps organizations meet compliance requirements by keeping detailed records of access activities.

**6. Integration with Cloud Services**

eMudhra’s IAM solutions integrate with popular cloud platforms like AWS, Azure, and Google Cloud, leveraging native IAM features for cohesive security management. By aligning with cloud providers’ capabilities, eMudhra ensures organizations benefit from enhanced security and resource efficiency.

Challenges in Managing Access and Identities on the Cloud

IAM in cloud computing is vital but also presents unique challenges. eMudhra’s solutions address these challenges to create a streamlined and secure IAM framework:

* Security Risks: Unauthorized access is a persistent threat, making strong IAM protocols essential. eMudhra’s IAM mitigates risks through advanced authentication and access control.
* Complexity: Managing users, devices, and applications across cloud environments can be complex. eMudhra simplifies IAM with a centralized management interface.
* Scalability: As organizations grow, so does the need for robust access management. eMudhra’s IAM scales seamlessly with cloud usage, maintaining effective controls.
* Integration Issues: Integrating IAM with cloud infrastructures can be challenging. eMudhra’s solutions integrate smoothly with major cloud platforms, streamlining the transition.
* Identity Lifecycle Management: Adding and removing users securely can be complicated. eMudhra’s IAM automates lifecycle management to keep permissions updated and reduce errors.
* Compliance Challenges: Regulatory requirements add complexity. eMudhra’s IAM aligns with compliance standards, helping organizations meet industry regulations like GDPR and HIPAA.
* Data Safety: Securing sensitive cloud data is paramount. eMudhra ensures data protection through secure access controls and regular auditing.

**Why IAM is Critical in Cloud Computing**

IAM is indispensable in cloud computing as it safeguards digital environments by controlling access management. By managing user identities and enforcing authentication protocols, IAM minimizes unauthorized access risks and protects data integrity. eMudhra’s IAM solutions support the user lifecycle, simplify compliance, and offer centralized management to enable a secure, scalable cloud infrastructure.

**Advantages of IAM Architecture in Cloud Computing**

**1. Improved Security**

eMudhra’s IAM architecture intensifies security by enforcing strict authentication and authorization, limiting unauthorized access to sensitive resources and preventing data breaches.

**2. Compliance Support**

With industry-aligned access controls, eMudhra’s IAM helps companies adhere to regulations such as GDPR and HIPAA, protecting organizations from potential legal issues.

**3. Efficient Identity Lifecycle Management**

eMudhra streamlines identity lifecycle management by automating the onboarding, modification, and offboarding processes, ensuring access permissions are always up to date and reducing administrative burdens.

**4. Centralized Control**

Our IAM solutions provide a centralized platform for managing permissions across cloud services, enabling consistent policy enforcement, activity tracking, and rapid response to security incidents.

**5. Operational Efficiency**

eMudhra’s IAM automates identity management tasks like provisioning and de-provisioning, improving IT workflows, optimizing resource utilization, and freeing teams for strategic projects.

**6. Scalability**

Our IAM architecture adapts effortlessly to growing user bases, maintaining effective access controls for expanding organizations without compromising performance or security.

**7. Cost Efficiency**

By automating identity management and reducing manual processes, eMudhra’s IAM solutions save time and operational costs, leading to resource optimization.

**OPEN SOURCE PRIVATE CLOUD SOFTWARE**

Open source clouds are cloud computing platforms constructed utilizing open source software and technologies. They are not owned or operated by a single vendor. The beauty of open source clouds is that the code is developed collaboratively by a community of developers and users.

The best example of open source Cloud Computing is OpenStack. This is one of the most frequently asked cloud computing interview questions. Cloud computing lets us store and access our applications or data over remote computers instead of your own computer.

Open source databases in the cloud

* Amazon relational database.
* Microsoft Azure SQL database.
* Oracle database.
* IBM Db2 on cloud.
* Google Cloud SQL.

**Open-Source Cloud Management Solutions**

The best open-source cloud management software platforms make it easier to manage cloud services and reduce overhead costs. This is mainly as cloud software has become an essential component in any business IT environment. It can manage and run applications through a SaaS model or set up your cloud architecture with an IaaS or PaaS platform. Managing many different services and providers can be difficult without the right cloud management software or services. Therefore, many cloud management solutions have become available. You can either manage cloud architecture directly or else bring in different cloud services under a single third party for easier management.

A cloud management platform is ideal if it has:

* Strong integration with IT infrastructure
* Automating manual tasks
* Service management
* Governance and security

**List of Open-Source Cloud Platforms and Tools**

**1. OpenStack**

Open Stack is used for cloud computing. It is used as an infrastructure as a service (Iaas). The software is meant to create private and public clouds. Users will be able to control compute, storage and networking resources throughout a data centre. This is operated from a dashboard or through OpenStack’s API. Backed by the most influential companies in software development and hosting, with thousands of individual community members, many think that OpenStack is the future of cloud computing.

**Key Features**

* Services: Messaging, containers, clustering, compute, identity, metadata indexing as service, events, workflows, DNS, bare-metal provisioning, governance, benchmarking, optimization and deployment
* Web front-end, Big Data processing framework, container orchestration engine and NFV orchestration.

**2. Eucalyptus**

Eucalyptus is open-source software for creating AWS-compatible private and hybrid clouds. It is also a Linux-based software architecture that performs scalable private and hybrid clouds within your IT infrastructure. It lets you use your collections of resources using a self-service interface whenever needed. As an infrastructure as a Service (IaaS), Eucalyptus also enables your users to provision your compute and storage resources on demand.

**Key Features**

* Works with various hypervisors.
* Communication within internal processes is guarded through SOAP and WS-Security.
* Administrative features like user and group management and reports.

**3. OpenNebula**

Open Nebula is a flexible turnkey open-source solution to create Private Clouds and maintain Data Center virtualisation. It implements IaaS. The first open-source version was released in March 2008. It is intended to be a simple but feature-rich customisable solution to manage enterprise clouds. It is also simple to install, update and operate by the administrators; and easy to use by end-users.

**Key Features**

* Supports APIs like AWS EC2 and OGF OCCI.
* It supports SSH and X.509 for security and even supports token login functionality.
* Powerful UNIX based CLI for administration.

**4. Apache CloudStack**

It is known to be a top-level project of the Apache Software Foundation. Apache CloudStack maintains networks of virtual machines as an Infrastructure as a Service (IaaS). CloudStack is a multi-hypervisor, high-availability cloud management platform. It is software that provides a cloud Orchestration layer, providing automation of the creation, provisioning and configuration of IaaS components.

**Key Features**

* Self-service user interface: AJAX console access, network virtualisation, multi-role support usage metering, virtual routers.
* LVM support: Block storage volumes, LDAP integration, OpenStack Swift integration, and domains and delegated administration.

**5. VirtEngine**

Virt Engine can be utilized to build private or public clouds supporting IaaS, PaaS and SaaS. It allows customers to make use of applications in a few clicks. VirtEngine has a lot of applications, and a simple user interface for you to self-serve your needs. VirtEngine by DET.io is open as two separate solutions for the public and private cloud. The Public Cloud lets users build their own cloud and present servers to customers. The Public Cloud is open as a minified edition and a complete solution. The private cloud is an open-source and free solution for enterprises supporting HA and more enterprise features.

**Key Features**

* Access control, cost management, multi-cloud management, one-click apps, demand and supply monitoring and automatic launch.
* DNS support, cloud-native, multi-locations, Docker containers, cloud virtual machines, self-healing and migration tools.

**19IINFCX1: CLOUD COMPUTING**

**Unit-3**

**Cloud Application Design and Development:** Design consideration- Reference Architecture for Cloud Application - Cloud Application Design Methodologies - Data Storage Approaches- Development in Python: Design Approaches – Application: Image Processing – Document Storage - Map Reduce - Social Media Analytics.

**CLOUD APPLICATION DESIGN AND DEVELOPMENT**

**Cloud Application**

An online application that saves data, permits remote access and administration, and runs at least some of its code on a distant server is an example of a "cloud-based application."

**Types of Cloud Applications in Development**

**Cloud-Based Web Applications**

Web apps allow users to access computer programs from any computer or mobile device with an internet link because they are saved and run in the cloud. Users might have to sign up for an account and check in on their devices before accessing the application. Examples include social networking sites like Facebook and LinkedIn and online storage and sharing services for documents like Dropbox Paper and Google Docs.

Customer Relationship Management(CRM)

This application makes arranging clients, prospects, and other relationships possible. Email marketing, lead management tools, and contact lists are the most useful features. The information you get from these will help you enhance customer insights and business management.

**Enterprise Resource Planning(ERP)**

Enterprise resource planning, or ERP, is the software businesses use to monitor everything from customer support to financial reporting and inventory administration. Such software frequently includes tools for managing customers, goods, workers, and money.

**Property Management Software(PMS)**

PMS, or property management system, is the acronym. It's a piece of software made to save property managers' time by handling repetitive tasks. This involves requesting replacement estimates and gathering funding. A database can also create financial measures and keep track of possessions.

**Solutions for Cloud-based Application Development**

The cloud has made classic on-premises programming tools more adaptable, scalable, and affordable. Because of this, more and more people are turning to cloud-based platforms to create their applications. Some of the most common kinds of cloud-based application development solutions are as follows −

**Platform as a Service (PaaS)**

These tools and resources allow for the comprehensive development, distribution, and administration of cloud-based applications. To streamline the app-creation process, many companies now give coders access to a "platform as a service."

**Infrastructure as a Service (IaaS)**

Developers now have a consolidated platform from which to create, distribute, and manage their applications thanks to Infrastructure as a Service. Services like networking, storage, and virtual computers are just some of what IaaS providers can provide for their clients.

**Software as a Service (SaaS)**

Users may acquire cloud-based programs using this kind of service from a distance. Most SaaS providers provide access to a library of programs that can be accessed from any location with an internet link. CRM (client relationship management) and project management are two examples of these programs' possible uses.

**Mobile Backend as a Service (MBaaS)**

Using these choices, authors enter into a comprehensive framework designed with mobile applications in mind. An MBaaS should assist its customers with user identification, data storage, SMS notifications, and data analysis.

**Primary Benefits of Cloud Application Development**

**Flexibility**

Apps hosted in the cloud are much more flexible in terms of scalability. If your business grows and needs more, you can easily add more memory or disk room to your computer. Turning off your online applications when you're not using them is a fantastic way to cut costs. Because of their adaptability, cloud-based applications make changing suppliers less hassle.

**Scalability**

Cloud-based applications have greater scalability than legacy IT systems regarding user population and data volume, and they are kept from being slowed down or made less secure by this. If a server in the cloud crashes, another one will be standing by to assume its position in next to no time. This guarantees that your system will continue to function normally regardless of the number of users or the state of any individual component.

**Efficiency and productivity**

Developing application in the cloud allows you to put your attention where it matters to your customers. Cloud application creation allows for easy scalability, regardless of your IT infrastructure or other technical hurdles. You won't have to worry about finding a solution on your own. Adjustments are easy but will take some time to complete. As a result, less investment will be made in hardware, software, work, etc.

**CLOUD APPLICATION DESIGN CONSIDERATIONS**

When designing applications for the cloud, irrespective of the chosen platform, we have often found it useful to consider four specific topics during my initial discussions: scalability, availability, manageability and feasibility.

**(i). Scalability**

Conversations about scalability should focus on any requirement to add additional capacity to the application and related services to handle increases in load and demand. It is particularly important to consider each application tier when designing for scalability, how they should scale individually and how we can avoid contention issues and bottlenecks. Key areas to consider include:

**Capacity**

* Will we need to scale individual application layers and, if so, how can we achieve this without affecting availability?
* How quickly will we need to scale individual services?
* How do we add additional capacity to the application or any part of it?
* Will the application need to run at scale 24x7, or can we scale-down outside business hours or at weekends for example?

**Platform / Data**

* Can we work within the constraints of our chosen persistence services while working at scale (database size, transaction throughput, etc.)?
* How can we partition our data to aid scalability within persistence platform constraints (e.g. maximum database sizes, concurrent request limits, etc.)?
* How can we ensure we are making efficient and effective use of platform resources? As a rule of thumb, I generally tend towards a design based on many small instances, rather than fewer large ones.
* Can we collapse tiers to minimise internal network traffic and use of resources, whilst maintaining efficient scalability and future code maintainability?

**Load**

* How can we improve the design to avoid contention issues and bottlenecks? For example, can we use queues or a service bus between services in a co-operating producer, competing consumer pattern?
* Which operations could be handled asynchronously to help balance load at peak times?
* How could we use the platform features for rate-leveling (e.g. Azure Queues, Service Bus, etc.)?
* How could we use the platform features for load-balancing (e.g. Azure Traffic Manager, Load Balancer, etc.)?

**(ii). Availability**

Availability describes the ability of the solution to operate in a manner useful to the consumer in spite of transient and enduring faults in the application and underlying operating system, network and hardware dependencies. In reality, there is often some crossover between items useful for availability and scalability. Conversations should cover at least the following items:

**Uptime Guarantees**

* What Service Level Agreements (SLAs) are the products required to meet?
* Can these SLAs be met? Do the different cloud services we are planning to use all conform to the levels required? Remember that SLAs are composite.

**Replication and failover**

* Which parts of the application are most at risk from failure?
* In which parts of the system would a failure have the most impact?
* Which parts of the application could benefit from redundancy and failover options?
* Will data replication services be required?
* Are we restricted to specific geopolitical areas? If so, are all the services we are planning to use available in those areas?
* How do we prevent corrupt data from being replicated?
* Will recovery from a failure put excess pressure on the system? Do we need to implement retry policies and/or a circuit-breaker?

**Disaster recovery**

* In the event of a catastrophic failure, how do we rebuild the system?
* How much data, if any, is it acceptable to lose in a disaster recovery scenario?
* How are we handling backups? Do we have a need for backups in addition to data-replication?
* How do we handle “in-flight” messages and queues in the event of a failure?
* Are we idempotent? Can we replay messages?
* Where are we storing our VM images? Do we have a backup?

**Performance**

* What are the acceptable levels of performance? How can we measure that? What happens if we drop below this level?
* Can we make any parts of the system asynchronous as an aid to performance?
* Which parts of the system are the mostly highly contended, and therefore more likely to cause performance issues?
* Are we likely to hit traffic spikes which may cause performance issues? Can we auto-scale or use queue-centric design to cover for this?

**Security**

This is clearly a huge topic in itself, but a few interesting items to explore which relate directly to cloud-computing include:

* What is the local law and jurisdiction where data is held? Remember to include the countries where failover and metrics data are held too.
* Is there a requirement for federated security (e.g. ADFS with Azure Active Directory)?
* Is this to be a hybrid-cloud application? How are we securing the link between our corporate and cloud networks?
* How do we control access to the administration portal of the cloud provider?
* How do we restrict access to databases, etc. from other services (e.g. IP Address white-lists, etc.)?
* How do we handle regular password changes?
* How does service-decoupling and multi-tenancy affect security?
* How we will deal with operating system and vendor security patches and updates?

**(iii). Manageability**

This topic of conversation covers our ability to understand the health and performance of the live system and manage site operations. Some useful cloud specific considerations include:

**Monitoring**

* How are we planning to monitor the application?
* Are we going to use off-the-shelf monitoring services or write our own?
* Where will the monitoring/metrics data be physically stored? Is this in line with data protection policies?
* How much data will our plans for monitoring produce?
* How will we access metrics data and logs? Do we have a plan to make this data useable as volumes increase?
* Is there a requirement for auditing as well as logging?
* Can we afford to lose some metrics/logging/audit data (i.e. can we use an asynchronous design to “fire and forget” to help aid performance)?
* Will we need to alter the level of monitoring at runtime?
* Do we need automated exception reporting?

**Deployment**

* How do we automate the deployment?
* How do we patch and/or redeploy without disrupting the live system? Can we still meet the SLAs?
* How do we check that a deployment was successful?
* How do we roll-back an unsuccessful deployment?
* How many environments will we need (e.g. development, test, staging, production) and how will deploy to each of them?
* Will each environment need separate data storage?
* Will each environment need to be available 24x7?

**(iv). Feasibility**

When discussing feasibility we consider the ability to deliver and maintain the system, within budgetary and time constraints. Items worth investigating include:

* Can the SLAs ever be met (i.e. is there a cloud service provider that can give the uptime guarantees that we need to provide to our customer)?
* Do we have the necessary skills and experience in-house to design and build cloud applications?
* Can we build the application to the design we have within budgetary constraints and a timeframe that makes sense to the business?
* How much will we need to spend on operational costs (cloud providers often have very complex pricing structures)?
* What can we sensibly reduce (scope, SLAs, resilience)?
* What trade-offs are we willing to accept?

**REFERENCE ARCHITECTURE FOR CLOUD APPLICATION**

Cloud computing technology is used by both small and large organizations to **store the information** in cloud and **access** it from anywhere at anytime using the internet connection.

Cloud computing architecture is a combination of **service-oriented architecture** and **event-driven architecture**.

Cloud computing architecture is divided into the following two parts –

* **Front End**
* **Back End**

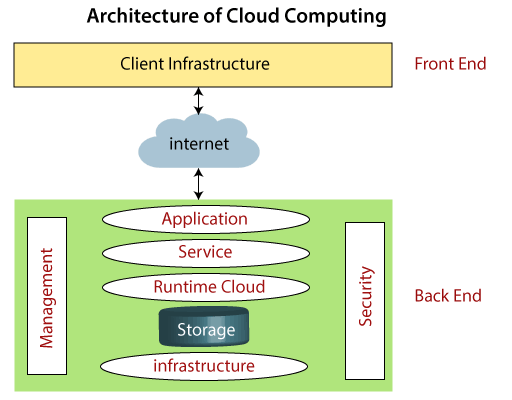
The below diagram shows the architecture of cloud computing –

**Front End**

The front end is used by the client. It contains client-side interfaces and applications that are required to access the cloud computing platforms. The front end includes web servers (including Chrome, Firefox, internet explorer, etc.), thin & fat clients, tablets, and mobile devices.

**Back End**

The back end is used by the service provider. It manages all the resources that are required to provide cloud computing services. It includes a huge amount of data storage, security mechanism, virtual machines, deploying models, servers, traffic control mechanisms, etc.



Components of Cloud Computing Architecture

There are the following components of cloud computing architecture -

**1. Client Infrastructure**

Client Infrastructure is a Front end component. It provides GUI (Graphical User Interface)  to interact with the cloud.

**2. Application**

The application may be any software or platform that a client wants to access.

**3. Service**

A Cloud Services manages that which type of service you access according to the client’s requirement.

Cloud computing offers the following three type of services:

**(i). Software as a Service (SaaS) –**It is also known as **cloud application services**. Mostly, SaaS applications run directly through the web browser means we do not require to download and install these applications. Some important example of SaaS is given below –

**Example:** Google Apps, Salesforce Dropbox, Slack, Hubspot, Cisco WebEx.

**(ii). Platform as a Service (PaaS) –** It is also known as **cloud platform** **services**. It is quite similar to SaaS, but the difference is that PaaS provides a platform for software creation, but using SaaS, we can access software over the internet without the need of any platform.

**Example:** Windows Azure, Force.com, Magento Commerce Cloud, OpenShift.

**(iii). Infrastructure as a Service (IaaS) –**It is also known as **cloud infrastructure services**. It is responsible for managing applications data, middleware, and runtime environments.

**Example:** Amazon Web Services (AWS) EC2, Google Compute Engine (GCE), Cisco Metapod.

**4. Runtime Cloud**

Runtime Cloud provides the **execution and runtime environment** to the virtual machines.

**5. Storage**

Storage is one of the most important components of cloud computing. It provides a huge amount of storage capacity in the cloud to store and manage data.

**6. Infrastructure**

It provides services on the **host level**, **application level**, and **network level**. 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.

**7. Management**

Management is used to manage components such as application, service, runtime cloud, storage, infrastructure, and other security issues in the backend and establish coordination between them.

**8. Security**

Security is an in-built back end component of cloud computing. It implements a security mechanism in the back end.

**9. Internet**

The Internet is medium through which front end and back end can interact and communicate with each other.

**CLOUD APPLICATION DESIGN METHODOLOGIES**

Some of the key features or applications of this design methodology are as follows:

* Use declarative formats for setup automation, to minimize time and cost for new developers joining the project
* Have a clean contract with the underlying operating system, offering maximum portability between execution environments
* Are suitable for deployment on modern cloud platforms (Google cloud, Heroku, AWS etc..), obviating the need for servers and systems administration
* Minimize divergence between development and production, enabling continuous deployment for maximum agility and can scale up without significant changes to tooling, architecture, or development practices

Design methodology is nothing but a collection of 12 factors which act as building blocks for deploying or developing an app in the cloud. Listed below are the 12 Factors:

**1. Codebase**: A 12 Factor App is always tracked in a version control system such as Git or Apache Subversion (SVN) in the form of code repository. This will essentially help you to build your code on top of one codebase, fully backed up with many deployments and revision control. As there is a one to one relationship between a 12 factor app and the codebase repository, so in case of multiple repositories, there is always a need to consider this relationship as a distributed system consisting of multiple 12 factored apps. Deployments should be automatic, so everything can run in different environments.

**2. Dependencies:** As the app is standalone and needs to install dependencies, it is important to explicitly declare and isolate dependencies. Moreover, it is always recommended to keep your development, production and QA identical to 12 Factor apps. This will help you to build applications in order to scale web and other such applications that do not have any room for error. As a solution to this, you can use a dependency isolation tool such as ex VirtualEnv for Python uniformly to remove several explicit dependency specifications to both production and development phases and environments.

**3. Config:**This factor manages the configuration information for the app. Here you store your configuration files in the environment. This factor focuses on how you store your data – the database Uniform Resource Identifier (URI) will be different in development, QA and production.

**4. Backing Services**: This includes backing service management services (local database service or any third party service) which depends on over a network connection. In case of a 12 factor app, the interface to connect these services should be defined in a standard way. You need to treat backing services like attached resources because you may want different databases depending on which team you are working with. Sometimes developers will want a lot of logs, while QA will want less. With this method, even each developer can have their own config file.

**5. Build, Run, Release**: It is important to run separately all the build and run stages making sure everything has the right libraries. For this, you can make use of required automation and tools to generate build and release packages with proper tags. This is further backed up by running the app in the execution environment while using proper release management tools like Capistrano for ensuring timely rollback.

**6. Stateless Processes**: This factor is about making sure the app is executed in the execution environment as one or more processes. In other words, you want to make sure that all your data is stored in a backing store, which gives you the right to scale out anything and do what you need to do. During stateless processes, you do not want to have a state that you need to pass along as you scale up and out.

**7. Port Binding**: Twelve factor apps are self-contained and do not rely on runtime injection of a web server into the execution environment to create a web-facing service. With the help of port binding, you can directly access your app via a port to know if it’s your app or any other point in the stack that is not working properly.

**8. Concurrency**: This factor looks into the best practices for scaling the app. These practices are used to manage each process in the app independently i.e. start/stop, clone to different machines etc. The factor also deals with breaking your app into much smaller pieces and then look for services out there that you either have to write or can consume.

**9. Disposability**: Your app might have different multiple processes handling different tasks. So, the ninth factor looks into the robustness of the app with fast startup and shutdown methods. Disposability is about making sure your app can startup and takes down fast and can handle any crash anytime. You can use some high quality robust queuing backend (Beanstalk, RabbitMQ etc.) that would help return unfinished jobs back to the queue in the case of a failure.

**10. Dev/Prod Parity**: Development, staging and production should be as similar as possible. In case of continuous deployment, you need to have continuous integration based on matching environments to limit deviation and errors. Some of the features of keeping the gap between development and production small are as follows:

1. **Make the time gap small**: a developer may write code and have it deployed hours or even just minutes later.
2. **Make the personnel gap small**: developers who wrote code are closely involved in deploying it and watching its behavior in production.
3. **Make the tools gap small**: keep development and production as similar as possible.

**11. Logs**: Logging mechanisms are critical for debugging. Having proper logging mechanisms allows you to output the log info as a continuous stream rather than managing the entire database of log files. Then, depending on the configuration, you can decide where that log will publish.

**12. Admin Processes:**One-off admin processes help in collecting data from the running application. In order to avoid any synchronization issues, you need to ensure that all these processes are a part of all deploys.

**DATA STORAGE APPROACHES**

Cloud Storage is a service that allows to save data on offsite storage system managed by third-party and is made accessible by a web services API.

**Storage Devices**

Storage devices can be broadly classified into two categories:

* Block Storage Devices
* File Storage Devices

**Block Storage Devices**

The block storage devices offer raw storage to the clients. These raw storage are partitioned to create volumes.

**File Storage Devices**

The file Storage Devices offer storage to clients in the form of files, maintaining its own file system. This storage is in the form of Network Attached Storage (NAS).

**Cloud Storage Classes**

Cloud storage can be broadly classified into two categories:

* Unmanaged Cloud Storage
* Managed Cloud Storage

**Unmanaged Cloud Storage**

Unmanaged cloud storage means the storage is preconfigured for the customer. The customer can neither format, nor install his own file system or change drive properties.

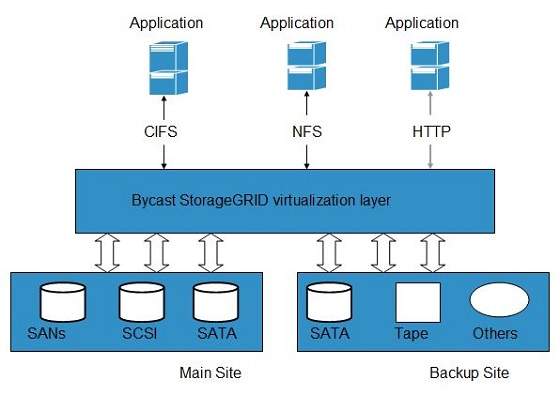
**Managed Cloud Storage**

Managed cloud storage offers online storage space on-demand. The managed cloud storage system appears to the user to be a raw disk that the user can partition and format.

**Creating Cloud Storage System**

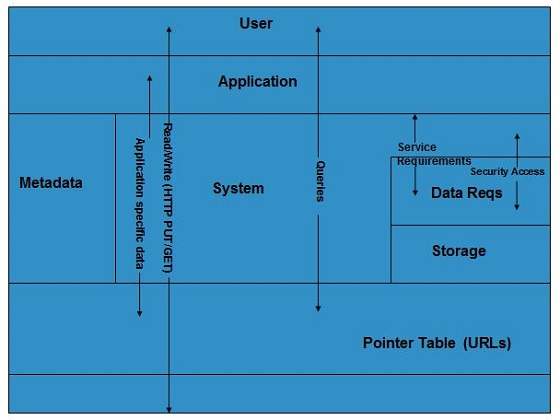
The cloud storage system stores multiple copies of data on multiple servers, at multiple locations. If one system fails, then it is required only to change the pointer to the location, where the object is stored.

To aggregate the storage assets into cloud storage systems, the cloud provider can use storage virtualization software known as **StorageGRID.** It creates a virtualization layer that fetches storage from different storage devices into a single management system. It can also manage data from **CIFS** and **NFS** file systems over the Internet. The following diagram shows how StorageGRID virtualizes the storage into storage clouds:



Virtual Storage Containers

The **virtual storage containers** offer high performance cloud storage systems. **Logical Unit Number (LUN)** of device, files and other objects are created in virtual storage containers. Following diagram shows a virtual storage container, defining a cloud storage domain:



**Challenges**

Storing the data in cloud is not that simple task. Apart from its flexibility and convenience, it also has several challenges faced by the customers. The customers must be able to:

* Get provision for additional storage on-demand.
* Know and restrict the physical location of the stored data.
* Verify how data was erased.
* Have access to a documented process for disposing of data storage hardware.
* Have administrator access control over data.

**DEVELOPMENT IN PYTHON: DESIGN APPROACHES**

1

UNIT-III

Cloud Application Development in Python: Design Approaches

Introduction: -

1

UNIT-III

Cloud Application Development in Python: Design Approaches

Introduction: -

1

UNIT-III

Cloud Application Development in Python: Design Approaches

Introduction: -

Cloud computing offers an infrastructure that lets developers run and manage applications without the need for in-house servers. This approach significantly reduces overhead, allowing developers to focus on innovation and agility. Python’s compatibility with cloud services and the flexibility it offers make it a preferred language for cloud-based applications. By using Python, developers can work with frameworks, libraries, and tools tailored to handle the unique challenges of the cloud environment, including data storage, security, and performance.

**Key Components of Cloud-based Python Development**

Building Python applications in the cloud requires a mix of knowledge in cloud infrastructure and Python-specific tools. Here are the primary components:

* **Cloud Infrastructure**: A strong foundation in cloud infrastructure (servers, databases, and networking) is essential. The main cloud providers—AWS, Google Cloud, and Azure—each offer unique services and architecture that impact the way Python applications are developed and run.
* **Frameworks and Libraries**: Python’s compatibility with a wide array of frameworks, such as Django and Flask, enables the creation of cloud-native applications. Additionally, libraries for handling database operations (like SQLAlchemy) and data processing (such as Pandas) are invaluable.
* **Containers and Microservices**: Containers like Docker and orchestration tools like Kubernetes are commonly used to deploy Python applications in a modular and scalable way. Microservices architecture, where applications are split into smaller, manageable services, allows for easier maintenance.

**Choosing the Right Cloud Platform for Python Applications**

Selecting the right cloud provider is a critical step. Each platform offers unique features, so it’s important to match the requirements of your Python application with the strengths of the cloud provider.

* **Amazon Web Services (AWS)**: AWS is a leading choice for developers needing a range of services, from compute power to machine learning tools. With services like EC2, Lambda, and S3, AWS offers flexibility and scalability, especially for Python applications that demand high performance.
* **Google Cloud Platform (GCP)**: Google Cloud provides strong data analysis tools and machine learning support, making it a good option for Python applications that work with large datasets or need real-time processing.
* **Microsoft Azure**: Known for its excellent integration with enterprise systems, Azure is ideal for businesses already using Microsoft products. Azure offers solid support for Python applications and has an extensive selection of services.

**Building Scalable Python Applications in the Cloud**

Scalability is crucial in cloud environments, where applications must handle varying loads and performance needs. Here are some essential practices for creating scalable cloud-based Python applications:

* **Designing with Microservices**: Adopting a microservices architecture allows applications to scale by dividing functionalities into smaller, independent services. Each service can scale independently, providing resilience and flexibility.
* **Using Load Balancing**: Implementing load balancers within the cloud environment distributes incoming traffic, maintaining performance levels. This is especially helpful when scaling up Python applications that may experience high traffic surges.
* **Auto-scaling**: Most cloud platforms offer auto-scaling services that add or remove resources based on demand. This feature is particularly beneficial for applications that experience variable traffic or seasonality, helping to manage costs and performance efficiently.

**Data Management in Cloud-based Python Applications**

Data storage and management are core aspects of cloud applications, as cloud platforms provide a range of data handling solutions tailored for different needs.

* **Cloud Databases**: Major cloud providers offer both SQL and NoSQL databases, such as AWS DynamoDB, Google Cloud Firestore, and Azure Cosmos DB. Python’s libraries like SQLAlchemy support easy interaction with these databases.
* **Data Processing and Storage**: Handling large datasets in Python applications often requires tools that can manage structured and unstructured data. Data lakes, like AWS Lake Formation or Google BigQuery, support large-scale data processing, ideal for applications with significant data needs.
* **Security and Compliance**: Ensuring data security is critical in cloud-based applications. Cloud providers offer encryption, role-based access, and data governance tools to protect sensitive data.

**Setting Up Continuous Integration/Continuous Deployment (CI/CD)**

Automation is key in cloud development, and CI/CD pipelines are essential for efficient, consistent deployment.

* **Automated Testing and Deployment**: CI/CD pipelines help maintain code quality by automating tests and deployments. Tools like GitHub Actions, Jenkins, and CircleCI support Python applications and enable a seamless deployment process.
* **Code Quality and Version Control**: Integrating code reviews and version control with tools like Git and GitLab ensures that teams can track changes and revert when necessary. CI/CD setups help deliver reliable applications, with the option to roll back changes if issues arise.

**Enhancing Security for Cloud-based Python Applications**

Security is a top priority in cloud environments, where applications face risks ranging from data breaches to DDoS attacks.

* **Encryption and Data Privacy**: Encrypting data both in transit and at rest is essential to prevent unauthorized access. Most cloud providers offer built-in encryption features to protect data integrity.
* **Authentication and Authorization**: Using secure identity management tools, such as AWS IAM or Azure Active Directory, helps regulate access to sensitive data and application functions, protecting against unauthorized access.
* **Network Security**: Setting up firewalls, VPNs, and network monitoring services on cloud platforms offers another layer of protection. Regular audits and vulnerability assessments further strengthen security.

**Cost Management for Cloud-based Python Applications**

Effective cloud cost management is essential for budget-friendly development.

* **Understand Pricing Structures**: Each cloud provider offers a different pricing model. Familiarity with compute time, data storage costs, and API call charges allows developers to estimate and control expenses effectively.
* **Use Auto-scaling Features**: Implementing auto-scaling not only supports performance but also helps control costs by adding resources only when needed. Setting thresholds for scaling prevents unnecessary charges.
* **Monitor and Optimize Resources**: Regularly reviewing cloud resources can help identify and eliminate underutilized services. Many providers offer budgeting tools and alerts to keep costs in check.

**Future Trends in Cloud-based Python Development**

The cloud continues to evolve, opening up new possibilities for Python developers:

* **Edge Computing**: Edge computing, which moves processing closer to data sources, is becoming increasingly popular for applications that require low-latency responses. Python’s lightweight nature makes it a good fit for edge devices.
* **Quantum Computing**: Some cloud providers are exploring quantum computing, a technology that promises breakthroughs in processing power. Python’s versatility means it could play a role in this emerging area as tools and libraries are developed.
* **Hybrid and Multi-cloud Solutions**: Many businesses are opting for hybrid solutions, combining public and private clouds to achieve flexibility. Python applications can easily integrate across multiple platforms, providing a cohesive experience.

**CLOUD APPLICATIONS**

**1. IMAGE PROCESSING**

Cloud computing has many applications in image processing, including:

* **Image analysis**

Cloud computing can analyze images using APIs like Google Cloud Vision:

* + **Event-driven processing**: Users can call a REST API with an image reference and optional annotation features. The API triggers a Cloud Function, which calls Cloud Vision to analyze the image.
  + **Batch uploads**: Users can upload images to Cloud Storage, which triggers a Cloud Function to analyze the images.
  + **Content moderation**: Cloud Run can use ImageMagick to blur images that contain adult or violent content.
* **Cloud RAW image processing**

Users can send RAW images from their camera to image.canon for processing in the cloud. This can reduce noise and false color, and is effective for images like night scenes, astronomical images, and images of buildings with tile or brick patterns.

* **Athlete training**

Cloud computing can be used to calculate energy consumption and optimize task unloading and wireless resource allocation.

* **Surveillance systems**

Digital signal processing (DSP) can be used to analyze video streams in real-time to identify suspicious activities, monitor crowded areas, and enhance public safety.

Other applications of cloud computing in image processing include:

* Using Cloud Functions and Google Cloud Vision to perform AI/ML high speed image processing
* Using the Cloud Storage trigger to process images asynchronously or by batch upload

**2. DOCUMENT STORAGE**

The key features of cloud computing are as follows.

* It has a greater availability of resources.
* Easy maintenance is one of the key benefits of using Cloud computing.
* Cloud computing has a Large Network Access.
* It has an automatic system.
* Security is one of the major components and using cloud computing you can secure all over the networks.

**Storage Systems in the Cloud**

There are 3 types of storage systems in the Cloud as follows.

* Block-Based Storage System
* File-Based Storage System
* Object-Based Storage System

**1. Block-Based Storage System –**

* Hard drives are block-based storage systems. Your operating system like Windows or Linux actually sees a hard disk drive. So, it sees a drive on which you can create a volume, and then you can partition that volume and format them.
* For example, If a system has 1000 GB of volume, then we can partition it into 800 GB and 200 GB for local C and local D drives respectively.
* Remember with a block-based storage system, your computer would see a drive, and then you can create volumes and partitions.

**2. File-Based Storage System–**

* In this, you are actually connecting through a Network Interface Card (NIC). You are going over a network, and then you can access the network-attached storage server (NAS). NAS devices are file-based storage systems.
* This storage server is another computing device that has another disk in it. It is already created a file system so that it’s already formatted its partitions, and it will share its file systems over the network. Here, you can actually map the drive to its network location.
* In this, like the previous one, there is no need to partition and format the volume by the user. It’s already done in file-based storage systems. So, the operating system sees a file system that is mapped to a local drive letter.

**3. Object-Based Storage System –**

* In this, a user uploads objects using a web browser and uploads an object to a container i.e., Object Storage Container. This uses the HTTP Protocols with the rest of the APIs (for example: GET, PUT, POST, SELECT, DELETE).
* For example, when you connect to any website, you need to download some images, text, or anything that the website contains. For that, it is a code HTTP GET request. If you want to review any product then you can use PUT and POST requests.
* Also, there is no hierarchy of objects in the container. Every file is on the same level in an Object-Based storage system.

**3. MAP REDUCE**

MapReduce is a programming model that is used to process large amounts of data in a distributed and parallel manner in a cloud environment. It is a popular model for parallel data processing in the cloud because of its scalability, simplicity, and ability to parallelize complex tasks.

Here are some applications of MapReduce in the cloud:

* **Analyzing application logs**
* **Aggregating data from external sources**
* **Transforming data from one format to another**
* **Exporting data for external analysis**
* **Human genome decoding**
* **Natural language processing**
* **Analyzing large datasets for AI applications and machine learning systems**

MapReduce works in two phases: Map and Reduce. The programmer specifies two functions, Map and Reduce, to write an application using the MapReduce framework. The inputs to these functions are key-value pairs.

The Hadoop project is the main open source implementation of MapReduce and is widely used in the cloud computing community.

**4. SOCIAL MEDIA ANALYTICS**

Cloud-based social media analytics can be used in many ways, including:

* **Data storage**

Cloud-based data storage allows users to access, store, and retrieve files from any device with an internet connection. This can help organizations save money by only paying for the amount of cloud storage they use.

* **Data integration**

Cloud analytics can integrate data from various sources, such as databases, applications, and IoT devices. This can help businesses create a comprehensive view of their operations.

* **Collaboration**

Cloud-based analytics can make it easier for employees, partners, and customers to access analytics from anywhere. This can help increase collaboration and real-time teamwork.

Social media analytics can be used in many ways, including:

* Identifying target audiences
* Analyzing audiences
* Designing marketing strategies
* Tracking campaigns
* Analyzing competitors
* Managing brands
* Managing crises

Some examples of social media analytics metrics include: Click-through rates, Conversion rates, Impressions per mille, Engagement rate, and Leads.

**Advantages of Cloud-based social media analytics**

* Agile (News - Alert)
* Easily Assessable
* Enhances efficiency
* Reduces cost
* Saves time
* Reduces errors
* Secured
* Better data governance
* Mobile friendly

**19IINFCX1: CLOUD COMPUTING**

**Unit-4**

**Python for Cloud:** Introduction- Installing Python- Data types & Data Structures- Control Flow-

Functions- Modules- Packages- File Handling-Date/Time Operations – Classes- Python for Cloud: Amazon Web Services –Google Cloud Platform - Windows Azure –Map Reduced – Packages of Interest – Designing a RESTful Web API.

**Python for Cloud: Introduction**

Python is one of the most used and best-loved computer languages. Even 30 years after its creation, it remains one of the most popular languages, with millions of users. What inspires this long-term devotion among developers? And how can Python help you build your app, program your website backend, analyze your data, and anything else you need to program in your cloud environment?

Because of its simple syntax and easy-to-read formatting, Python is designed from the start to be more readable and easy to understand. It even includes English words where other languages might use punctuation like curly brackets or semicolons.

By utilizing Python on a cloud server, you can perform multiple operations.

Many Python packages can be utilized for DevOps automation operations and infrastructure orchestration. Here are a few examples:

* [Ansible](https://www.kamatera.com/services/ansible/) – a powerful automation tool that allows you to automate tasks such as deployment, configuration management, and orchestration. It uses Python to execute tasks on remote machines and can manage both on-premises and cloud infrastructure.
* Terraform – a tool for building, changing, and versioning infrastructure. It uses a declarative syntax and can be used to manage infrastructure resources across multiple providers, such as AWS, Azure, and Google Cloud.
* Boto3 – the Amazon Web Services (AWS) Software Development Kit (SDK) for Python. It allows you to use Python for AWS services such as EC2, S3, and RDS. Boto3 can be used for tasks such as provisioning and managing AWS resources.

Python can be used efficiently in many cloud computing tasks. It can process multiple data types (text, images, audio, video), thus it is very suitable for cloud applications.

Python possesses readily available libraries that can be utilized on the cloud. For instance, Python is successfully used for the following cloud-oriented tasks:

* Cloud monitoring
* Cloud logging
* Tracing cloud
* Reporting errors in the cloud
* Help in diagnosing bottlenecks and setbacks

**DATA TYPES & DATA STRUCTURES**

A variable can contain a variety of values. On the other hand, a person's id must be stored as an integer, while their name must be stored as a string. The storage method for each of the standard data types that Python provides is specified by Python. The following is a list of the Python-defined data types.

1. Numbers
2. Sequence Type
3. Boolean
4. Set
5. Dictionary



**Numbers**

Numeric values are stored in numbers. The whole number, float, and complex qualities have a place with a Python Numbers datatype. Python offers the type() function to determine a variable's data type. The instance () capability is utilized to check whether an item has a place with a specific class.

When a number is assigned to a variable, Python generates Number objects. For instance,

a = 5

**print**("The type of a", type(a))

b = 40.5

**print**("The type of b", type(b))

c = 1+3j

**print**("The type of c", type(c))

**print**(" c is a complex number", isinstance(1+3j,complex))

**Output:**

*The type of a <class 'int'>*

*The type of b <class 'float'>*

*The type of c <class 'complex'>*

*c is complex number: True*

Python supports three kinds of numerical data.

* **Int:** Whole number worth can be any length, like numbers 10, 2, 29, - 20, - 150, and so on. An integer can be any length you want in Python. Its worth has a place with int.
* **Float:** Float stores drifting point numbers like 1.9, 9.902, 15.2, etc. It can be accurate to within 15 decimal places.
* **Complex:** An intricate number contains an arranged pair, i.e., x + iy, where x and y signify the genuine and non-existent parts separately. The complex numbers like 2.14j, 2.0 + 2.3j, etc.

**Sequence Type**

**String**

The sequence of characters in the quotation marks can be used to describe the string. A string can be defined in Python using single, double, or triple quotes.

String dealing with Python is a direct undertaking since Python gives worked-in capabilities and administrators to perform tasks in the string.

When dealing with strings, the operation "hello"+" python" returns "hello python," and the operator + is used to combine two strings.

Because the operation "Python" \*2 returns "Python," the operator \* is referred to as a repetition operator.

The Python string is demonstrated in the following example.

**Example - 1**

str = "string using double quotes"

**print**(str)

s = '''''A multiline

string'''

**print**(s)

**Output:**

*string using double quotes*

*A multiline*

*string*

**Lists**

Lists in Python are like arrays in C, but lists can contain data of different types. The things put away in the rundown are isolated with a comma (,) and encased inside square sections [].

To gain access to the list's data, we can use slice [:] operators. Like how they worked with strings, the list is handled by the concatenation operator (+) and the repetition operator (\*).

Look at the following example.

**Example:**

list1  = [1, "hi", "Python", 2]

#Checking type of given list

**print**(type(list1))

#Printing the list1

**print** (list1)

# List slicing

**print** (list1[3:])

# List slicing

**print** (list1[0:2])

# List Concatenation using + operator

**print** (list1 + list1)

# List repetation using \* operator

**print** (list1 \* 3)

**Output:**

*[1, 'hi', 'Python', 2]*

*[2]*

*[1, 'hi']*

*[1, 'hi', 'Python', 2, 1, 'hi', 'Python', 2]*

*[1, 'hi', 'Python', 2, 1, 'hi', 'Python', 2, 1, 'hi', 'Python', 2]*

**Tuple**

In many ways, a tuple is like a list. Tuples, like lists, also contain a collection of items from various data types. A parenthetical space () separates the tuple's components from one another.

Because we cannot alter the size or value of the items in a tuple, it is a read-only data structure.

Let's look at a straightforward tuple in action.

**Example:**

tup  = ("hi", "Python", 2)

# Checking type of tup

**print** (type(tup))

#Printing the tuple

**print** (tup)

# Tuple slicing

**print** (tup[1:])

**print** (tup[0:1])

# Tuple concatenation using + operator

**print** (tup + tup)

# Tuple repatation using \* operator

**print** (tup \* 3)

# Adding value to tup. It will throw an error.

t[2] = "hi"

**Output:**

*<class 'tuple'>*

*('hi', 'Python', 2)*

*('Python', 2)*

*('hi',)*

*('hi', 'Python', 2, 'hi', 'Python', 2)*

*('hi', 'Python', 2, 'hi', 'Python', 2, 'hi', 'Python', 2)*

*Traceback (most recent call last):*

*File "main.py", line 14, in <module>*

*t[2] = "hi";*

*TypeError: 'tuple' object does not support item assignment*

**Dictionary**

A dictionary is a key-value pair set arranged in any order. It stores a specific value for each key, like an associative array or a hash table. Value is any Python object, while the key can hold any primitive data type.

The comma (,) and the curly braces are used to separate the items in the dictionary.

Look at the following example.

d = {1:'Jimmy', 2:'Alex', 3:'john', 4:'mike'}

# Printing dictionary

**print** (d)

# Accesing value using keys

**print**("1st name is "+d[1])

**print**("2nd name is "+ d[4])

**print** (d.keys())

**print** (d.values())

**Output:**

*1st name is Jimmy*

*2nd name is mike*

*{1: 'Jimmy', 2: 'Alex', 3: 'john', 4: 'mike'}*

*dict\_keys([1, 2, 3, 4])*

*dict\_values(['Jimmy', 'Alex', 'john', 'mike'])*

**Boolean**

True and False are the two default values for the Boolean type. These qualities are utilized to decide the given assertion valid or misleading. The class book indicates this. False can be represented by the 0 or the letter "F," while true can be represented by any value that is not zero.

Look at the following example.

# Python program to check the boolean type

**print**(type(True))

**print**(type(False))

**print**(false)

**Output:**

*<class 'bool'>*

*<class 'bool'>*

*NameError: name 'false' is not defined*

**Set**

The data type's unordered collection is Python Set. It is iterable, mutable(can change after creation), and has remarkable components. The elements of a set have no set order; It might return the element's altered sequence. Either a sequence of elements is passed through the curly braces and separated by a comma to create the set or the built-in function set() is used to create the set. It can contain different kinds of values.

Look at the following example.

# Creating Empty set

set1 = set()

set2 = {'James', 2, 3,'Python'}

#Printing Set value

**print**(set2)

# Adding element to the set

set2.add(10)

**print**(set2)

#Removing element from the set

set2.remove(2)

**print**(set2)

**Output:**

*{3, 'Python', 'James', 2}*

*{'Python', 'James', 3, 2, 10}*

*{'Python', 'James', 3, 10}*

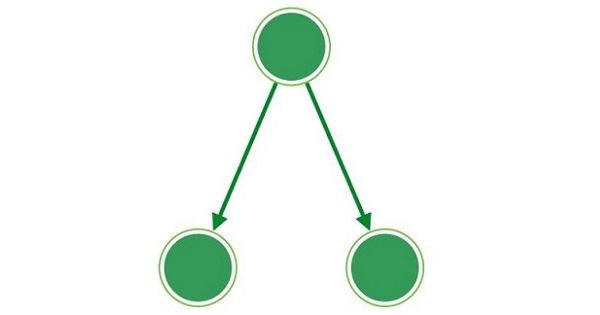
**CONTROL FLOW**

Python program control flow is regulated by various types of conditional statements, loops, and function calls. By default, the instructions in a computer program are executed in a sequential manner, from top to bottom, or from start to end. However, such sequentially executing programs can perform only simplistic tasks. We would like the program to have a decision-making ability, so that it performs different steps depending on different conditions.

## Decision Making Statements

Decision making statements are used in the Python programs to make them able to decide which of the alternative group of instructions to be executed, depending on value of a certain Boolean expression.

The following diagram illustrates how decision-making statements work



### The if Statements

Python provides [if..elif..else](https://www.tutorialspoint.com/python/python_decision_making.htm) control statements as a part of decision marking. It consists of three different blocks, which are **if** block, **elif** (short of else if) block and **else** block.

#### Example

Following is a simple example which makes use of if..elif..else. You can try to run this program using different marks and verify the result.

marks = 80

result = ""

if marks < 30:

result = "Failed"

elif marks > 75:

result = "Passed with distinction"

else: result = "Passed"

print(result)

This will produce following result:

Passed with distinction

### The match Statement

Python supports Match-Case statement, which can also be used as a part of decision making. If a pattern matches the expression, the code under that case will execute.

#### Example

Following is a simple example which makes use of match statement.

def checkVowel(n):

match n:

case 'a': return "Vowel alphabet"

case 'e': return "Vowel alphabet"

case 'i': return "Vowel alphabet"

case 'o': return "Vowel alphabet"

case 'u': return "Vowel alphabet"

case \_: return "Simple alphabet"

print (checkVowel('a'))

print (checkVowel('m'))

print (checkVowel('o'))

This will produce following result:

Vowel alphabet

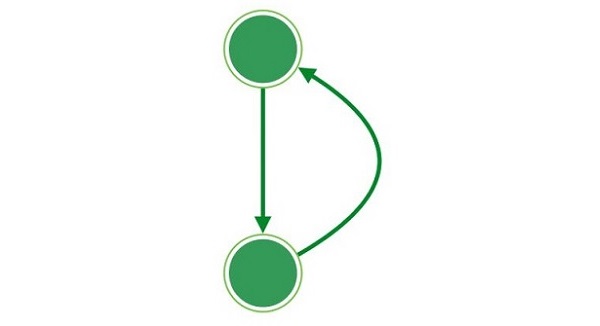
Simple alphabet

Vowel alphabet

## Loops or Iteration Statements

Most of the processes require a group of instructions to be repeatedly executed. In programming terminology, it is called a **loop**. Instead of the next step, if the flow is redirected towards any earlier step, it constitutes a loop.

The following diagram illustrates how the looping works −



If the control goes back unconditionally, it forms an infinite loop which is not desired as the rest of the code would never get executed.

In a conditional loop, the repeated iteration of block of statements goes on till a certain condition is met.

### The for Loop

The for loop iterates over the items of any sequence, such as a list, tuple or a string .

#### Example

Following is an example which makes use of For Loop to iterate through an array in Python:

words = ["one", "two", "three"]

for x in words:

print(x)

This will produce following result:

one

two

three

### The while Loop

The while loop repeatedly executes a target statement as long as a given boolean expression is true.

#### Example

Following is an example which makes use of While Loop to print first 5 numbers in Python:

i = 1

while i < 6:

print(i)

i += 1

This will produce following result:

1

2

3

4

5

**Jump Statements**

The jump statements are used to jump on a specific statement by breaking the current flow of the program. In Python, there are two jump statements break and continue.

### The break Statement

It terminates the current loop and resumes execution at the next statement.

#### Example

The following example demonstrates the use of break statement,

x = 0

while x < 10:

print("x:", x)

if x == 5:

print("Breaking...")

break

x += 1

print("End")

This will produce following result:

x: 0

x: 1

x: 2

x: 3

x: 4

x: 5

Breaking...

End

### The continue Statement

It skips the execution of the program block and returns the control to the beginning of the current loop to start the next iteration.

#### Example

The following example demonstrates the use of continue statement

for letter in "Python":

# continue when letter is 'h'

if letter == "h":

continue

print("Current Letter :", letter)

This will produce following result:

Current Letter : P

Current Letter : y

Current Letter : t

Current Letter : o

Current Letter : n

Print Page

**FUNCTIONS**

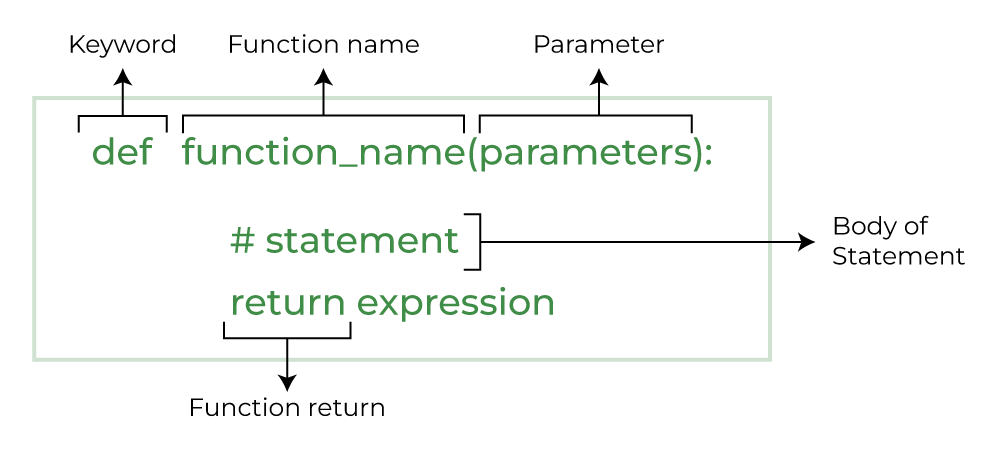
Python Functions is a block of statements that return the specific task. The idea is to put some commonly or repeatedly done tasks together and make a function so that instead of writing the same code again and again for different inputs, we can do the function calls to reuse code contained in it over and over again.

Some Benefits of Using Functions

* Increase Code Readability
* Increase Code Reusability

Python Function Declaration

The syntax to declare a function is:



**Types of Functions in Python**

Below are the different types of functions in Python:

* **Built-in library function:** These are [Standard functions](https://www.geeksforgeeks.org/python-built-in-functions/) in Python that are available to use.
* **User-defined function:** We can create our own functions based on our requirements.

**Creating a Function in Python**

We can define a function in Python, using the **def** keyword. We can add any type of functionalities and properties to it as we require. By the following example, we can understand how to write a function in Python. In this way we can create Python function definition by using def keyword.

# A simple Python function

def fun():

print("Welcome to GFG")

**Calling a Function in Python**

After creating a function in Python we can call it by using the name of the functions Python followed by parenthesis containing parameters of that particular function. Below is the example for calling def function Python.

# A simple Python function

def fun():

print("Welcome to GFG")

# Driver code to call a function

fun()

**Output:**

Welcome to GFG

**Python Function with Parameters**

If you have experience in C/C++ or Java then you must be thinking about the *return type* of the function and *data type* of arguments. That is possible in Python as well (specifically for Python 3.5 and above).

**Python Function Syntax with Parameters**

# A simple Python function

def fun():

print("Welcome to GFG")

# Driver code to call a function

fun()

The following example uses arguments and parameters.

def add(num1: int, num2: int) -> int:

"""Add two numbers"""

num3 = num1 + num2

return num3

# Driver code

num1, num2 = 5, 15

ans = add(num1, num2)

print(f"The addition of {num1} and {num2} results {ans}.")

**Output:**

The addition of 5 and 15 results 20.

**MODULES**

**Python Module** is a file that contains built-in functions, classes,**its** and variables. There are many **Python modules**, each with its specific work.

A Python module is a file containing Python definitions and statements. A module can define functions, classes, and variables. A module can also include runnable code.

Grouping related code into a module makes the code easier to understand and use. It also makes the code logically organized.

**Create a Python Module**

To create a Python module, write the desired code and save that in a file with**.py**extension. Let’s understand it better with an example:

**Example:**

Let’s create a simple calc.py in which we define two functions, one **add** and another **subtract**.

# A simple module, calc.py

def add(x, y):

return (x+y)

def subtract(x, y):

return (x-y)

**Import module in Python**

We can import the functions, and classes defined in a module to another module using the **import statement** in some other Python source file.

When the interpreter encounters an import statement, it imports the module if the module is present in the search path.

***Note****: A search path is a list of directories that the interpreter searches for importing a module.*

For example, to import the module calc.py, we need to put the following command at the top of the script.

**Syntax to Import Module in Python**

import module

**Importing modules in Python Example**

Now, we are importing the **calc** that we created earlier to perform add operation.

# importing module calc.py

import calc

print(calc.add(10, 2))

**Output:**

12

**PACKAGES**

Python Packages are a way to organize and structure your Python code into reusable components. Think of it like a folder that contains related Python files (modules) that work together to provide certain functionality. Packages help keep your code organized, make it easier to manage and maintain, and allow you to share your code with others. They’re like a toolbox where you can store and organize your tools (functions and classes) for easy access and reuse in different projects.

**How to Create Package in Python?**

Creating packages in Python allows you to organize your code into reusable and manageable modules. Here’s a brief overview of how to create packages:

* **Create a Directory:** Start by creating a directory (folder) for your package. This directory will serve as the root of your package structure.
* **Add Modules:** Within the package directory, you can add Python files (modules) containing your code. Each module should represent a distinct functionality or component of your package.
* **Init File:** Include an \_\_init\_\_.py file in the package directory. This file can be empty or can contain an initialization code for your package. It signals to Python that the directory should be treated as a package.
* **Subpackages:** You can create sub-packages within your package by adding additional directories containing modules, along with their own \_\_init\_\_.py files.
* **Importing:** To use modules from your package, import them into your Python scripts using dot notation. For example, if you have a module named module1.py inside a package named mypackage, you would import its function like this: from mypackage.module1 import greet.
* **Distribution:** If you want to distribute your package for others to use, you can create a setup.py file using Python’s setuptools library. This file defines metadata about your package and specifies how it should be installed.

**Code Example**

Here’s a basic code sample demonstrating how to create a simple Python package:

1. Create a directory named mypackage.
2. Inside mypackage, create two Python files: module1.py and module2.py.
3. Create an \_\_init\_\_.py file inside mypackage (it can be empty).
4. Add some code to the modules.
5. Finally, demonstrate how to import and use the modules from the package.

mypackage/  
│  
├── \_\_init\_\_.py  
├── module1.py  
└── module2.py

**Example:**Now, let’s create a Python script outside the mypackage directory to import and use these modules:

# module1.py

def greet(name):

print(f"Hello, {name}!")

When you run the script, you should see the following output:

Hello, Alice!  
The result of addition is: 8

**FILE HANDLING**

Python supports file handling and allows users to handle files i.e., to read and write files, along with many other file handling options, to operate on files. The concept of file handling has stretched over various other languages, but the implementation is either complicated or lengthy, like other concepts of Python, this concept here is also easy and short.

Python treats files differently as text or binary and this is important. Each line of code includes a sequence of characters, and they form a text file. Each line of a file is terminated with a special character, called the **EOL**or **End of Line**characters like **comma {,}**or **newline character.**It ends the current line and tells the interpreter a new one has begun. Let’s start with the reading and writing files.

we will consider the following ” **geeks.txt**” file as an example.

Hello world  
GeeksforGeeks  
123 456

**Python File Open**

Before performing any operation on the file like reading or writing, first, we have to open that file. For this, we should use Python’s inbuilt function [open()](https://www.geeksforgeeks.org/python-open-function)but at the time of opening, we have to specify the mode, which represents the purpose of the opening file.

f = open(filename, mode)

Where the following mode is supported:

1. **r:**open an existing file for a read operation.
2. **w:**open an existing file for a write operation. If the file already contains some data, then it will be overridden but if the file is not present then it creates the file as well.
3. **a:**open an existing file for append operation. It won’t override existing data.
4. **r+:**To read and write data into the file. This mode does not override the existing data, but you can modify the data starting from the beginning of the file.
5. **w+:**To write and read data. It overwrites the previous file if one exists, it will truncate the file to zero length or create a file if it does not exist.
6. **a+:**To append and read data from the file. It won’t override existing data.

**Working in Read mode**

There is more than one way to How to read from a file in Python . Let us see how we can read the content of a file in read mode.

**Example 1:**The open command will open the Python file in the read mode and the for loop will print each line present in the file.

# a file named "geek", will be opened with the reading mode.

file = open('geek.txt', 'r')

# This will print every line one by one in the file

for each in file:

print (each)

**Output:**

Hello world  
GeeksforGeeks  
123 456

**Example 2:**In this example, we will extract a string that contains all characters in the Python file then we can use **file.read()**.

# Python code to illustrate read() mode

file = open("geeks.txt", "r")

print (file.read())

**Output:**

Hello world  
GeeksforGeeks  
123 456

**Example 3:**In this example, we will see how we can read a file using the **with**statement in Python.

# Python code to illustrate with()

with open("geeks.txt") as file:

data = file.read()

print(data)

**Output:**

Hello world  
GeeksforGeeks  
123 456

**Creating a File using the write() Function**

Just like reading a file in Python, there are a number of ways to Writing to file in Python . Let us see how we can write the content of a file using the write() function in Python.

**Working in Write Mode**

Let’s see how to create a file and how the write mode works.

**Example 1:**In this example, we will see how the write mode and the write() function is used to write in a file. The close() command terminates all the resources in use and frees the system of this particular program.

# Python code to create a file

file = open('geek.txt','w')

file.write("This is the write command")

file.write("It allows us to write in a particular file")

file.close()

**Output:**

This is the write commandIt allows us to write in a particular file

**DATE/TIME OPERATIONS**

Python has an in-built module named DateTime to deal with dates and times in numerous ways. In this article, we are going to see basic DateTime operations in Python. There are six main object classes with their respective components in the datetime module mentioned below:

1. datetime.date
2. datetime.time
3. datetime.datetime
4. datetime.tzinfo
5. datetime.timedelta
6. datetime.timezone

**1. datetime.date():**

We can generate date objects from the date class. A date object represents a date having a year, month, and day.

***Syntax:****datetime.date( year, month, day)*

**strftime to print day, month, and year in various formats. Here are some of them are:**

* **current.strftime(“%m/%d/%y”)** that prints in **month(Numeric)/date/year** format
* **current.strftime(“%b-%d-%Y”)** that prints in **month(abbreviation)-date-year** format
* **current.strftime(“%d/%m/%Y”)** that prints in **date/month/year** format
* **current.strftime(“%B %d, %Y”)**that prints in **month(words) date, year** format

**2. datetime.time():**

A time object generated from the time class represents the local time.

**Components:**

* hour
* minute
* second
* microsecond
* tzinfo

***Syntax: datetime.time(hour, minute, second, microsecond)***

**3. datetime.datetime():**

datetime.datetime() module shows the combination of a date and a time.

**Components:**

* **year**
* **month**
* **day**
* **hour**
* **minute**
* **second,**
* **microsecond**
* **tzinfo**

***Syntax: datetime.datetime( year, month, day )***

***or***

***datetime.datetime(year, month, day, hour, minute, second, microsecond)***

Current date and time using the strftime() method in different ways:

* **strftime(“%d”) gives current day**
* **strftime(“%m”) gives current month**
* **strftime(“%Y”) gives current year**
* **strftime(“%H:%M:%S”) gives current time in an hour, minute, and second format**
* **strftime(“%m/%d/%Y, %H:%M:%S”) gives date and time together**

**4. datetime.timedelta():**

It shows a duration that expresses the difference between two date, time, or datetime instances to microsecond resolution.

Here we implemented some basic functions and printed past and future days. Also, we will print some other attributes of timedelta max, min, and resolution that show maximum days and time, minimum date and time, and the smallest possible difference between non-equal timedelta objects respectively. Here we will also apply some arithmetic operations on two different dates and times.

**5. datetime.tzinfo():**

It is an abstract base class for time zone information objects. They are used by the datetime and time classes to provide a customizable notion of time adjustment.

There are the following four methods available for tzinfo base class:

* **utcoffset(self, dt):**returns the offset of the datetime instance passed as an argument
* **dst(self, dt):**dst stands for Daylight Saving Time. dst denotes advancing the clock 1 hour in summer so that darkness falls later according to the clock.  It is set to on or off. It is checked on the basis of the following elements:

***(dt.year, dt.month, dt.day, dt.hour, dt.minute, dt.second, dt.weekday(), 0, 0)***

* **tzname(self, dt):**It returns a Python String object. It is used to find the time zone name of the datetime object passed.
* **fromutc(self, dt) :** This function returns the equivalent local time and takes up the date and time of the object in UTC. It is mostly used to adjust the date and time. It is called from default datetime.astimezone() implementation. The dt.tzinfo will be passed as self, dst date and time data will be returned as an equivalent local time.

**6. datetime.timezone():**

Description: It is a class that implements the tzinfo abstract base class as a fixed offset from the UTC.

***Syntax: datetime.timezone()***

**Let’s see different Functions with description under time module:-**

| **Function** | **Description** |
| --- | --- |
| time( ) | Returns the time in floating point number in seconds |
| ctime( ) | Returns the current date and time |
| sleep( ) | Stops execution of a thread for the given duration |
| localtime( ) | Returns the date and time in time.struct\_time format |
| gmtime( ) | Returns time.struct\_time in UTC format |
| mktime( ) | Returns the seconds passed since epochs are output |
| asctime( ) | Returns a string representing the same |

**CLASSES**

The class creates a user-defined data structure, which holds its own data members and member functions, which can be accessed and used by creating an instance of that class. A class is like a blueprint for an object.

**Creating a Python Class**

Here, the class keyword indicates that you are creating a class followed by the name of the class

**class Dog:**

**sound = "bark"**

**Some points on Python class:**

* Classes are created by keyword class.
* Attributes are the variables that belong to a class.
* Attributes are always public and can be accessed using the dot (.) operator. Eg.: My class.Myattribute

**Object of Python Class**

In Python programming an Object is an instance of a Class. A class is like a blueprint while an instance is a copy of the class with *actual values*.

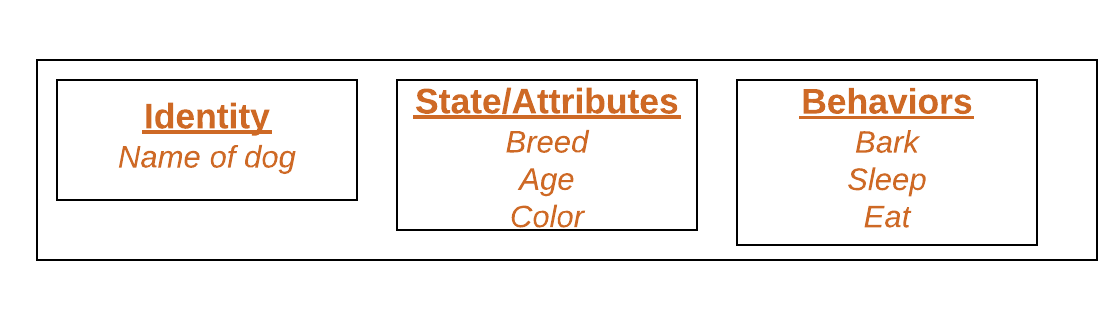
**obj = ClassName()**

**print(obj.atrr)**

It’s not an idea anymore, it’s an actual dog, like a dog of breed pug who’s seven years old. You can have many dogs to create many different instances, but without the class as a guide, you would be lost, not knowing what information is required.

An object consists of:

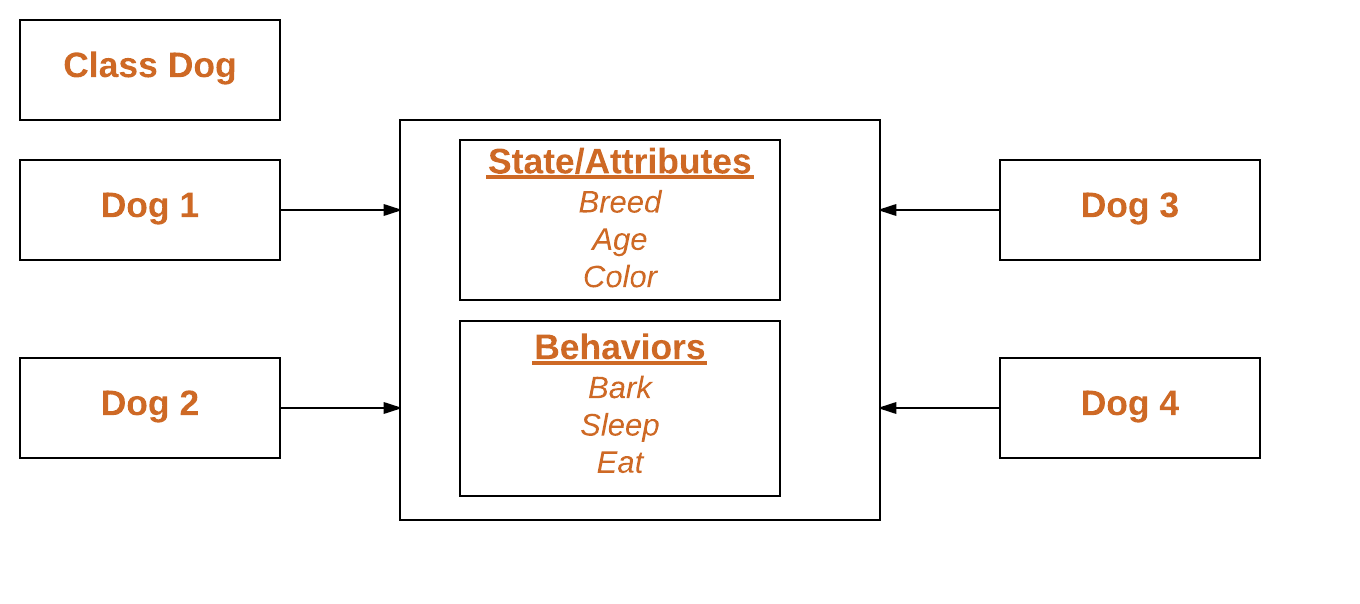
* **State:** It is represented by the attributes of an object. It also reflects the properties of an object.
* **Behavior:** It is represented by the methods of an object. It also reflects the response of an object to other objects.
* **Identity:** It gives a unique name to an object and enables one object to interact with other objects.



**Declaring Class Objects (Also called instantiating a class)**

When an object of a class is created, the class is said to be instantiated. All the instances share the attributes and the behavior of the class. But the values of those attributes, i.e. the state are unique for each object. A single class may have any number of instances.

**Example:**



**PYTHON FOR CLOUD**

**Why Python on the Cloud?**

By utilizing Python on a cloud server, you can perform multiple operations. Many Python packages can be utilized for DevOps automation operations and infrastructure orchestration. Here are a few examples:

* [Ansible](https://www.kamatera.com/services/ansible/)**–** a powerful automation tool that allows you to automate tasks such as deployment, configuration management, and orchestration. It uses Python to execute tasks on remote machines and can manage both on-premises and cloud infrastructure.
* **Terraform** – a tool for building, changing, and versioning infrastructure. It uses a declarative syntax and can be used to manage infrastructure resources across multiple providers, such as AWS, Azure, and Google Cloud.
* **Boto3** – the Amazon Web Services (AWS) Software Development Kit (SDK) for Python. It allows you to use Python for AWS services such as EC2, S3, and RDS. Boto3 can be used for tasks such as provisioning and managing AWS resources.

**Cloud Automation and Monitoring**

Python scripts can be used to automate tasks, manipulate data, or perform any number of other functions. They can be run from the command line, integrated into larger software systems, or run as standalone applications.

By utilizing Python’s asynchronous libraries, such as Asyncio, developers can manage hundreds of network connections at once, a capability needed for streamlining cloud resource management.

Python’s native support for robust logging and monitoring tasks enriches automation, thereby empowering cloud applications against failures and ensuring they meet strict latency optimization requirements.

**What Can You Do with Python on the Cloud?**

Python can be used efficiently in many cloud computing tasks. It can process multiple data types (text, images, audio, video), thus it is very suitable for cloud applications.

Python possesses readily available libraries that can be utilized on the cloud. For instance, Python is successfully used for the following cloud-oriented tasks:

* Cloud monitoring
* Cloud logging
* Tracing cloud
* Reporting errors in the cloud
* Help in diagnosing bottlenecks and setbacks

**AMAZON WEB SERVICES**

**What is AWS?**

* AWS stands for Amazon Web Services.
* The AWS service is provided by the Amazon that uses distributed IT infrastructure to provide different IT resources available on demand. It provides different services such as infrastructure as a service (IaaS), platform as a service (PaaS) and packaged software as a service (SaaS).
* Amazon launched AWS, a cloud computing platform to allow the different organizations to take advantage of reliable IT infrastructure.

**Uses of AWS**

* A small manufacturing organization uses their expertise to expand their business by leaving their IT management to the AWS.
* A large enterprise spread across the globe can utilize the AWS to deliver the training to the distributed workforce.
* An architecture consulting company can use AWS to get the high-compute rendering of construction prototype.
* A media company can use the AWS to provide different types of content such as ebox or audio files to the worldwide files.

**Pay-As-You-Go**

Based on the concept of Pay-As-You-Go, AWS provides the services to the customers. AWS provides services to customers when required without any prior commitment or upfront investment. Pay-As-You-Go enables the customers to procure services from AWS.

* Computing
* Programming models
* Database storage
* Networking



**Advantages of AWS**

**1) Flexibility**

* We can get more time for core business tasks due to the instant availability of new features and services in AWS.
* It provides effortless hosting of legacy applications. AWS does not require learning new technologies and migration of applications to the AWS provides the advanced computing and efficient storage.
* AWS also offers a choice that whether we want to run the applications and services together or not. We can also choose to run a part of the IT infrastructure in AWS and the remaining part in data centres.

**2) Cost-effectiveness**

AWS requires no upfront investment, long-term commitment, and minimum expense when compared to traditional IT infrastructure that requires a huge investment.

**3) Scalability/Elasticity**

Through AWS, autoscaling and elastic load balancing techniques are automatically scaled up or down, when demand increases or decreases respectively. AWS techniques are ideal for handling unpredictable or very high loads. Due to this reason, organizations enjoy the benefits of reduced cost and increased user satisfaction.

**4) Security**

* AWS provides end-to-end security and privacy to customers.
* AWS has a virtual infrastructure that offers optimum availability while managing full privacy and isolation of their operations.
* Customers can expect high-level of physical security because of Amazon's several years of experience in designing, developing and maintaining large-scale IT operation centers.
* AWS ensures the three aspects of security, i.e., Confidentiality, integrity, and availability of user's data.

**GOOGLE CLOUD PLATFORM**

Thomas Kurian (born 1966) is an Indian-American business executive and Chief Executive Officer of Google Cloud (under Alphabet Inc.)

**Google Cloud Platform (GCP) is a suite of cloud computing services provided by Google**. It offers a wide range of services, including computing power, storage, databases, machine learning, networking, and more, all delivered over the internet. GCP enables businesses to build, deploy, and scale applications and services quickly and efficiently without the need to invest in or manage physical infrastructure.

Starting from 1998 with the launch of Google Search. Google has developed one of the largest and most powerful IT Infrastructures in the world. Today, this infrastructure is used by billions of users to use services such as Gmail, YouTube, Google Photos, and Maps. In 2008, Google decided to open its network and IT infrastructure to business customers, taking an infrastructure that was initially developed for consumers’ applications to public service and launching the Google Cloud platform.

**History of Google Cloud Platform**

Starting from 1998 with the launch of Google Search. google has developed one of the largest and most powerful IT Infrastructures in the world. Today, this infrastructure is used by billions of users to use services such as Gmail, YouTube, Google Photos, and Maps. In 2008, Google decided to open its network and IT infrastructure to business customers, taking an infrastructure that was initially developed for consumers’ applications to public service and launching the Google Cloud platform. Over the next decade, Google expanded its offerings. Key milestones included the launch of BigQuery in 2010 for serverless analytics, Cloud Storage in 2013 and Compute Engine in 2014 offering Infrastructure-as-a-Service (IaaS). The debut of Google Kubernetes Engine (GKE) in the same year revolutionized container management setting GCP apart as a leader in cloud innovation. Today GCP is a powerhouse in cloud computing offering cutting-edge solutions that empower businesses to innovate, scale and succeed in a digital-first world.

**Google Cloud Services Interaction**

Google Cloud Platform (GCP) offers three primary methods for interacting with its services and resources:

**1. Google Cloud Console**

The Google Cloud Console is a web-based, graphical interface that allows you to manage and configure your GCP projects and resources. You can either create a new project or select an existing one to use resources within the project. The console provides an easy-to-navigate dashboard to monitor and control various Google Cloud services.

**2. Command-Line Interface (CLI)**

For those who prefer command-line operations, Google Cloud provides the Cloud SDK, which includes the gcloud CLI. This tool allows you to manage GCP resources directly from a terminal window. For example, to create a Compute Engine virtual machine (VM), you can use the `gcloud compute instances create` command. You can use the gcloud CLI in two ways:

* Install it locally on your computer.
* Use Cloud Shell, a browser-based terminal environment accessible directly from the Google Cloud Console, eliminating the need for local installation. Cloud Shell provides features such as a built-in code editor, 5 GB of persistent storage, and pre-installed tools, including the gcloud CLI. It supports multiple programming languages like Java, Python, Go, Node.js, and more.

To know more about GCP installation in windows & linux you can refer to this link How To Install GCP CLI In Linux, Windows ?

**3. Client Libraries**

Google Cloud also offers client libraries that simplify resource management and application development. These libraries expose APIs tailored to specific languages such as Python and Node.js, allowing you to interact with GCP services more intuitively. Client libraries are available for:

* App APIs for accessing services with less code and seamless integration with GCP.
* Admin APIs for managing resources, ideal for building automation tools.

Additionally, these libraries can be used for services like Google Maps, Drive, and YouTube.

**Google Cloud Platform Services**

Here are some of the services offered by Google Cloud Platform categorized by their functionalities:

| **Category** | **Service** | **Description** |
| --- | --- | --- |
| **Cloud Computing** | Compute Engine | Compute Engine lets you create and manage virtual machines, giving you control over the resources you need, like memory, storage, and security settings for your applications. |
|  | Google Kubernetes Engine (GKE) | GKE is a managed Kubernetes service that helps you deploy applications while automatically handling scaling and load balancing for you. |
|  | App Engine | App Engine is a scalable platform for running web applications. It dynamically adjusts to changing demand and provides a secure environment to ensure your app performs well. |
| **Storage** | Cloud Storage | Cloud Storage is designed to store large amounts of data that need to be highly available and easily accessed. |
|  | Persistent Disk | Persistent Disk provides durable storage that can be attached to virtual machines and reused when necessary, making it versatile for various use cases. |
|  | Cloud SQL | Cloud SQL is a fully managed database service that supports MySQL, PostgreSQL, and SQL Server, taking the hassle out of database management. |
| **Networking** | Virtual Private Cloud (VPC) | With VPC, you can run your applications inside a private network, offering more control and security for your infrastructure. |
|  | Cloud Load Balancing | This service ensures your application traffic is distributed evenly across multiple instances, helping your app stay fast and responsive. |
|  | Cloud CDN | Cloud CDN caches content and delivers it from the closest edge location to users, speeding up delivery and reducing load times. |
| **Data Analytics** | BigQuery | BigQuery is a powerful data warehouse that makes analyzing huge amounts of data easy and fast, giving organizations deep insights from their data. |
|  | Dataflow | Dataflow helps you understand how data flows through your system, allowing you to optimize and analyze data processes for better performance. |
|  | Pub/Sub | Pub/Sub is a messaging system that decouples services, allowing them to communicate asynchronously, improving system efficiency and preventing bottlenecks. |
| **Machine Learning** | Vertex AI Platform | Vertex AI Platform is Google Cloud’s suite for building and managing AI models, helping organizations unlock the potential of AI for business transformation. |
|  | AI Platform Training | AI Platform Training lets you train machine learning models in the cloud, giving you the computing power to handle complex AI tasks. |
|  | AI Platform Prediction | This service allows you to make predictions using your trained machine learning models, helping you apply AI to real-world problems. |
| **Productivity and Collaboration** | Google Workspace | Google Workspace includes popular tools like Gmail, Calendar, and Drive, helping teams collaborate seamlessly and stay organized in their daily work. |
|  | Cloud Identity and Access Management (IAM) | IAM allows administrators to control who can access what within an organization, ensuring that only authorized users have the appropriate level of access to resources. |

**Advantages and Disadvantages of Google Cloud Platform**

The following are the advantages and disadvantages of Google Cloud Platform:

| **Advantages** | **Disadvantages** |
| --- | --- |
| **Good Documentation**: Detailed API Reference guide. | **High Support Fee**: Around $150 per month for the most basic service (Silver class). |
| **Different Storage Classes**: Regional (frequent use), Nearline (infrequent use), Coldline (long-term storage). | **Expensive Data Downloads**: $0.12 per GB for downloading from Google Cloud Storage. |
| **High Durability**: Data survives even with the loss of two disks simultaneously. | **Confusing Web Interface**: Navigation can be tricky, with menus sometimes hard to follow. |
| **Multiple Regions for Data Storage**: Available in North America, South America, Europe, Asia, and Australia. | **Higher Storage Costs**: Prices are higher compared to Microsoft Azure and Backblaze B2. |
| **Console Tab for SDKs**: Try different SDKs for free, useful for developers. | **High Pricing Schema**: Similar to AWS S3, which can lead to unexpected costs (e.g., requests, transfers). |

**WINDOWS AZURE**

Azure is Microsoft’s cloud platform, just like Google has its Google Cloud and Amazon has its Amazon Web Service or AWS.000. Generally, it is a platform through which we can use Microsoft’s resources. For example, to set up a huge server, we will require huge investment, effort, physical space, and so on. In such situations, Microsoft Azure comes to our rescue. It will provide us with virtual machines, fast processing of data, analytical and monitoring tools, and so on to make our work simpler. The pricing of Azure is also simpler and more cost-effective. Popularly termed as *“Pay As You Go”*, which means how much you use, pay only for that.

**Types of Azure Services**

Microsoft Azure is a cloud computing platform which offers the following types of services:

* Infrastructure as a service ( IaaS )
* Platform as a service (PaaS)
* Software as a service (SaaS)

**Infrastructure as a service (IaaS)**

Virtual machines, storage, and networking will come under the category of infrastructure as a service but the users have to do manually the build and deploy of the applications. Azure will support a wide range of operating systems because of its Hyper-hypervisor.

**Platform as a service (PaaS)**

Azure app service, Azure functions, and logic apps are some services that are offered by Azure under the platform as a service. This service will provide autoscaling and load balancing and also there will be a pre-configured environment for the application.

**Software as a service (SaaS)**

Office 365, Dynamics 365, and Azure Active Directory are some of the services provided by Microsoft Azure under Software as a Service (SaaS) the complete application will be managed by the Microsoft azure including deploying, scaling and load balancing.

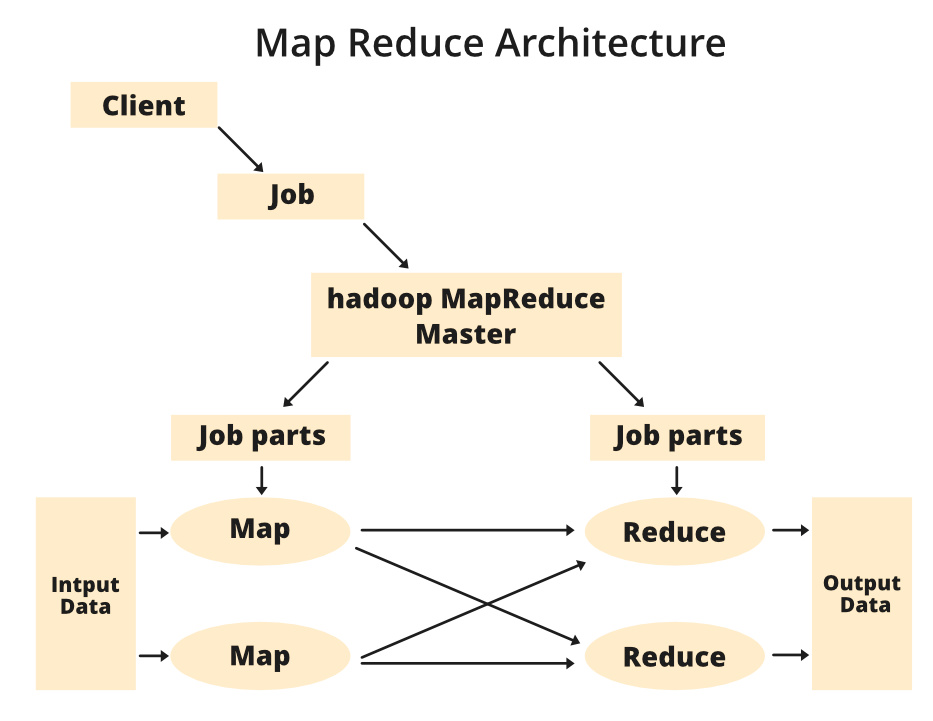
Following are the some the use cases that Microsoft Azure Used.

* **Deployment Of applications:**You can develop and deploy the application in the azure cloud by using the service called Azure App Service and Azure Functions after deploying the applications end users can access it.
* **Identity and Access Management:**The application and data which is deployed and stored in the Microsoft Azure can be secured with the help of Identity and Access Management. It’s commonly used for single sign-on, multi-factor authentication, and identity governance.
* **Data Storage and Databases:**You can store the data in Microsoft azure in service like blob storage for unstructured data, table storage for NoSQL data, file storage, and Azure SQL Database for relational databases. The service can be scaled depending on the amount of data we are getting.
* **DevOps and Continuous Integration/Continuous Deployment (CI/CD):**Azure [DevOps](https://www.geeksforgeeks.org/how-devops-works/) will provide some tools like including version control, build automation, release management, and application monitoring.

**MAP REDUCE**

[MapReduce](https://www.geeksforgeeks.org/map-reduce-in-hadoop/) and HDFS are the two major components of [Hadoop](https://www.geeksforgeeks.org/hadoop-introduction/) which makes it so powerful and efficient to use. MapReduce is a programming model used for efficient processing in parallel over large data-sets in a distributed manner. The data is first split and then combined to produce the final result. The libraries for MapReduce is written in so many programming languages with various different-different optimizations. The purpose of MapReduce in Hadoop is to Map each of the jobs and then it will reduce it to equivalent tasks for providing less overhead over the cluster network and to reduce the processing power. The MapReduce task is mainly divided into two phases Map Phase and Reduce Phase.

**MapReduce Architecture:**



**Components of MapReduce Architecture:**

1. **Client:** The MapReduce client is the one who brings the Job to the MapReduce for processing. There can be multiple clients available that continuously send jobs for processing to the Hadoop MapReduce Manager.
2. **Job:** The MapReduce Job is the actual work that the client wanted to do which is comprised of so many smaller tasks that the client wants to process or execute.
3. **Hadoop MapReduce Master:** It divides the particular job into subsequent job-parts.
4. **Job-Parts:**  The task or sub-jobs that are obtained after dividing the main job. The result of all the job-parts combined to produce the final output.
5. **Input Data:** The data set that is fed to the MapReduce for processing.
6. **Output Data:** The final result is obtained after the processing.

**PACKAGES OF INTEREST**

The total number of Python packages in existence exceeds **200,000** (and that figure only includes those stored on PyPI, the official Python Package Index). Python packages offer a user-friendly and effective solution to tackle challenging issues in a variety of domains, including scientific computing, data visualisation, data modelling, and many more.

Some of the most significant Python packages are listed below:

**1. NumPy**

Without any specialised Python packages, we can do fundamental mathematical operations. However, the NumPy module will make our lives as programmers' lot easier if we need to perform any kind of difficult math.

Tools are available in NumPy to assist in creating multi-dimensional arrays and carrying out calculations on the data they contain. We can complete common statistical procedures, solve algebraic equations, and much more.

**2. Pendulum**

One well-liked Python Date Time library for simple Date Time manipulation is the pendulum. It offers a more streamlined and user-friendly API. It makes the issue of intricate date manipulations including time zones, which are not properly handled in native datetime instances, more manageable.

If user have now done any Python programming at all, we're undoubtedly already aware of how to utilise the datetime module to control dates and times in an application.

**3. Pandas**

Pandas is the go-to Python library if we work with tabular, time series, or matrix data. It is regarded as a quick, effective, and user-friendly tool for data analysis and modification. A data frame is a unique kind of two-dimensional data structure that works with data frame elements. Like Excel spreadsheets or database tables, data frames contain rows and columns.

Pandas can be utilized for a variety of purposes, such as:

* Transferring and reading data among SQL databases, Excel, and CSV files.
* altering and rotating datasets.
* datasets that have been sliced, indexed, and subset.
* combining and changing data.
* the joining and merging of datasets.

## 4. Matplotlib

The most popular data exploration and visualisation library is Matplotlib. It can be used to create simple graphs such as linear plots, histograms, scatter graph, bar graphs, and pie charts. With the help of this library, you can also produce animated and interactive visualisations. Matplotlib is the foundation of every other visualisation library.

**5. MoviePy**

MoviePy is to films what Pillow is to photographs. For common import, edit, and export procedures involving video files, it provides several functions. Additionally, it enables us to add titles to videos and rotate them 90 ° (if for some reason we decide we want to do that).

Like Pillow, MoviePy is not intended to function as a tool for sophisticated data manipulation. If we're creating a video editing application, we'll likely also need to rely on OpenCV (which supports both videos and photos) to add the sophisticated features that MoviePy is lacking. However, MoviePy does a good job of handling most common video-related Python programming tasks.

**DESIGNING A RESTFUL WEB API**

A REST API (also called a RESTful API or RESTful web API) is an application programming interface (API) that conforms to the design principles of the representational state transfer (REST) architectural style.

A RESTful API is an architectural style for an application programming interface that uses HTTP requests to access and use data. That data can be used to GET , PUT , POST and DELETE data types, which refers to reading, updating, creating and deleting operations related to resources.

**How to Make a REST API**

1. Identify the Resources – Object Modeling: Begin by pinpointing the core entities or resources for your API. ...
2. Create Model URIs: Design intuitive URIs for each resource. ...
3. Determine Resource Representations: Decide on the data format for your resources.

**The steps are:**

1. Step 1: Determine what the API is intended to do. The first step in the API design process is for all stakeholders to agree on the API's business use case. ...
2. Step 2: Define the API contract with a specification. ...
3. Step 3: Validate your assumptions with mocks and tests. ...
4. Step 4: Document the API.

**The following are some of the principles of the REST architectural style:**

* Uniform interface. The uniform interface is fundamental to the design of any RESTful webservice.
* Statelessness
* Layered system.
* Cacheability.
* Code on demand.

**REST Key Components**  
  
**Representations:** Data format representing resource state (e.g., JSON, XML).

**Hypermedia Links:** Embedded links in responses to enable dynamic navigation.

**Status Codes:** Indicate the status of a request (e.g., 200 OK, 404 Not Found).

**19IINFCX1: CLOUD COMPUTING**

**Unit-5**

**Big Data Analytics, Multimedia Cloud & Cloud Security:** Big Data Analytics: Clustering Big

data - Classification of Big Data – Recommendation systems. Multimedia Cloud: Case Study: Live Video Stream App - Streaming Protocols – Case Study: Video Transcoding App-Cloud Security: CSA Cloud Security Architecture - Authentication - Authorization - Identity and Access management - Data Security - Key Management- Auditing- Cloud for Industry, Healthcare & Education.

**Current Streams of thought:**Service Modeling, Infrastructure Services, Platform Services, Software Services - Software as service modes, Moving existing data to cloud- Using the Wave

approach.

**BIG DATA ANALYTICS**

**What is big data analytics?**

Big data analytics refers to the methods, tools, and applications used to collect, process, and derive insights from varied, high-volume, high-velocity data sets. These data sets may come from a variety of sources, such as web, mobile, email, social media, and networked smart devices. They often feature data that is generated at a high speed and varied in form, ranging from structured (database tables, Excel sheets) to semi-structured (XML files, webpages) to unstructured (images, audio files).

Traditional forms of data analysis software aren't equipped to support this level of complexity and scale, which is where the systems, tools, and applications designed specifically for big data analysis come into play.

**Why is big data analytics important?**

Now you know what big data analytics is. But why does it matter? And most importantly, how can the understanding and use of big data assist us?

Data is woven into the everyday fabric of our lives. With the rise of mobile, social media, and smart technologies associated with the Internet of Things (IoT), we now transmit more data than ever before—and at a dizzying speed. Thanks to big data analytics, organizations can now use that information to rapidly improve the way they work, think, and provide value to their customers. With the assistance of tools and applications, big data can help you gain insights, optimize operations, and predict future outcomes.

This ability to derive insights to inform better decision making is why big data is important. It's how a retailer might hone their targeted ad campaigns, or how a wholesaler might resolve bottlenecks in the supply chain. It's also how a health care provider might discover new options for clinical care based on patient data trends. Big data analytics enables a more holistic, data-driven approach to decision-making, in turn promoting growth, efficiency, and innovation.

**How does big data analytics work?**

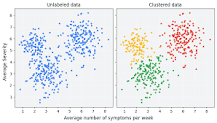
Analytics solutions glean insights and predict outcomes by analyzing data sets. However, in order for the data to be successfully analyzed, it must first be stored, organized, and cleaned by a series of applications in an integrated, step-by-step preparation process:

* **Collect.** The data, which comes in structured, semi-structured, and unstructured forms, is collected from multiple sources across web, mobile, and the cloud. It is then stored in a repository—a data lake or data warehouse—in preparation to be processed.
* **Process.** During the processing phase, the stored data is verified, sorted, and filtered, which prepares it for further use and improves the performance of queries.
* **Scrub.** After processing, the data is then scrubbed. Conflicts, redundancies, invalid or incomplete fields, and formatting errors within the data set are corrected and cleaned.
* **Analyze.** The data is now ready to be analyzed. Analyzing big data is accomplished through tools and technologies such as data mining, AI, predictive analytics, machine learning, and statistical analysis, which help define and predict patterns and behaviors in the data.

**CLUSTERING BIG DATA**

Clustering is an unsupervised machine learning algorithm that organizes and classifies different objects, data points, or observations into groups or clusters based on similarities or patterns.

Clustering is an unsupervised machine learning technique designed to group unlabeled examples based on their similarity to each other. (If the examples are labeled, this kind of grouping is called classification.) Consider a hypothetical patient study designed to evaluate a new treatment protocol.



**Types of clustering**

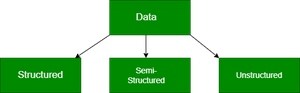
* Centroid-based clustering.
* Density-based clustering.
* Distribution-based clustering.
* Hierarchical clustering.

Clustering is used to group together common characteristics of traffic sources, then create clusters to classify and differentiate the traffic types. This allows more reliable traffic blocking while enabling better insights into driving traffic growth from desired sources. Marketing and sales.

Clusters promote both competition and cooperation. Rivals compete intensely to win and retain customers. Without vigorous competition, a cluster will fail. Yet there is also cooperation, much of it vertical, involving companies in related industries and local institutions.

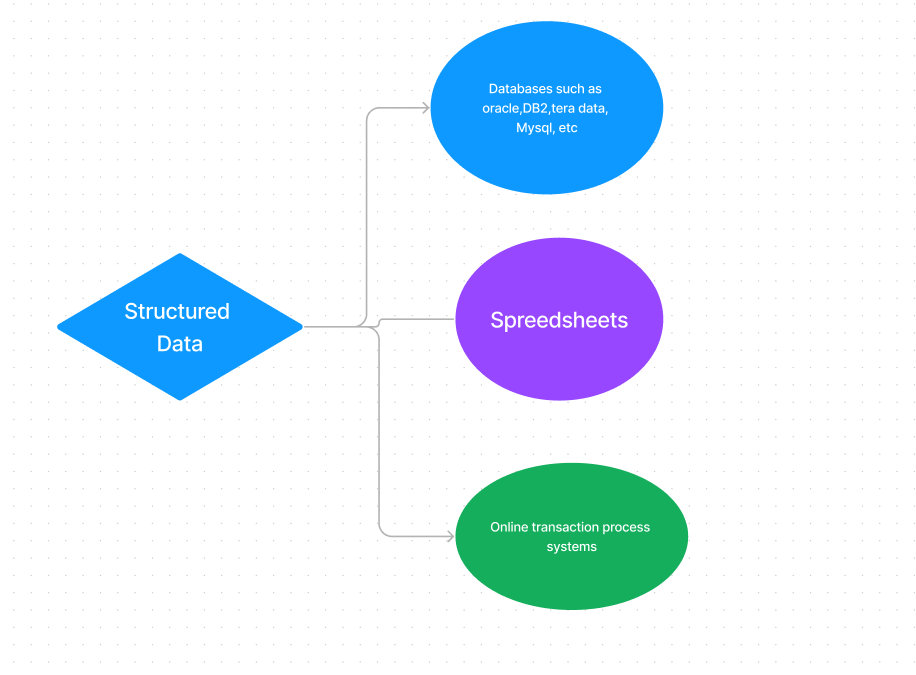
Clustering offers Increased Processing Power: By distributing tasks across multiple nodes, clusters can handle larger workloads and process data more quickly than a single computer. Improved Data Integrity: With redundancies built into the system, data loss or corruption is less likely, ensuring the integrity of your data.

**CLASSIFICATION OF BIG DATA**



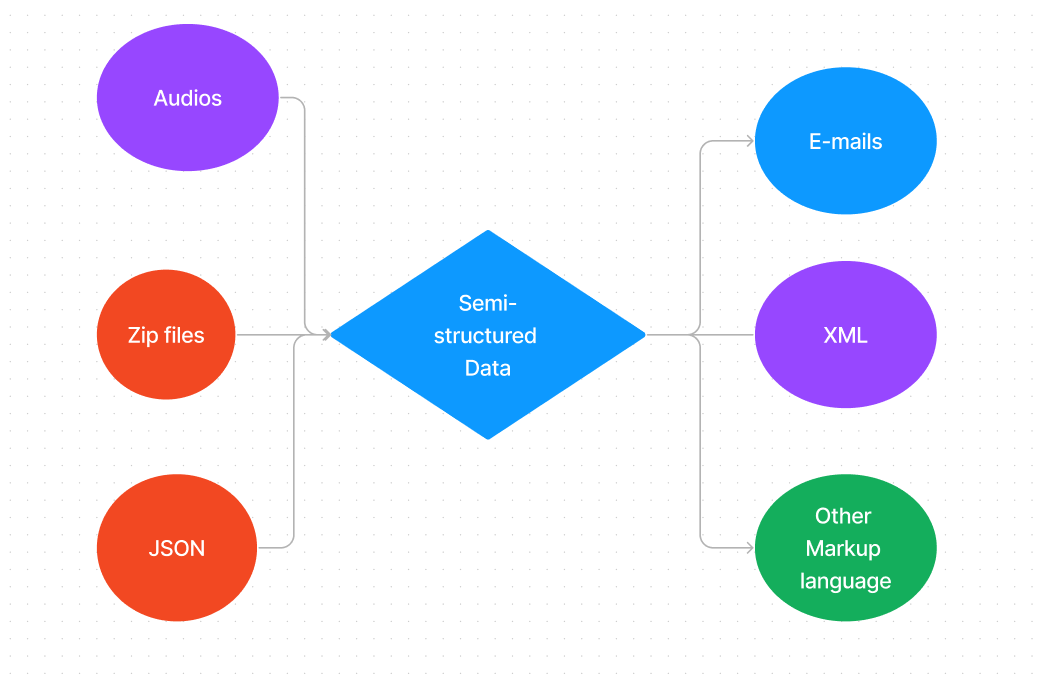
**Structured Data**

* Structured data can be crudely defined as the data that resides in a fixed field within a record.
* It is type of data most familiar to our everyday lives. for ex: birthday,address
* A certain schema binds it, so all the data has the same set of properties. Structured data is also called relational data. It is split into multiple tables to enhance the integrity of the data by creating a single record to depict an entity. Relationships are enforced by the application of table constraints.
* The business value of structured data lies within how well an organization can utilize its existing systems and processes for analysis purposes.



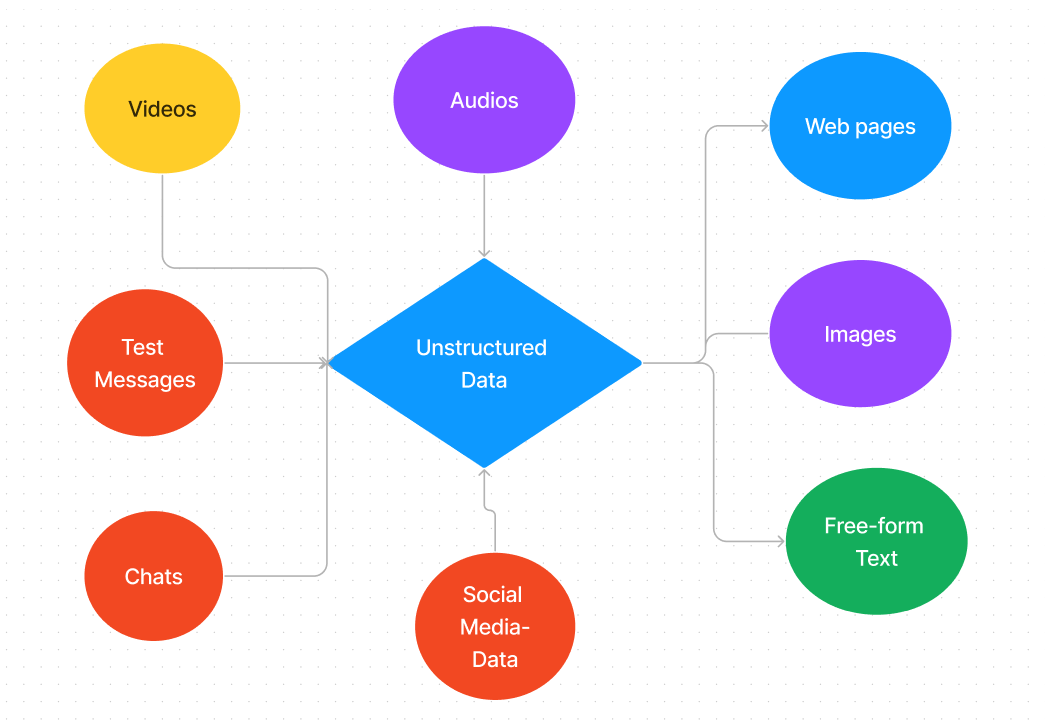
**Semi-Structured Data**

* Semi-structured data is not bound by any rigid schema for data storage and handling. The data is not in the relational format and is not neatly organized into rows and columns like that in a spreadsheet. However, there are some features like key-value pairs that help in discerning the different entities from each other.
* Since semi-structured data doesn’t need a structured query language, it is commonly called *NoSQL data*.
* A data serialization language is used to exchange semi-structured data across systems that may even have varied underlying infrastructure.
* Semi-structured content is often used to store metadata about a business process but it can also include files containing machine instructions for computer programs.
* This type of information typically comes from external sources such as social media platforms or other web-based data feeds.



**Unstructured Data**

* Unstructured data is the kind of data that doesn’t adhere to any definite schema or set of rules. Its arrangement is unplanned and haphazard.
* Photos, videos, text documents, and log files can be generally considered unstructured data. Even though the metadata accompanying an image or a video may be semi-structured, the actual data being dealt with is unstructured.
* Additionally, Unstructured data is also known as “dark data” because it cannot be analyzed without the proper software tools.



**RECOMMENDATION SYSTEMS MULTIMEDIA CLOUD: CASE STUDY**

Netflix, YouTube, Tinder, and Amazon are all examples of recommender systems in use. The systems entice users with relevant suggestions based on the choices they make. Recommender systems can also enhance experiences for: News Websites.

Recommendation system (or recommender system) is a class of machine learning that uses data to help predict, narrow down, and find what people are looking for among an exponentially growing number of options.

Recommendation letters and references are a significant step in the application, selection, and onboarding processes. They let the hiring managers verify information submitted on the resume and also speak to an applicant's overall professional qualities.

The future of recommender systems is real-time machine learning. Here's everything you need to know about what that means for enterprise applications. The future of machine learning is real-time and, when it comes to recommender systems, companies took several steps to help lay the foundation for that this year.

**What are the 6 types of recommendation systems?**

* Collaborative Recommender system. ...
* Content-based recommender system. ...
* Demographic based recommender system. ...
* Utility based recommender system. ...
* Knowledge based recommender system. ...
* Hybrid recommender system.

**Benefits of Recommendation Systems:**

* Personalization. ...
* Increased user satisfaction. ...
* Increased consumption. ...
* Improved user engagement. ...
* Reduced churn. ...
* Improved marketing.

Recommendation letters and references are a significant step in the application, selection, and on boarding processes. They let the hiring managers verify information submitted on the resume and also speak to an applicant's overall professional qualities.

**MULTIMEDIA CLOUD**

Multimedia cloud computing is the use of cloud computing to store, process, and transmit multimedia data, such as audio, video, and images. Some research issues in multimedia cloud computing include:

* Collaborative content management workflows
* Flexible content storage, compression, and indexing
* Content personalization and contextualization
* Quality of service optimization

Multimedia computing is the use of a computer to combine text, graphics, audio, and video, and to allow users to interact, navigate, create, and communicate. Cloud computing allows users to access digital information stored on cloud platforms from anywhere. The three main types of cloud computing are public cloud, private cloud, and hybrid cloud.

For multimedia computing in a cloud, continuous bursts of multimedia data access, huge processing, and transmission in the cloud would create a threshold in a general-purpose cloud because of tough multimedia QoS requirements and large amounts of users simultaneous accesses at the Internet scale.

**CASE STUDY: LIVE VIDEO STREAM APP**

The Livestream app grants you mobile access to your Livestream account and includes the ability to view events on Livestream, edit some of your basic account and event settings, and go live to your events directly from your phone. To get the Livestream mobile app, visit the App Store (iOS) or Google Play (Android).

Streamed video content can include movies, TV shows, YouTube videos and livestreamed content. Services such as Netflix and Hulu have had great success in streaming videos to subscribers. The term streaming refers to the continual transmission of audio and video files from a server to a client.

The best live streaming options for recording include Riverside and Vimeo. The best free streaming software includes OBS and Streamlabs. The best streaming software for beginners includes Crowdcast, Dacast, and Butter. The best gaming live streaming software includes vXSplit Broadcastr, and Castr

**Best social media live streaming apps**

* Instagram.
* TikTok.
* Facebook Live.
* Twitch.
* Streamlabs.
* Riverside.
* Streamlabs Talk Studio.
* OBS.

**The 7 best video capture apps**

* Loom for quickly recording and sharing on the desktop.
* Camtasia for advanced video editing.
* OBS Studio for advanced recording tools and live streaming.
* Movavi Screen Recorder for screen capture when you're away from your desk.
* ScreenPal for affordable editing features.

**The Best Streamers**

* Disney+ Unlimited entertainment. ...
* Max: Stream HBO, TV, & Movies. The One to Watch. ...
* Hulu: Stream TV shows & movies. Watch films & download series. ...
* Peacock TV: Stream TV & Movies. Hit TV shows, movies & sports. ...
* Paramount+ Originals, Movies, Sports, TV. ...
* Amazon Prime Video. ...
* YouTube: Watch, Listen, Stream. ...
* Netflix.

**STREAMING PROTOCOLS**

A streaming protocol is a set of rules that define how data communicates from one device or system to another across the Internet. Video streaming protocols standardized the method of segmenting a video stream into smaller chunks that are more easily transmitted.

**Different Types of Protocols**

Knowing if your protocol of choice is compatible with your user's favorite streaming devices is important. Each protocol serves a purpose and brings a unique set of strengths, like low latency or high adaptability. The nine preferred protocols we will cover later on will fall into one of these three categories:

* **Legacy Protocols:** Legacy protocols use Basic Authentication (usernames and passwords required by the app) to connect to email clients, calendars, and web services.
* **HTTP-Based Protocols:** A request-response protocol, HTTP allows users to interact with web resources such as HTML files by transmitting hypertext messages between clients and servers.
* **Modern Protocols:** Typically open-source and not yet widely supported, modern protocols are cutting-edge technologies that solve some of the problems preceding video streaming protocols have had.

**Nine Common Streaming Protocols**

* HTTP Live Streaming (HLS) ...
* Dynamic Adaptive Streaming over HTTP (MPEG-DASH) ...
* WebRTC. ...
* Secure Reliable Transport (SRT) ...
* Real-Time Messaging Protocol (RTMP) ...
* Real-Time Streaming Protocol (RTSP) ...
* Transmission Control Protocol (TCP) ...
* User Datagram Protocol (UDP)

**CASE STUDY: VIDEO TRANSCODING**

Video transcoding is essential for making video files compatible across various devices and platforms, optimizing them for playback anywhere. It involves converting videos to different formats, adjusting resolution, bitrate, and codec to suit the target environment without losing quality. This process ensures videos can be easily uploaded, shared, and viewed without issues, catering to diverse technical requirements.

**The Importance Of Video Transcoding**

Video transcoding is crucial in today’s multimedia landscape, ensuring videos are accessible and provide a seamless viewing experience across diverse devices and platforms. It converts video files into various formats, catering to different internet speeds and screen resolutions. This process is key to engaging a broader audience, as it optimizes playback, minimizes buffering, and meets platform-specific requirements.

Additionally, transcoding preserves video archives by updating older formats to current standards, safeguarding content’s future accessibility. Essentially, transcoding is indispensable for maximizing content reach and viewer satisfaction, making it a fundamental aspect of digital content distribution and consumption.

**How to Transcode Video**

Transcoding video can seem intimidating at first, but with the right tools and techniques, it’s a simple process. Whether you’re a content creator, marketer, or just looking to make your videos accessible on different devices, the steps to transcoding videos are straightforward. Here’s how to transcode video with ease:

1. **Choose your transcoding software:** There are many software options available for video transcoding, ranging from free to paid. Select the option that aligns with both your requirements and financial plan. ZEGOCLOUD [Video Call API](https://www.zegocloud.com/product/video-call)offers a great video transcoding feature.
2. **Select your source video:** Choose the video file you want to transcode and select it as the source file.
3. **Choose your target format:** Determine the format that you want to transcode your video into and select it from the list of options in your transcoding software.
4. **Adjust video settings:** Based on the target format you have selected, you may need to adjust the video’s resolution, framerate, bitrate, and codec.
5. **Start the transcoding process:** Once you’ve made all necessary adjustments, start the transcoding process. This may take a while, depending on the size and length of your video.
6. **Verify the transcoded video:** After the transcoding process is complete, verify the video to make sure it has been properly transcoded and that the video and audio quality are acceptable.

**Types of Video Transcoding**

Video transcoding can take many forms, each with its own set of advantages and disadvantages. Understanding the different types of video transcoding can help you choose the best approach for your specific needs. Here are some of the most common types of video transcoding solutions:

**1. Format Conversion**

To ensure compatibility with different devices and platforms, format conversion transcoding converts a video file from one format to another, such as from AVI to MP4.

**2. Resolution Change**

When looking to reduce file size and make videos easier to upload and share, resolution change transcoding adjusts the resolution of a video, for example from 1080p to 720p.

**3. Bitrate Adjustment**

Bitrate adjustment transcoding, the process of changing the amount of data used to encode each second of video, can either reduce the file size of a video or improve its quality. That’s, lowering the bitrate can reduce the file size of a video while increasing the bitrate can improve its quality.

**4. Codec Change**

This type of transcoding involves converting a video from one codec to another, such as converting a video from H.264 to H.265. This type of transcoding is often done to improve the quality or compatibility of a video.

**5. Framerate Change**

To improve the smoothness of video playback, framerate change transcoding adjusts the framerate of a video, for example from 30 fps to 60 fps.

Choosing the right type of video transcode depends on your specific needs and goals. Whether you’re looking to improve compatibility, reduce file size, or enhance video quality, there’s a type of transcoding that can help you achieve your goals. Understanding the different types of video transcoding is important in getting the most out of your videos.

**5 Common Use Cases for Video Transcoding**

As digital video consumption continues to soar, the importance of video transcoding cannot be overstated. Transcoding ensures seamless playback across devices and platforms by converting video files into compatible formats. Let’s delve into its key applications and benefits across different industries.

**1. Streaming Services**

Video streaming platforms like Netflix, YouTube, and Amazon Prime Video often receive video content in various formats and resolutions from content creators. Transcoding is essential to optimize these videos for streaming across different devices and internet speeds. Adaptive bitrate streaming, enabled by transcoding, ensures smooth playback by dynamically adjusting video quality based on the viewer’s bandwidth.

**2. E-Learning Platforms**

E-learning platforms, such as Coursera, Udemy, and Khan Academy, host a vast array of educational video content. Transcoding allows these platforms to deliver videos in formats suitable for different devices, including smartphones, tablets, and desktop computers. By transcoding videos into multiple resolutions and bitrates, e-learning platforms ensure that students can access course materials seamlessly, regardless of their device or internet connection.

**3. Social Media Sharing**

Social media platforms like Facebook, Instagram, and TikTok have become popular channels for sharing video content. Transcoding is essential for these platforms to support a wide range of video formats and aspect ratios while maintaining optimal quality and compatibility. Transcoded videos are also optimized for fast streaming and playback, enhancing the user experience on social media feeds.

**4. Live Event Broadcasting**

Live event broadcasting, including sports events, concerts, and conferences, often requires transcoding to reach a global audience. Transcoding enables broadcasters to stream live video feeds in multiple resolutions and bitrates, accommodating viewers with varying internet speeds and device capabilities. Additionally, transcoding allows broadcasters to adapt to changing network conditions in real time, ensuring uninterrupted live streams for viewers worldwide.

**5. Video Conferencing and Webinars**

Video conferencing platforms like Zoom, Microsoft Teams, and WebEx rely on transcoding to support seamless communication between participants. Transcoding optimizes video streams for different screen sizes, internet speeds, and device capabilities, ensuring high-quality video calls even in bandwidth-constrained environments. Additionally, transcoding enables features like screen sharing and virtual backgrounds, enhancing the overall user experience during virtual meetings and webinars.

**CLOUD SECURITY**

Cloud security is a major concern in cloud implementation, so an organization have to plan for security based on some factors like below represents the three main factors on which planning of cloud security depends.

* Resources that can be moved to the cloud and test its sensitivity risk are picked.
* The type of cloud is to be considered.
* The risk in the deployment of the cloud depends on the types of cloud and service models.

**Types of Cloud Computing Security Controls :**

There are 4  types of cloud computing security controls i.e.

1. Deterrent Controls : Deterrent controls are designed to block nefarious attacks on a cloud system. These come in handy when there are insider attackers.
2. Preventive Controls : Preventive controls make the system resilient to attacks by eliminating vulnerabilities in it.
3. Detective Controls : It identifies and reacts to security threats and control. Some examples of  detective control software are Intrusion detection software and network security monitoring tools.
4. Corrective Controls : In the event of a security attack these controls are activated. They limit the damage caused by the attack.

**Importance of cloud security :**

For the organizations making their transition to cloud, cloud security is an essential factor while choosing a cloud provider. The attacks are getting stronger day by day and so the security needs to keep up with it. For this purpose it is essential to pick a cloud provider who offers the best security and is customized with the organization’s infrastructure. Cloud security has a lot of benefits –

* Centralized security : Centralized security results in centralizing protection. As managing all the devices and endpoints is not an easy task cloud security helps in doing so. This results in enhancing traffic analysis and web filtering which means less policy and software updates.
* Reduced costs : Investing in cloud computing and cloud security results in less expenditure in hardware and also less manpower in administration
* Reduced Administration : It makes it easier to administer the organization and does not have manual security configuration and constant security updates.
* Reliability : These are very reliable and the cloud can be accessed from anywhere with any device with proper authorization.

When we are thinking about cloud security it includes various types of security like access control for authorized access, network segmentation for maintaining isolated data, encryption for encoded data transfer, vulnerability check for patching vulnerable areas, security monitoring for keeping eye on various security attacks and disaster recovery for backup and recovery during data loss.

There are different types of security techniques which are implemented to make the cloud computing system more secure such as SSL (Secure Socket Layer) Encryption,  Multi Tenancy based Access Control, Intrusion Detection System, firewalls, penetration testing, tokenization, VPN (Virtual Private Networks), and avoiding public internet connections and many more techniques.

But the thing is not so simple how we think, even implementation of number of security techniques there is always security issues are involved for the cloud system. As cloud system is managed and accessed over internet so a lot of challenges arises during maintaining a secure cloud. Some cloud security challenges are

* Control over cloud data
* Misconfiguration
* Ever changing workload
* Access Management
* Disaster recovery

**CSA-CLOUD SECURITY ARCHITECTURE**

**Key Points to CSA Model**

* IaaS is the most basic level of service, with PaaS and SaaS next two above levels of services.
* Moving upwards, each service inherits the capabilities and security concerns of the model beneath.
* IaaS provides the infrastructure, PaaS provides the platform development environment, and SaaS provides the operating environment.
* IaaS has the lowest integrated functionality and security level, while SaaS has the highest.
* This model describes the security boundaries at which cloud service providers' responsibilities end and customers' responsibilities begin.
* Any protection mechanism below the security limit must be built into the system and maintained by the customer.

**Why is cloud security architecture important?**

The difference between "cloud security" and "cloud security architecture" is that the former is built from problem-specific measures while the latter is built from threats. A cloud security architecture can reduce or eliminate the holes in Security that point-of-solution approaches are almost certainly about to leave.

It does this by building down - defining threats starting with the users, moving to the cloud environment and service provider, and then to the applications. Cloud security architectures can also reduce redundancy in security measures, which will contribute to threat mitigation and increase both capital and operating costs.

The cloud security architecture also organizes security measures, making them more consistent and easier to implement, particularly during cloud deployments and redeployments. Security is often destroyed because it is illogical or complex, and these flaws can be identified with the proper cloud security architecture.

**Elements of cloud security architecture**

The best way to approach cloud security architecture is to start with a description of the goals. The architecture has to address three things: an attack surface represented by external access interfaces, a protected asset set that represents the information being protected, and vectors designed to perform indirect attacks anywhere, including in the cloud and attacks the system.

The goal of the cloud security architecture is accomplished through a series of functional elements. These elements are often considered separately rather than part of a coordinated architectural plan. It includes access security or access control, network security, application security, contractual Security, and monitoring, sometimes called service security. Finally, there is data protection, which are measures implemented at the protected-asset level.

**AUTHENTICATION**

Authentication is the process of verifying the identity of an individual, system, or entity to ensure that they are who they claim to be. It is a fundamental aspect of security in various contexts, including computer systems, networks, and online services. Authentication is typically the first step in gaining access to protected resources or systems.

Authentication is all about verifying someone’s identity, like asking for your Driving License on the road to confirm you’re old enough to drive. In the digital world, it’s the process of making sure someone trying to access a system or resource is who they claim to be.

Authentication is like a gatekeeper who checks credentials before granting access.

**How Authentication Works**

1. **You claim your identity:** This could be by entering a username and password, using a fingerprint scanner, or receiving a one-time code on your phone.
2. **The system verifies your claim:** It compares your provided credentials to what it already knows about you (e.g., stored passwords, and biometric data).
3. **Access granted or denied:** If the credentials match, you’re considered authenticated and allowed access. If not, you’re denied.

**Importance of Authentication**

* **Security:** Prevents unauthorized access to sensitive information and resources.
* **Privacy:** Protects your personal data from being accessed by others.
* **Accountability:** Makes it clear who is responsible for actions within a system.

**Different types of Authentication**

* **Single-factor authentication (SFA):** Uses one factor, like a password, which can be vulnerable to breaches.
* **Multi-factor authentication (MFA):** Combines several factors, like a password and a one-time code, for stronger security.
* **Biometric authentication:** Uses physical characteristics like fingerprints or facial recognition for added security.

**AUTHORIZATION**

Authorization is the process of granting or denying access to a resource or a set of resources. In the context of computer systems, networks, and information security, authorization ensures that individuals or entities are allowed to access specific resources based on their identity, permissions, and privileges.

Authorization is all about controlling who can do what in a system, application, or even in real-life situations. It’s like a set of rules that determines what someone is allowed to access or perform. Let’s break it down further:

**Imagine it like this:**

* **Authentication:** *Verifying someone’s identity*, like checking your Driving License on the road. You prove you’re who you say you are.
* **Authorization:** *Granting or denying access based on your identity and your privileges.* If you’re 18+, you’re authorized to drive the vehicle, while someone under 18 isn’t.

In a digital world, Authorization lets systems decide whether a user (or machine) should be able to:

* Read files like documents or emails.
* Edit or delete data.
* Perform actions like sending emails or managing users.

Authorization is often part of a broader security framework that includes authentication, which verifies the identity of users or systems. Once authentication is successful, authorization controls what actions or resources the authenticated entity is allowed to access.

**Key components of authorization**

1. **Access Control:** This is about deciding who gets to use what and do what. It’s like setting rules to say who can read, write, or execute certain things, like files or actions on a computer.
2. **Permissions:** These are the rules that say what someone can or can’t do. For instance, a person might have permission to read a file but not change it.
3. **Roles:** Instead of giving permissions to each person, you put them in groups called roles. Each role has a bunch of permissions, so you only have to manage roles, not every single person.
4. **Policies:** These are like the rulebook. They say when someone can or can’t do something. It might depend on the time, place, or what kind of information they’re trying to access.
5. **Token-based Access:** Some systems use tokens like special passes. When you log in, you get a token that shows who you are and what you’re allowed to do. The system then checks this token whenever you try to do something to make sure you’re allowed.

**IDENTITY AND ACCESS MANAGEMENT**

Identity Access Management is used by the root user (administrator) of the organization. The users represent one person within the organization, and the users can be grouped in that all the users will have the same privileges to the services.

Shared Responsibility Model for Identity Access Management

**Cloud Service Provider (CSP)**

* Infrastructure (Global Security of the Network)
* Configuration and Vulnerability Analysis
* Compliance Validation

**Customer**

* Users, Groups, Roles, Policies Management and Monitoring
* Use IAM tools to apply for appropriate permissions.
* Analyze access patterns and review permissions.

Identity Access Management (IAM) ensures secure access to cloud resources. For those looking to integrate IAM within a DevOps pipeline, the DevOps Engineering – Planning to Production course covers security best practices in cloud environments using DevOps.

**The Architecture of Identity Access Management**

User Management:- It consists of activities for the control and management over the identity life cycles.

**Authentication Management:-** It consists of activities for effectively controlling and managing the processes for determining which user is trying to access the services and whether those services are relevant to him or not.

**Authorization Management:-** It consists of activities for effectively controlling and managing the processes for determining which services are allowed to access according to the policies made by the administrator of the organization.

**Access Management:-** It is used in response to a request made by the user wanting to access the resources with the organization.

**Data Management and Provisioning:-** The authorization of data and identity are carried towards the IT resource through automated or manual processes.

**Monitoring and Auditing:-** Based on the defined policies the monitoring, auditing, and reporting are done by the users regarding their access to resources within the organization.

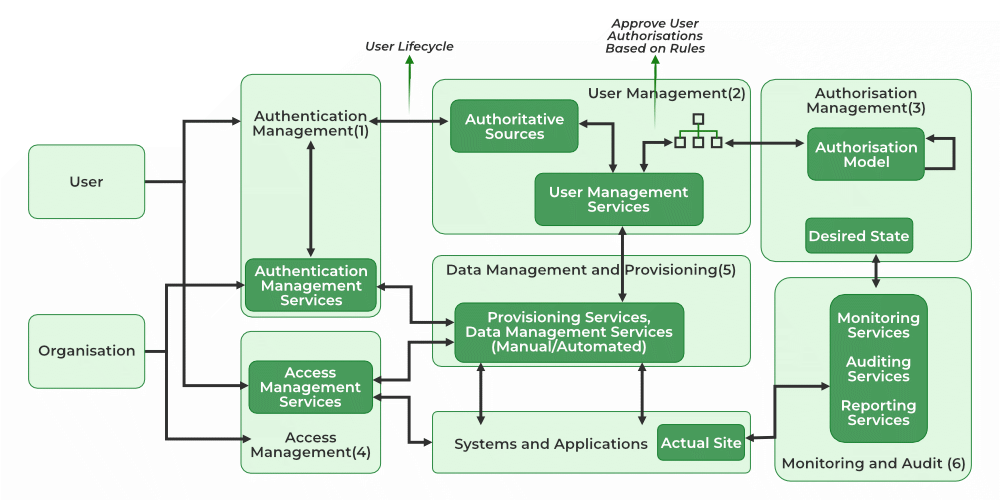
**Operational Activities of IAM:-** In this process, we onboard the new users on the organization’s system and application and provide them with necessary access to the services and data. Deprovisioning works completely opposite in that we delete or deactivate the identity of the user and de-relinquish all the privileges of the user.

**Credential and Attribute Management:-** Credentials are bound to an individual user and are verified during the authentication process. These processes generally include allotment of username, static or dynamic password, handling the password expiration, encryption management, and access policies of the user.

**Entitlement Management:-** These are also known as authorization policies in which we address the provisioning and de-provisioning of the privileges provided to the user for accessing the databases, applications, and systems. We provide only the required privileges to the users according to their roles. It can also be used for security purposes.

**Identity Federation Management:-** In this process, we manage the relationships beyond the internal networks of the organization that is among the different organizations. The federations are the associate of the organization that came together for exchanging information about the user’s resources to enable collaboration and transactions.

**Centralization of Authentication and Authorization:-** It needs to be developed in order to build custom authentication and authorization features into their application, it also promotes the loose coupling architecture.



**DATA SECURITY**

Cloud data security protects data that is stored (at rest) or moving in and out of the cloud (in motion) from security threats, unauthorized access, theft, and corruption. It relies on physical security, technology tools, access management and controls, and organizational policies.

Data privacy, integrity, and accessibility

Cloud data security best practices follow the same guiding principles of information security and data governance:

* **Data confidentiality:**Data can only be accessed or modified by authorized people or processes. In other words, you need to ensure your organization’s data is kept private.
* **Data integrity:**Data is trustworthy—in other words, it is accurate, authentic, and reliable. The key here is to implement policies or measures that prevent your data from being tampered with or deleted.
* **Data availability:**While you want to stop unauthorized access, data still needs to be available and accessible to authorized people and processes when it’s needed. You’ll need to ensure continuous uptime and keep systems, networks, and devices running smoothly.

Often referred to as the CIA triad, these three broad pillars represent the core concepts that form the basis of strong, effective security infrastructure—or any organization’s security program. Any attack, vulnerability, or other security incident will likely violate one (or more) of these principles. This is why security professionals use this framework to evaluate potential risk to an organization’s data assets.

**What are the challenges of cloud data security?**

As more data and applications move out of a central data center and away from traditional security mechanisms and infrastructure, the higher the risk of exposure becomes. While many of the foundational elements of on-premises data security remain, they must be adapted to the cloud.

**Common challenges with data protection in cloud or hybrid environments include:**

* **Lack of visibility.**Companies don’t know where all their data and applications live and what assets are in their inventory.
* **Less control.**Since data and apps are hosted on third-party infrastructure, they have less control over how data is accessed and shared.
* **Confusion over shared responsibility.**Companies and cloud providers share cloud security responsibilities, which can lead to gaps in coverage if duties and tasks are not well understood or defined.
* **Inconsistent coverage.**Many businesses are finding multicloud and hybrid cloud to better suit their business needs, but different providers offer varying levels of coverage and capabilities that can deliver inconsistent protection.
* **Growing cybersecurity threats.**Cloud databases and cloud data storage make ideal targets for online criminals looking for a big payday, especially as companies are still educating themselves about data handling and management in the cloud.
* **Strict compliance requirements.**Organizations are under pressure to comply with stringent data protection and privacy regulations, which require enforcing security policies across multiple environments and demonstrating strong data governance.
* **Distributed data storage.**Storing data on international servers can deliver lower latency and more flexibility. Still, it can also raise data sovereignty issues that might not be problematic if you were operating in your own data center.

**What are the benefits of cloud data security?**

**Greater visibility**

Strong cloud data security measures allow you to maintain visibility into the inner workings of your cloud, namely what data assets you have and where they live, who is using your cloud services, and the kind of data they are accessing.

**Easy backups and recovery**

Cloud data security can offer a number of solutions and features to help automate and standardize backups, freeing your teams from monitoring manual backups and troubleshooting problems. Cloud-based disaster recovery also lets you restore and recover data and applications in minutes.

**Cloud data compliance**

Robust cloud data security programs are designed to meet compliance obligations, including knowing where data is stored, who can access it, how it’s processed, and how it’s protected. Cloud data loss prevention (DLP) can help you easily discover, classify, and de-identify sensitive data to reduce the risk of violations.

**Data encryption**

Organizations need to be able to protect sensitive data whenever and wherever it goes. Cloud service providers help you tackle secure cloud data transfer, storage, and sharing by implementing several layers of advanced encryption for securing cloud data, both in transit and at rest.

**Lower costs**

Cloud data security reduces  total cost of ownership (TCO) and the administrative and management burden of cloud data security. In addition, cloud providers offer the latest security features and tools, making it easier for security professionals to do their jobs with automation, streamlined integration, and continuous alerting.

**Advanced incident detection and response**

An advantage of cloud data security is that providers invest in cutting-edge AI technologies and built-in security analytics that help you automatically scan for suspicious activity to identify and respond to security incidents quickly.

**KEY MANAGEMENT**

Cloud key management (KMS) is the secure administration of encryption keys in the cloud. It's a critical component of data security in the cloud, as it protects data from unauthorized access and breaches.

**Some aspects of cloud key management:**

* **Key generation, storage, and rotation**

KMS involves the generation, storage, and periodic rotation of encryption keys.

* **Access control**

KMS provides mechanisms for controlling who can access and manage encryption keys.

* **Compliance**

KMS helps organizations meet data security and compliance regulations.

* **Integration with cloud services**

KMS can integrate with cloud services to provide an extra layer of protection for cloud-stored data.

**Some cloud service providers, like Google Cloud, offer KMS services. Google Cloud KMS allows users to:**

* Create, import, and manage cryptographic keys
* Use hardware security modules (HSMs)
* Use an external key manager (EKM)
* Use Customer-Managed Encryption Keys (CMEK) integrations

**When implementing KMS, it's important to consider:**

* The threats and risks facing your organization
* Your organization's regulatory and compliance needs

**Some best practices for key management include:**

* Using cryptographic algorithms to generate keys
* Using key management systems, such as hardware security modules
* Following reference standards, such as NIST and PCI DSS
* Avoiding building your own public key infrastructure (PKI)

**AUDITING**

A cloud audit is a review of an organization's cloud environment to assess its security, compliance, and performance. The goal of a cloud audit is to ensure that the cloud environment meets industry standards and regulatory requirements, and that it's effectively mitigating potential risks.

**Here are some things that are typically assessed during a cloud audit:**

* **Security controls**: Whether the security controls are implemented correctly, working as expected, and effective at mitigating threats
* **Compliance**: Whether the cloud environment is compliant with relevant regulatory requirements and data protection and privacy regulations
* **Risk management**: How well the cloud environment is managing potential risks
* **Data access controls**: How well the cloud environment controls access to data
* **Security practices**: How well the cloud provider's security practices are

Cloud audits are usually conducted by an independent third-party auditor, but an organization can also conduct an internal audit. The audit process typically involves:

* **Data collection**: The cloud provider collects data for analysis
* **Data analysis**: The data is analyzed using test cases to identify deviations and non-functioning controls
* **Reporting**: The auditor creates a cloud compliance confirmation based on the results of the data analysis

The time and cost of a cloud audit can vary depending on the size of the cloud storage, the data stored, and the level of testing.

**CLOUD FOR INDUSTRY**

An industry cloud is a cloud-based solution that's customized to meet the needs of a specific industry. Industry clouds are also known as vertical cloud platforms. They're designed to help businesses:

* Accelerate innovation
* Improve efficiency
* Achieve competitive differentiation
* Streamline processes
* Improve the customer experience
* Create value

Industry clouds are different from general-purpose cloud services, which are designed to deliver the same functions to all organizations. Industry clouds are customized to meet the unique challenges, requirements, and regulatory constraints of a specific industry.

Industry clouds include:

Computing resources, APIs, Data models, Tools, Workflows, Security controls, and Governance controls.

Industry clouds are delivered by major public cloud service providers. Some examples of industry cloud platforms include:

* AWS
* PwC
* [Splunk](https://www.splunk.com/en_us/blog/learn/industry-cloud-platforms.html)

**Which cloud is most used in IT industry?**

**Amazon Web Services (AWS)**

Amazon Web Services (AWS) is the most widely used cloud platform, dominating the market due to its comprehensive service offerings, scalability, and strong ecosystem, making it a preferred choice for businesses worldwide.

Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP) are the big three cloud service providers today. Together, they take up 66% of the worldwide cloud infrastructure market, an increase from 63% the previous year, according to Synergy Research Group.

**HEALTHCARE & EDUCATION**

Cloud computing has many applications in healthcare and education, including:

**Healthcare**

Cloud computing can help healthcare organizations in many ways, including:

* + Patient engagement: Cloud computing can help patients participate in decisions about their health.
  + Data storage: Cloud storage can help healthcare facilities store and retrieve large amounts of patient data, such as medical histories, imaging data, and patient records.
  + Data management: Cloud computing can help streamline data management, reduce EHR latency, and make data backup and restoration quicker.
  + EHR: Electronic health records (EHR) and practice management solutions can help healthcare institutes connect with patient data and improve efficiency.

**Benefits of Cloud Computing in Healthcare Industry**

**Data Centralization and Access**

Cloud computing keeps all patient records and data in one central location. It has abilities that bring different information together. This makes diagnosing and prescribing much easier, positively affecting patient health.

Additionally, having access to multiple data saves time and reduces labor costs to maintain it. It allows them to concentrate more on patient treatment instead of administrative tasks.

**Integration**

Cloud computing in healthcare brings information together and allows it to work with different systems. This means healthcare companies can manage only a few on-site systems. Cloud computing simplifies processes instantly and gets valuable insights from analytics.

**Lower Cost**

Everybody knows investing in an in-house cloud computing system will take your time and money. Because of its building, they were deploying and maintaining. Cloud systems give healthcare companies more flexibility with costs because they only pay for the services they use. There’s no need for extra in-house IT resources to manage a cloud environment, and switching platforms is accessible if needed.

Therefore, using cloud computing in the healthcare industry can reduce costs in the long run. And those savings can be implemented on other resources.

**Education**

Cloud computing enables students and teachers to easily share resources and work collaboratively on projects in real time. Furthermore, it offers improved data storage and security, ensuring that educational materials and student information are securely stored and easily retrievable.

Cloud computing can help students and teachers in many ways, including:

* + Collaboration: Cloud computing can help students and teachers share resources and work together on projects in real time.
  + Data storage and security: Cloud computing can help ensure that educational materials and student information are securely stored and easily accessible.

Some cloud platforms that are HIPAA-compliant include:

* Microsoft Azure
* AWS
* Atlantic.Net Cloud
* Google Cloud Platform
* Oracle Cloud
* IBM Cloud

Hospitals typically use public, private, or hybrid clouds, depending on their specific needs and security requirements.

**Benefits of Cloud Computing in Education**

**Flexibility**

Cloud computing allows students and teachers to access resources from anywhere with an internet connection. This can help students learn at their own pace and from home, and teachers can spend more time on instruction.

**Collaboration**

Cloud computing can help students work together on group projects and communicate with each other and teachers. Teachers can also use cloud-based tools to create teacher management portals and share lesson plans with other teachers.

**Cost savings**

Cloud computing can help reduce costs for both students and educational institutions:

* + Textbooks: Students don't need to buy expensive textbooks.
  + Hardware: Educational institutions don't need to purchase or maintain on-site networks.
  + Data storage: Cloud computing can help reduce the cost of data storage.
* Centralized control

Cloud computing allows for centralized control over educational resources, which can make it easier to distribute resources and improve accessibility.

**Remote learning**

Cloud computing can help make remote learning possible. Students can participate in virtual classrooms, and teachers can use remote video learning to reach students who can't be in class.

**Data backup**

Cloud servers can safely back up homework and notes, so students don't need to worry about losing their work.

**CURRENT STREAMS OF THOUGHT: SERVICE MODELING**

The three main cloud computing service models are:

* **Infrastructure as a Service (IaaS)**

Provides on-demand access to cloud-hosted physical and virtual servers, storage, and networking. The service provider owns and operates the infrastructure, but the customer is responsible for purchasing and managing software.

* **Platform as a Service (PaaS)**

Provides on-demand access to a cloud-hosted platform for developing, running, maintaining, and managing applications. The provider typically offers middleware, development tools, and cloud databases.

* **Software as a Service (SaaS)**

Provides on-demand access to ready-to-use, cloud-hosted application software. The cloud service provider manages and maintains the applications.

Each model offers a different level of control and responsibility to customers. The right model or hybrid model depends on the needs of the organization.

Some newer cloud computing service models include: Database as a Service (DbaaS), Desktop as a Service (DaaS), Security as a Service, and Network as a Service (NAAS).

**SOFTWARE AS SERVICE MODES**

SaaS is also known as "On-Demand Software." It is a software distribution model in which services are hosted by a cloud service provider. These services are available to end-users over the internet, so the end-users do not need to install any software on their devices to access these services.

**Characteristics of SaaS:**

* **Web-based Delivery:** SaaS apps can be accessed from anywhere with an internet connection because they are supplied over the internet, often through a web browser. Users no longer need to install and maintain software programs on their local machines as a result.
* **Multiple Users** or "tenants" can access SaaS applications from a single instance of the program thanks to the concept of multi-tenancy. As a result, the provider can serve several clients with the same application without administering unique program instances for every client.
* **Automatic Updates:** SaaS providers are in charge of keeping the software up to date and making sure that everyone has access to the newest features and security patches. Users are no longer required to manually install updates or fixes as a result.
* **Scalable:** SaaS systems are scalable, which can readily grow or shrink in response to user demand. This frees up enterprises from worrying about infrastructure or licensing fees and lets them add or remove users as needed.
* **Pricing on a Subscription Basis:** SaaS programs are frequently sold using a subscription-based pricing model, in which customers pay a monthly or yearly price to access the program. As a result, companies won't need to invest significantly in software licenses upfront.
* **Data Security**, including data encryption, access restrictions, and backups, is the responsibility of SaaS providers. Users no longer need to handle their own data security because of this.

In conclusion, SaaS is a type of cloud computing where software applications are distributed online.

**Services Provided by SaaS:**

**Business Services** - SaaS Provider provides various business services to start up the business. The SaaS business services include ERP (Enterprise Resource Planning), CRM (Customer Relationship Management), billing, and sales.

**Document Management** - SaaS document management is a software application offered by a third party (SaaS provider) to create, manage, and track electronic documents.

Examples: Slack, Samepage, Box, and Zoho Forms.

**Social Networks -** As we all know, social networking sites are used by the general public, so social networking service providers use SaaS for their convenience and handle the general public's information.

**Mail Services** - To handle the unpredictable number of users and load on e-mail services, many e-mail providers offer their services using SaaS.

**Collaboration Tools:** SaaS companies provide collaboration solutions that let teams collaborate effectively no matter where they are physically located. Platforms for project management, apps for team communication, and file-sharing services are some of these resources.

Examples include Slack, Microsoft Office 365, and Google Workspace (formerly G Suite).

**Human Resources Management:** SaaS-based HR management systems give companies tools to simplify key HR procedures, such as employee onboarding, payroll administration, timekeeping, performance reviews, and employee self-service portals.

Workday, BambooHR, and ADP Workforce Now, as examples.

**Customer Support and Help Desk:** SaaS platforms provide customer support and help desk solutions that enable firms to manage customer inquiries, track support tickets, and promptly address customer issues.

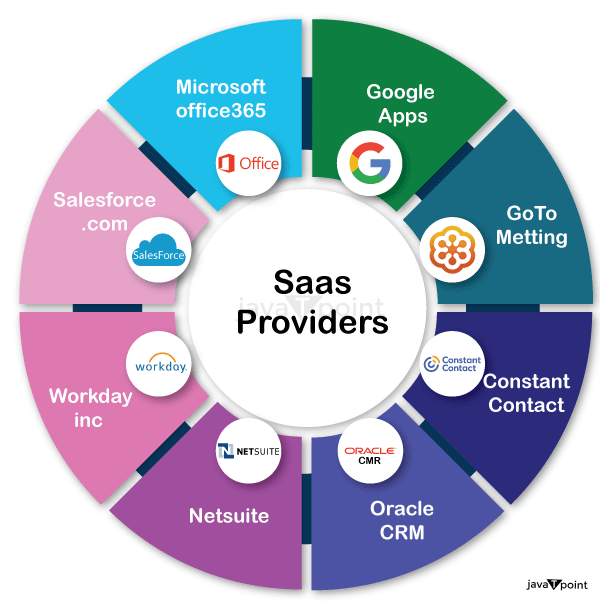
For instance, Salesforce Service Cloud, Freshdesk, and Zendesk.

**Marketing and Sales Automation:** To increase productivity and boost income, firms can automate marketing campaigns, lead generation, customer relationship management, and sales activities using SaaS marketing and sales automation technologies.

Examples include Marketo, Pardot, and HubSpot.

**E-commerce Platforms:** SaaS-based e-commerce platforms make it simpler for businesses to launch and run online storefronts, maintain product catalogs, handle payments, and keep track of orders.

**Popular SaaS Providers**



**MOVING EXISTING DATA TO CLOUD**

Cloud Migration is a transformation from old traditional business operations to digital business operations and the process refers to moving the digital business operations to cloud. That means data, applications or other business elements are moved into a cloud computing environment. For example moving data and applications from a local, on-premises data center to the cloud.

**On-premises to cloud migration process :**

Every business starting from small to large organizations follows slightly different process for cloud migrations. Some of the common elements which are considered before cloud migration are

* Evaluation of requirement and performance
* Selection of cloud provider
* Calculation of operational costs

The basic steps which are followed as follows

* Establishing migration goals
* Creating a security strategy
* Replicating existing database
* Move business intelligence
* Then switch production from on-premises to cloud

**Cloud Migration Strategy :**

5 R’s represents the cloud migration strategy.

1. **Rehost :** It refers to take the application to the new hosted cloud environment by selecting IaaS (Infrastructure as a Service).
2. **Refactor :** It refers to reuse the application code and frameworks and running the application on a PaaS (Platform as a Service).
3. **Revise :** It refers to expanding code base and then deploying it either by rehosting or refactoring.
4. **Rebuild :** It refers to re-architecting the application from the beginning up on a PaaS provider’s platform.
5. **Replace :** It refers to replacing the old application with a new built SaaS (software as a Service).

**Benefits of cloud migration :**

1. **Scalability:** Scalable enough to support various workloads and users. So it offers to expand without impacting performance.
2. **Performance:** Moving into cloud provides higher performance and customer satisfaction as compared to traditional business processes.
3. **Productivity:**As it manages the complexity of infrastructure, so improved productivity is more focused with a continuous process of growing business.
4. **Flexibility:** It allows to use the services flexibly as well as from any where and any time cloud services can be accessed as per demand/need.
5. **Cost:** Moving into cloud technology offers reduced cost in managing, operating, upgrading and maintaining IT operations or infrastructure.
6. **Security:** Security is a major concern which is taken care by cloud service providers.
7. **Profitability:** As it follows pay per use model so it delivers a greater profitability to the customers.
8. **Agility:** It is flexible enough to go with rapid changes in technology and it provides producing newer and advanced setup quickly as per requirement.
9. **Recovery:** It provides backup and recovery solutions to businesses with less time and upfront investment.

**Cloud migration Challenges:**

1. Moving a database is a difficult task as there are large amounts of data involved and mostly transferred over internet.
2. After data is transferred into cloud database, another problem is to check the transferred data is intact and secure as well as there is no data loss has been occurred during this process.
3. During migration a problem arises as some of operations or data are already moved into cloud and some are still available on-premises. So ensuring current system is operational and ensuring on going cloud migration process is taking place correctly needs a careful attention.
4. Interoperability becomes a problem as it is not easy to establish a perfect communication in between existing applications and newer cloud environments.
5. Using cloud services, getting good with newer cloud procedures, managing resources and cloud activities requires trained IT professionals who can work in the cloud eco system.

**DATA MIGRATION TO CLOUD USING WAVE APPROACH**

The wave-based approach to data migration in cloud computing involves migrating data in multiple phases, or waves. In each wave, a specific set of functionalities or business units are migrated, which allows for a gradual transition to the new system.

Here are some benefits of the wave-based approach: Minimizes disruption, Better control and management, and Testing and validation.

However, the wave-based approach also has some disadvantages: Longer timeline, Takes longer to reap benefits, and Can increase total project cost.

Other approaches to data migration in cloud computing include:

* **Rehosting**

Also known as "lift and shift," this approach involves moving data and applications from on-premises hosting to the cloud with little or no modifications.

* **Big-bang migration**

This approach involves transferring all data in a single, major phase. It's typically used when there's a planned date for decommissioning older systems.

* **Multicloud migration**

This approach involves using services or resources from multiple cloud service providers.

* **Hybrid cloud**

This approach merges the benefits of both public and private clouds.