

Table of Contents

[Cloud Computing (Advantage)](#_heading=h.gjdgxs) **4**

[**Cloud Services(IAAS,PAAS,SAAS)**](#_heading=h.fy7u7lj4udm1) **4**

[IAAS,PAAS,SAAS Difference](#_heading=h.ppvfx8lsvqxb) 6

[IAAS](#_heading=h.yx7z738f9qmz) 7

[PAAS](#_heading=h.b4t154obcm5m) 8

[SAAS](#_heading=h.1awmtegou6yr) 9

[**Deployment Models(Private, Public, Hybrid and Community)**](#_heading=h.wwnrgwd6la9z) **10**

[Public Cloud](#_heading=h.2w3rissu1s2i) 10

[Private Cloud](#_heading=h.2upt2ziar2nh) 11

[Hybrid Cloud](#_heading=h.hz17wbicn6dm) 12

[Community Cloud](#_heading=h.7g7omy4ti1ul) 13

[**Cloud Infrastructure**](#_heading=h.k9mk0t4knco4) **14**

[**Cloud Provider**](#_heading=h.jueaolvcr460) **14**

[**Characteristics of cloud computing**](#_heading=h.lrl1moe3rng1) **14**

[**Virtualization**](#_heading=h.os34ffudaplz) **15**

[What is Virtualization](#_heading=h.mapj8lrs2fo) 15

[What does Virtualization do/ How is virtualization achieved](#_heading=h.hp0l8556gyzl) 15

[Why is Virtualization required](#_heading=h.px2kb2rnb1se) 15

[Basics of Virtualization](#_heading=h.wk9y6sf54rpq) 15

[Virtualization architecture](#_heading=h.fpay8q61c0jp) 16

[The Advantages of Virtualization](#_heading=h.ftiil8wdbey1) 16

[The Disadvantages of Virtualization](#_heading=h.xz52lvfk51xa) 17

[What is x86 hardware virtualization](#_heading=h.kif78k1kv9t8) 17

[The 7 Types of Virtualizations](#_heading=h.ghymgttwahlw) 17

[Virtualization challenges and how businesses can overcome them](#_heading=h.foby0fes9ord) 18

[**Application Virtualization**](#_heading=h.v6cv4ubeg98m) **19**

[Benefits of Application Virtualization](#_heading=h.4gns6y2vz8w5) 19

[Why Application Virtualization](#_heading=h.q1c4oh4y8sgs) 19

[Server without virtualization](#_heading=h.m9d79qg7kf2m) 20

[Server with Virtualization](#_heading=h.e6vrh5fndp9) 20

[Types of Virtualizations(Type - 1 & Type - 2)](#_heading=h.cky6qfdl39t7) 21

[**Hypervisor**](#_heading=h.no247vt75uso) **22**

[Hypervisor reduces](#_heading=h.25javn2m76z6) 22

[Benefits of hypervisors](#_heading=h.s2ftuxcao9m1) 22

[What are the goals of hypervisor](#_heading=h.5i6r1yx4vuz) 23

[**MicroKernel Vs Monolithic Kernel**](#_heading=h.awhm2wxfr6xa) **24**

[MONOLITHIC ARCHITECTURE](#_heading=h.dktk0uzdckp0) 25

[Microkernel](#_heading=h.s6swvgzcg3lk) 26

[**Container vs hypervisor(VM)**](#_heading=h.2f86tn77wd6h) **26**

[Hypervisors:](#_heading=h.twakf0mxg7n7) 26

[Containers:](#_heading=h.r448lrjztb0l) 26

[**Docker**](#_heading=h.bh5kxikvtpc) **26**

[Some Docker vocabulary](#_heading=h.30zzftru7wog) 26

[Advantages of Docker:](#_heading=h.hxeejj4up81n) 27

[Disadvantages of Docker:](#_heading=h.9gnj6hv8wj73) 27

[Docker Container Lifecycle:](#_heading=h.4bpey7i0cwzr) 28

[Dockerfile:](#_heading=h.12dnoknr4sfj) 29

[**Container**](#_heading=h.tuz0dxdwukt2) **29**

[Advantages of Container:](#_heading=h.rhr61m2solvj) 29

[Disadvantages of Container:](#_heading=h.esryxv3wd8lo) 30

[Popular container providers:](#_heading=h.1zljq9ldggxm) 30

[**VirtualMachineLifeCycle**](#_heading=h.j8jkzrhuoahr) **32**

[**Kubernetes**](#_heading=h.kjvp28upbpod) **32**

[**Linux Containers - LXC and LXD**](#_heading=h.xn1hmqvtjd7) **33**

[LXC:](#_heading=h.56tsic7x0wd) 33

[LXD:](#_heading=h.vymmb8yltrn3) 33

[**Cloud orchestration**](#_heading=h.nkrjhf2xtp9u) **33**

[Three aspects to cloud orchestration:](#_heading=h.h665i2699q2k) 33

[Orchestration tools:](#_heading=h.f6a9ndvjl3hp) 33

[OpenStack components:](#_heading=h.xgbncdtq893z) 33

[**Q n A**](#_heading=h.jieyouswwjyw) **34**

[**1. Platform as a Service and SaaS**](#_heading=h.8zpjd42sq6f6) **69**

[1.1 Introduction to PaaS](#_heading=h.fwot8t9v415d) 69

[1.2 PaaS examples](#_heading=h.9piusbx0a1pb) 69

[1.3 Windows Azure](#_heading=h.musiljtlheul) 69

[1.4 5 Principles of UI Design - AWS PaaS](#_heading=h.psyus0xnvxaa) 75

[1.5 Introduction to SaaS](#_heading=h.n85w3oqjhkmc) 75

[1.6 Pros and Cons of SaaS model and applications](#_heading=h.odx4t8cdpg5i) 76

[i. The pros](#_heading=h.m58u2qikkqoc) 76

[ii. The cons](#_heading=h.vdohqrl5dokt) 77

[**2. Capacity management and Scheduling in cloud computing**](#_heading=h.1id657j0ole5) **79**

[i. Cloud capacity management tools](#_heading=h.3yoiwkney7ai) 79

[ii. Steps to manage cloud capacity](#_heading=h.tzyp1jbvozuk) 80

[1. Assess baseline capacity requirements](#_heading=h.6p0rdimf9db1) 80

[2. Assess scalability needs](#_heading=h.rurrsi97rfjl) 80

[3. Make initial resource allocations](#_heading=h.c1ydivpi8cu0) 81

[4. Set up autoscaling policies](#_heading=h.j2wjszp4z6j4) 81

[5. Collect and analyze capacity data](#_heading=h.55t92qbyc8m2) 82

[2.1 Capacity management and Scheduling](#_heading=h.30vjv1qh3pul) 83

[2.2 Distributed management of virtual machines](#_heading=h.of3gogbz9kxn) 84

[2.3 Reservation-based provisioning of virtualized resource](#_heading=h.49542bngt66a) 84

[1.1 Provisioning to meet SLA commitments](#_heading=h.px03jwetdqw9) 85

[1.2 Stages of VM life cycle within OpenNebula](#_heading=h.a6rjj7lhw0f4) 85

[1.3 6.6. Network model for OpenNebula](#_heading=h.47gut3yky2sr) 87

[**2. Issues and Challenges : Availability, Multi-Tenancy, Security and SLA**](#_heading=h.oz9x7tdxzpl1) **88**

[2.1 Multi-Tenancy, 4 levels of multi tenancy](#_heading=h.r25qgj2yhaoi) 89

[2.2 Multi-tenant models for cloud](#_heading=h.ki4wovk7b94k) 89

[Single-Tenant Vs. Multi-Tenant](#_heading=h.7rf21bgb8oja) 90

[i. Advantages Of Single-Tenant Architecture](#_heading=h.p7ek7h2r3inp) 90

[ii. Disadvantages Of Single-Tenant Architecture](#_heading=h.lb0mit6fjlu7) 90

[iii. Advantages Of Multi-Tenant Architecture](#_heading=h.2ld8rjaj1iu2) 91

[iv. Disadvantages Of Multi-Tenant Architecture](#_heading=h.np9yh5t8vjks) 91

[2.3 Introduction to cloud security](#_heading=h.b0mh3hjavois) 91

[2.4 Cloud security Issues](#_heading=h.u7txk7ndr5e1) 92

[Multi-tenancy issues in cloud computing:](#_heading=h.hznave2632dx) 92

[2.5 Threat Model](#_heading=h.2zmr3564b1dt) 94

[2.6 Top 5 cloud security threats](#_heading=h.q8ltbydmuhvz) 94

[1. Unauthorized Access to Data](#_heading=h.wfg4jw89gsp9) 94

[2. Distributed Denial of Service (DDoS) Attacks](#_heading=h.7h42h8ya689l) 95

[3. Cloud Misconfiguration](#_heading=h.4jhb8zzfnz20) 95

[4. Data Leaks and Data Breaches](#_heading=h.xbm2m1257097) 96

[5. Insecure API](#_heading=h.l7a32pcq4f) 97

[2.7 who is responsible for managing security](#_heading=h.uy71sms3c4e1) 97

[2.8 Service License Agreements: Lifecycle and Management](#_heading=h.ypquad64cqor) 98

[2.9 Traditional approaches to SLO management](#_heading=h.e71qzfohlnoq) 102

[2.10 SLA Management in Cloud](#_heading=h.v1816s1axrg) 102

[i. What Is SLA Management?](#_heading=h.nxg1229qg6k1) 102

[ii. Common Service Level Agreement Metrics](#_heading=h.9eqf83xi6axy) 103

[2.11 Automated Policy based management](#_heading=h.w90uh7k5pk4w) 103

[2.12 Managing Clouds: Services and Infrastructure](#_heading=h.srb2bwchwsv1) 104

[**3. Distributed File System (DFS) and Hadoop**](#_heading=h.b2robkoljv0e) **104**

[3.1 Introduction to Distributed File System (DFS)](#_heading=h.6khzaj49sasg) 105

[3.2 Case Study HDFS](#_heading=h.9r1y6l1wafsl) 105

[3.3 Hadoop components and importance of MapReduce](#_heading=h.w319xyviuy3u) 106

[3.4 Setting started - Amazon EMR](#_heading=h.o8vf68btfqd8) 106

[3.5 Amazon EMR - Plan and Configure clusters (# only for CSI)](#_heading=h.4xqjtu4ef39q) 107

[Plan and configure clusters:](#_heading=h.k0dx925j7hl) 107

[3.6 AMazon EMC - Manage Clusters (# only for CSI)](#_heading=h.vki98onabo15) 108

[3.7 Understanding MapReduce (\* Not for CSI)](#_heading=h.5eh4ygh3d6tw) 108

[3.8 Explore word count Java program (\* Not for CSI)](#_heading=h.xp8hw5blfth1) 108

[Work Flow of the Program](#_heading=h.ukqbow8sw2i0) 110

[Now Let’s See the Word Count Program in Java](#_heading=h.y19tf0ax85h4) 110

[iii. Steps](#_heading=h.4qbqg3xma1eq) 110

[3.9 MapReduce Facts (\* Not for CSI)](#_heading=h.ir07zh1v27ga) 111

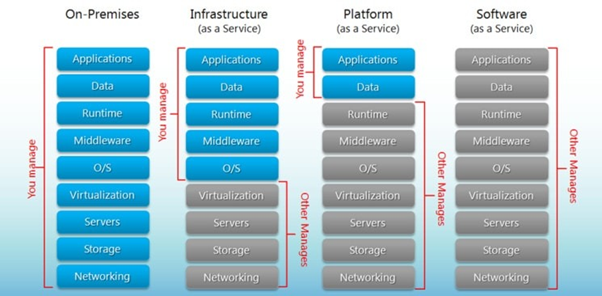
[How does MapReduce Work?](#_heading=h.fi4927cuk48t) 111

# Cloud Computing (Advantage)

* **Flexibility**: Ideal for businesses with growing or fluctuating bandwidth demands.
* **Disaster recovery**: Cloud computing helps organizations save Capex for robust Disaster Recovery.
* **Automatic software updates**: Cloud Service providers take the burden of software updates.
* **Capital-expenditure Free**: Cuts out the high cost of hardware. Pay as you go model.
* **Increased collaboration**: Cloud-based workflow and file sharing apps help teams make updates in real time and give them full visibility of their collaborations.
* **Work from anywhere**: With cloud computing, if you’ve got an internet connection you can be at work.
* **Security**: Cloud computing gives you greater security for your application and infrastructure.
* **Infrastructure as a Code**: Deploy a full scale Data Center using Orchestration Tools.
* **Time to Market**: In couple of minutes, applications can be deployed/scaled-up or scaled down

# Cloud Services(IAAS,PAAS,SAAS)

* **IaaS: cloud-based services, pay-as-you-go for services such as storage, networking, and virtualization.**
* **PaaS: hardware and software tools available over the internet.**
* **SaaS: software that’s available via a third-party over the internet.**
* **On-premise: software that’s installed in the same building as your business.**

****

## IAAS,PAAS,SAAS Difference

|  | **IAAS** | **PAAS** | **SAAS** |
| --- | --- | --- | --- |
|  | Infrastructure as a service (laaS) | Platform as a service (PaaS) | Software as a service (SaaS) |
| Description | A vendor provides clients pay-as-you-go  access to storage, networking, servers and  other computing resources in the cloud. | A service provider offers access to a cloud-  based environment in which users can build  and deliver applications. The provider  supplies underlying infrastructure. | A service provider delivers software and  applications through the internet. Users  subscribe to the software and access it Via  the web or vendor APIs. |
| Examples | AWS EC2, Rackspace,  Google Compute Engine (GCE),  Digital Ocean, Magento 1 Enterprise Edition | AWS Elastic Beanstalk, Heroku,  Windows Azure (mostly used as PaaS),  Force.com, OpenShift, Apache Stratos,  Magento Commerce Cloud. | BigCommerce, Google Apps, Salesforce,  Dropbox, MailChimp, ZenDesk, DocuSign,  Slack, Hubspot. gmail,  wordpress initially installed |
| Main Access & Management Tool | Virtual Infrastructure Manager | Cloud Development Environment | Web Browser |
| Service content | Cloud Infrastructure  Compute Servers, Data Storage,  Firewall, Load Balancer | Cloud Platform  Programming languages, Frameworks,  Mashups editors, Structured data | Cloud Applications  Social networks, Office suites, CRM,  Video processing |

## IAAS

Customers can deploy IaaS in one of three different service models:

1. Private Cloud: Infrastructure services are provisioned for exclusive use by a single organization. The physical infrastructure may be owned, managed and operated by the organization, a third party or some combination, and it may exist on or off premises.
2. Public Cloud: Infrastructure services are provisioned for use by multiple organizations (also known as a multi-tenant model). The physical infrastructure may be owned, managed and operated by a business, academic or government organization, or some combination. It exists on the premises of the cloud provider.
3. Hybrid Cloud: A company chooses to leverage both public cloud and private cloud for applications or overall architecture. The two cloud models remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability.

* + 1. **Advantages of IAAS**
* Pay for What You Use: Fees are computed via usage-based metrics
* Reduce Capital Expenditures: IaaS is typically a monthly operational expense
* Dynamically Scale: Rapidly add capacity in peak times and scale down as needed
* Increase Security: IaaS providers invest heavily in security technology and expertise
* Future-Proof: Access to state-of-the-art data center, hardware and operating systems
* Self-Service Provisioning: Access via simple internet connection
* Reallocate IT Resources: Free up IT staff for higher value projects
* Reduce Downtime: IaaS enables instant recovery from outages
* Boost Speed: Developers can begin projects once IaaS machines are provisioned
* Enable Innovation: Add new capabilities and leverage APIs
* Level the Playing Field: SMBs can compete with much larger firms
  + 1. **Disadvantages of IAAS**
* Unexpected Costs: Monthly fees can add up, or peak usage may be more than expected
* Process Changes: IaaS may require changes to processes and workflows
* Runaway Inventory: Instances may be deployed, but not taken down
* Security Risks: While IaaS providers secure the infrastructure, businesses are responsible for anything they host
* Lack of Support: Live help is sometimes hard to come by
* Complex Integration: Challenges with interaction with existing systems
* Security Risks: New vulnerabilities may emerge around the loss of direct control
* Limited Customization: Public cloud users may have limited control and ability to customize
* Vendor Lock-In: Moving from one IaaS provider to another can be challenging
* Broadband Dependency: Only as good as the reliability of the internet connection
* Providers Not Created Equally: Vendor vetting and selection can be challenging
* Managing Availability: Even the largest service providers experience downtime
* Confusing SLAs: Service level agreements (SLAs) can be difficult to understand
* Regulatory Uncertainty: Evolving federal and state laws can impact some industries’ use of IaaS, especially across country borders
* Vendor Consolidation: Providers may be acquired or go out of business
* Third-Party Expertise: Lack of mature service providers, guidance or ecosystem support
  + 1. **Examples of IaaS**
* Software development
* Software testing
* Hosting websites
* Supporting web apps
* High performance computing (HPC)
* Big data analysis

Popular IaaS providers include **Microsoft Azure, Amazon Web Services, Rackspace and Google Compute Engine.**

## PAAS

Customers can deploy PaaS in one of three different cloud deployment models:

1. Private Cloud: The development platform is built on infrastructure that is provisioned for exclusive use by a single organization comprising multiple consumers. The infrastructure may be owned, managed and operated by the organization, a third party or some combination, and it may exist on or off premises.
2. Public Cloud: The development platform is built on infrastructure provisioned for use by multiple organizations (also known as a multi-tenant model). The infrastructure may be owned, managed and operated by a business, academic or government organization, or some combination. It exists on the premises of the cloud provider.
3. Hybrid Cloud: The development platform is built across both public cloud and private cloud. The two cloud models remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability. Hybrid cloud is rarely used for PaaS solutions.
   * 1. **Advantages of PAAS**

* Cost Effective: No need to purchase hardware or pay expenses during downtime
* Time Savings: No need to spend time setting up/maintaining the core stack
* Speed to Market: Speed up the creation of apps
* Future-Proof: Access to state-of-the-art data center, hardware and operating systems
* Increase Security: PaaS providers invest heavily in security technology and expertise
* Dynamically Scale: Rapidly add capacity in peak times and scale down as needed
* Custom Solutions: Operational tools in place so developers can create custom software
* Flexibility: Allows employees to log in and work on applications from anywhere
  + 1. **Disadvantages of PAAS**
* Vendor Dependency: Very dependent upon the vendor’s capabilities
* Risk of Lock-In: Customers may get locked into a language, interface or program they no longer need
* Compatibility: Difficulties may arise if PaaS is used in conjunction with existing development platforms
* Security Risks: While PaaS providers secure the infrastructure and platform, businesses are responsible for security of the applications they build

PaaS providers include **AWS Elastic Beanstalk, Microsoft Azure App Services, Google App Engine, IBM Cloud and Red Hat OpenShift.**

## SAAS

Customers can deploy SaaS in one of three different cloud deployment models:

1. Private Cloud: Cloud software is built on infrastructure that is provisioned for exclusive use by a single organization comprising multiple consumers. The infrastructure may be owned, managed and operated by the organization, a third party or some combination, and it may exist on or off premises.
2. Public Cloud: Cloud software is built on infrastructure that is provisioned for open use by the public. The infrastructure may be owned, managed and operated by a business, academic or government organization, or some combination. It exists on the premises of the cloud provider.
3. Hybrid Cloud: Cloud software is primarily built on one type of infrastructure but has the ability to switch to another in times of high demand. Standardized or proprietary technology enables data and application portability.
   * 1. **Advantages of SAAS**

* Accessibility: Ability to run via an internet browser 24/7 from any device
* Operational Management: No installation, equipment updates or traditional licensing management
* Cost Effective: No upfront hardware costs and flexible payment methods such as pay-as-you-go models
* Scalability: Easily scale a solution to accommodate changing needs
* Data Storage: Data is routinely saved in the cloud
* Analytics: Access to data reporting and intelligence tools
* Increase Security: SaaS providers invest heavily in security technology and expertise
* Startups and small businesses will find SaaS handy when they don’t have the time, capital or expertise to build their own applications or host applications on-premises.
* Larger companies may use SaaS technology for short-term projects or applications that aren’t needed all year long.
* Any company can benefit from SaaS technology when dealing with applications that require both web and mobile access.
  + 1. **Disadvantages of SAAS**
* Loss of Control: The vendor manages everything, making you dependent upon the vendor’s capabilities
* Limited Customization: Most SaaS applications offer little in the way of customization from the vendor
* Slower Speed: SaaS solutions can have more latency than client/server apps
* Security Risks: While the SaaS provider secures the application itself, strict measures should be taken with sensitive data.

Popular software as a service examples include **Office 365, Google G Suite (Apps), Dropbox, Salesforce, SAP Concur and Zoom.**

# Deployment Models(Private, Public, Hybrid and Community)

NIST has defined four deployment models for cloud. Namely

* Private Cloud
* Public Cloud
* Community Cloud
* Hybrid Cloud

|  | **Public Cloud** | **Private Cloud** | **Hybrid Cloud** | **Community Cloud** |
| --- | --- | --- | --- | --- |
| **Description** | In public cloud, services are available for public users. | Private cloud is build up with existing private infrastructure. This type of cloud has some authentic users who can dynamically vision the resources. | Hybrid cloud heterogeneous is a distributed system, resulting from a private cloud, which incorporates different types Of services and resources from public clouds. | In monolithic kernel, both user services and kernel services are kept in the same address space. |
| **Scalablity** | Very High | Limited | Very High | Limited |
| **Reliablity** | Moderate | Very High | Medium to high | Very High |
| **Security** | Totally depends on service provider | High class security | Secure | Secure |
| **Performance** | Low to medium | Good | Good | Very good |
| **Cost** | Cheaper | High cost | Costly | Costly |
| **Example** | Amazon EC2, Google AppEngine | VMWare. Microsoft, KVM, Xen | IBM, HP, VMWare. vCloud, Eucalyptus | SolaS Community Cloud VMWare |

## Public Cloud

Public cloud is an IT model where on-demand computing services and infrastructure are managed by a third-party provider and shared with multiple organizations using the public Internet. Public cloud service providers may offer cloud-based services such as infrastructure as a service (IaaS), platform as a service (PaaS), or software as a service (Saas) to users for either a monthly or pay-per-use fee, eliminating the need for users to host these services on site in their own data center.

Cloud service providers use groups of data centers that are partitioned into virtual machines and shared by tenants. Tenants may simply rent the use of those virtual machines, or they may pay for additional cloud-based services such as software applications, application development tools, or storage. Companies often use public cloud services for less-sensitive applications that have unpredictable spikes in usage or for storing data that does not require frequent access.

Public cloud makes computing resources available to anyone for purchase. Multiple users typically share the use of a public cloud. In contrast, private cloud involves cloud-based services that are hosted within an organization’s own private servers.

* + 1. **Advantages of Public Cloud**
* Lower costs—no need to purchase hardware or software and you pay only for the service you use.
* No maintenance—your service provider provides the maintenance.
* Near-unlimited scalability—on-demand resources are available to meet your business needs.
* High reliability—a vast network of servers ensures against failure.
  + 1. **Disadvantages of Public Cloud**
* security and privacy—When access to data is easy, public clouds don’t allow users to see where the information is being kept and who has access to it.
* Lack of options—Public cloud providers usually have a one-size-fits-all approach with standard options. If a company has a unique need, they may not be able to meet those requirements.
* Loss of control—When you outsource to the public cloud, it’s literally out of reach. Any configuration and other aspects of IT management are left to a group that are not directly involved in daily operations.
  + 1. **Public Cloud Company Examples**
* Amazon Web Services
* Microsoft Azure
* IBM Cloud
* Google Cloud Platform
* Oracle Cloud Infrastructure
* Alibaba Cloud

## Private Cloud

A private cloud is a model of cloud computing where the infrastructure is dedicated to a single user organization. A private cloud can be hosted either at an organization’s own data center, at a third party colocation facility, or via a private cloud provider who offers private cloud hosting services and may or may not also offer traditional public shared multi-tenant cloud infrastructure.

Typically, the end-user organization is responsible for the operation of a private cloud as if it were a traditional on-premises infrastructure, which includes ongoing maintenance, upgrades, OS patches, middleware, and application software management.

Private cloud solutions offer organizations more control over and better security of private cloud servers, although it does require a much higher level of IT expertise than utilizing a public cloud.

* + 1. **Advantages of Private Cloud**
* More flexibility—your organisation can customise its cloud environment to meet specific business needs.
* More control—resources are not shared with others, so higher levels of control and privacy are possible.
* More scalability—private clouds often offer more scalability compared to on-premises infrastructure.
  + 1. **Disadvantages of Private Cloud**
* Cost—In terms of Price, a Private cloud model is more expensive. This is particularly because of the hardware expenses such as servers, network infrastructures, data centers and software licenses. Besides that you need to hire a separate on-site staff to look after and maintain the cloud. That is the main reason why Private cloud is most often preferred by large companies.
* Maintenance—Setup and maintenance cost is generally higher in Private cloud services. A Private cloud not only needs an investment in hardware, it requires continuous maintenance which can be time consuming. Therefore, a separate IT administration must be present for this purpose. The only way to lower this maintenance cost is through the use of a virtual environment.
* Deployment—Deployment of a Private cloud is considered to be very difficult since it requires many resources and takes much time. Especially, it is harder to deploy on a global scale. Apart from that deployment needs to be maintained by qualified IT personnel.
* Scalability—Unlike the Public cloud, the Private cloud is not much scalable. Whenever there is a demand for a particular service, the IT department will be falling short.
* Options—There are too many options available when a company tries to build a Private cloud. Some companies may be overwhelmed by the fact that there are arrays of different types and options for a particular software. They just need one or two out of them.
* Remote Access—Increased security in Private cloud means that that remote access is limited. This is especially true in the case of mobile users. Mobile users in the Private cloud will not be able to connect to the needed business functions whenever they want.
  + 1. **Private Cloud Company Examples**
* HPE
* VMware
* Dell EMC
* Oracle
* IBM / Red Hat OpenShift
* Microsoft
* Cisco
* NetApp
* AWS

## Hybrid Cloud

A hybrid cloud is a cloud computing model that uses a combination of at least one private cloud and at least one public cloud, which work together to provide a flexible mix of cloud computing services. Hybrid cloud computing extends infrastructure and operations consistently to provide a single operating model that manages application workloads across both environments, allowing for seamless migration of workloads from private to or from public cloud as business needs dictate.

Hybrid cloud solutions offer a single, seamless pool of resources that support modern application strategies and an organization’s digital transformation efforts. Most organizations have adopted hybrid cloud infrastructure to reduce risk, minimize overall IT and cloud costs, support cloud migration without refactoring, data center consolidation, and meet seasonal peaks in demand for compute and storage resources.

* + 1. **Advantages of Hybrid Cloud**
* Control—your organisation can maintain a private infrastructure for sensitive assets or workloads that require low latency.
* Flexibility—you can take advantage of additional resources in the public cloud when you need them.
* Cost-effectiveness—with the ability to scale to the public cloud, you pay for extra computing power only when needed.
* Ease—transitioning to the cloud does not have to be overwhelming because you can migrate gradually—phasing in workloads over time.
  + 1. **Disadvantages of Hybrid Cloud**
* Hardware expenses—While a Hybrid cloud is inexpensive in terms of operations, businesses must maintain and spend on on-premises hardware for handling the computing needs of the private cloud.
* Visibility—When running important IT workloads in a hybrid cloud deployment it can be difficult to maintain visibility over everything you are managing.
* Complexity—It is a complex task to manage applications, data, and workloads in a hybrid infrastructure. Instead of managing a relatively confined infrastructure architecture, companies now have many different architectures to manage. This can be difficult for companies with limited IT resources and may need an external party for additional expertise.
  + 1. **Hybrid Cloud Company Examples**
* Amazon
* Microsoft
* Google
* Cisco
* NetApp

## Community Cloud

Community cloud is a cloud infrastructure that allows systems and services to be accessible by a group of several organizations to share the information. It is owned, managed, and operated by one or more organizations in the community, a third party, or a combination of them.

* + 1. **Advantages of Community Cloud**
* Cost effective—Community cloud is cost effective because the whole cloud is shared between several organizations or a community.
* Flexible and Scalable—The community cloud is flexible and scalable because it is compatible with every user. It allows the users to modify the documents as per their needs and requirement.
* Security—Community cloud is more secure than the public cloud but less secure than the private cloud.
* Sharing infrastructure—Community cloud allows us to share cloud resources, infrastructure, and other capabilities among various organizations.
  + 1. **Disadvantages of Community Cloud**
* Community cloud is not a good choice for every organization.
* Slow adoption to data
* The fixed amount of data storage and bandwidth is shared among all community members.
* Community Cloud is costly than the public cloud.
* Sharing responsibilities among organizations is difficult.
  + 1. **Community Cloud Company Examples**
* Cisco
* Cloud4C
* Hewlett Packard Enterprise (HPE)
* IBM
* Microsoft

# Cloud Infrastructure

Cloud infrastructure refers to the hardware and software components, such as servers, storage, networking, virtualization software, services and management tools, that support the computing requirements of a cloud computing model.

Cloud infrastructure also includes an abstraction layer that virtualizes and logically presents resources and services to users through application programming interfaces and API-enabled command-line or graphical interfaces.

# Cloud Provider

A cloud service provider is a third-party company offering a cloud-based platform, infrastructure, application, or storage services. Much like a homeowner would pay for a utility such as electricity or gas, companies typically have to pay only for the amount of cloud services they use, as business demands require.

Besides the pay-per-use model, cloud service providers also give companies a wide range of benefits. Businesses can take advantage of scalability and flexibility by not being limited to physical constraints of on-premises servers, the reliability of multiple data centers with multiple redundancies, customization by configuring servers to your preferences, and responsive load balancing that can easily respond to changing demands. Businesses should also evaluate security considerations of storing information in the cloud to ensure industry-recommended access and compliance management configurations and practices are enacted and met.

# Characteristics of cloud computing

* **On-demand self-service**: Users can provision servers and networks with little human intervention
* **Broad network access**: Any computing capabilities are available over the network. Many different devices are allowed access through standardized mechanisms
* **Resource pooling**: Multiple users can access clouds that serve other consumers according to demand. Cloud services need to share resources between users and clients in order to reduce costs
* **Elasticity**: Provisioning is rapid and scales out or in based on need
* **Metered or measured service**: One of the compelling business use cases for cloud computing is the ability to "pay as you go“, where the consumer pays only for the resources that are actually used by his applications

# Virtualization

## What is Virtualization

Virtualization is technology that lets you create useful IT services using resources that are traditionally bound to hardware. It allows you to use a physical machine’s full capacity by distributing its capabilities among many users or environments.

## What does Virtualization do/ How is virtualization achieved

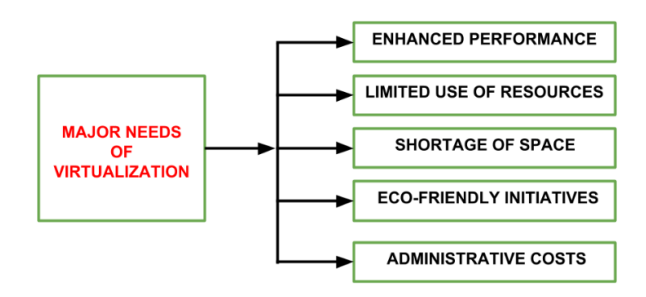
Virtualization relies on software to simulate hardware functionality and create a virtual computer system. This enables IT organizations to run more than one virtual system – and multiple operating systems and applications – on a single server. The resulting benefits include economies of scale and greater efficiency

## Why is Virtualization required

Virtualization is the act of migrating physical systems into a virtual environment.

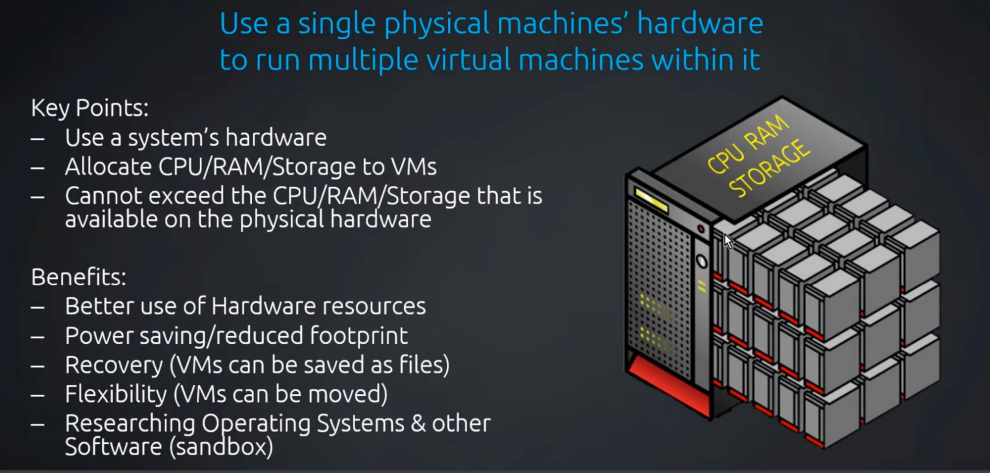
By providing a virtual view of computing resources, this allows you to turn one server into a

host for a group of servers that all share the same resources.



## Basics of Virtualization

A virtualization architecture is a conceptual model specifying the arrangement and interrelationships of the particular components involved in delivering a virtual -- rather than physical -- version of something, such as an operating system (OS), a server, a storage device or network resources



## Virtualization architecture

A virtualization architecture is a conceptual model specifying the arrangement and interrelationships of the particular components involved in delivering a virtual -- rather than physical -- version of something, such as an operating system (OS), a server, a storage device or network resources

## The Advantages of Virtualization

1. **It is cheaper**—Because virtualization doesn’t require actual hardware components to be used or installed, IT infrastructures find it to be a cheaper system to implement. There is no longer a need to dedicate large areas of space and huge monetary investments to create an on-site resource. You just purchase the license or the access from a third-party provider and begin to work, just as if the hardware were installed locally.
2. **It keeps costs predictable**—Because third-party providers typically provide virtualization options, individuals and corporations can have predictable costs for their information technology needs. For example: the cost of a Dell PowerEdge T330 Tower Server, at the time of writing, is $1,279 direct from the manufacturer. In comparison, services provided by Bluehost Web Hosting can be a slow as $2.95 per month.
3. **It reduces the workload**—Most virtualization providers automatically update their hardware and software that will be utilized. Instead of sending people to do these updates locally, they are installed by the third-party provider. This allows local IT professionals to focus on other tasks and saves even more money for individuals or corporations.
4. **It offers a better uptime**—Thanks to virtualization technologies, uptime has improved dramatically. Some providers offer an uptime that is 99.9999%. Even budget-friendly providers offer uptime at 99.99% today.
5. **It allows for faster deployment of resources**—Resource provisioning is fast and simple when virtualization is being used. There is no longer a need to set up physical machines, create local networks, or install other information technology components. As long as there is at least one point of access to the virtual environment, it can be spread to the rest of the organization.
6. **It promotes digital entrepreneurship**—Before virtualization occurred on a large scale, digital entrepreneurship was virtually impossible for the average person. Thanks to the various platforms, servers, and storage devices that are available today, almost anyone can start their own side hustle or become a business owner. Sites like Fiverr and UpWork make it possible for anyone to set a shingle and begin finding some work to do.
7. **It provides energy savings**—For most individuals and corporations, virtualization is an energy-efficient system. Because there aren’t local hardware or software options being utilized, energy consumption rates can be lowered. Instead of paying for the cooling costs of a data center and the operational costs of equipment, funds can be used for other operational expenditures over time to improve virtualization’s overall ROI.

## The Disadvantages of Virtualization

1. **It can have a high cost of implementation**—The cost for the average individual or business when virtualization is being considered will be quite low. For the providers of a virtualization environment, however, the implementation costs can be quite high. Hardware and software are required at some point and that means devices must either be developed, manufactured, or purchased for implementation.
2. **It still has limitations**—Not every application or server is going to work within an environment of virtualization. That means an individual or corporation may require a hybrid system to function properly. This still saves time and money in the long run, but since not every vendor supports virtualization and some may stop supporting it after initially starting it, there is always a level of uncertainty when fully implementing this type of system.
3. **It creates a security risk**—Information is our modern currency. If you have it, you can make money. If you don’t have it, you’ll be ignored. Because data is crucial to the success of a business, it is targeted frequently. The average cost of a data security breach in 2017, according to a report published by the Ponemon Institute, was $3.62 million. For perspective: the chances of being struck by lightning are about 1 in a million. The chances of experiencing a data breach while using virtualization? 1 in 4.
4. **It creates an availability issu**e—The primary concern that many have with virtualization is what will happen to their work should their assets not be available. If an organization cannot connect to their data for an extended period of time, they will struggle to compete in their industry. And, since availability is controlled by third-party providers, the ability to stay connected in not in one’s control with virtualization.
5. **It creates a scalability issue**—Although you can grow a business or opportunity quickly because of virtualization, you may not be able to become as large as you’d like. You may also be required to be larger than you want to be when first starting out. Because many entities share the same resources, growth creates lag within a virtualization network. One large presence can take resources away from several smaller businesses and there would be nothing anyone could do about it.
6. **It requires several links in a chain that must work together cohesively**—If you have local equipment, then you are in full control of what you can do. With virtualization, you lose that control because several links must work together to perform the same task. Let’s using the example of saving a document file. With a local storage device, like a flash drive or HDD, you can save the file immediately and even create a backup. Using virtualization, your ISP connection would need to be valid. Your LAN or Wi-Fi would need to be working. Your online storage option would need to be available. If any of those are not working, then you’re not saving that file.
7. **It takes time**—Although you save time during the implementation phases of virtualization, it costs users time over the long-run when compared to local systems. That is because there are extra steps that must be followed to generate the desired result.

## What is x86 hardware virtualization

x86 virtualization is the use of hardware-assisted virtualization capabilities on an x86/x86-64 CPU. In the late 1990s x86 virtualization was achieved by complex software techniques, necessary to compensate for the processor's lack of hardware-assisted virtualization capabilities while attaining reasonable performance.

## The 7 Types of Virtualizations

1. OS Virtualization—aka Virtual Machines
2. Application-Server Virtualization
3. Application Virtualization
4. Administrative Virtualization
5. Network Virtualization
6. Hardware Virtualization
7. Storage Virtualization
   * 1. **OS Virtualization—aka Virtual Machines**

Virtualizing an operating system environment is the most common form of virtualization. It involves putting a second instance or multiple instances of an operating system, like Windows, on a single machine. This empowers businesses to reduce the amount of physical hardware required to run their software by cutting down the number of actual machines. It saves companies cash on energy, cabling, hardware, rack space, and more, while still allowing them to run the same quantity of applications.

* + 1. **Application-Server Virtualization**

Application-server virtualization is another large presence in the virtualization space, and has been around since the inception of the concept. It is often referred to as ‘advanced load balancing,’ as it spreads applications across servers, and servers across applications.

This enables IT departments to balance the workload of specific software in an agile way that doesn’t overload a specific server or underload a specific application in the event of a large project or change. In addition to load balancing it also allows for easier management of servers and applications, since you can manage them as a single instance. Additionally, it gives way to greater network security, as only one server is visible to the public while the rest are hidden behind a reverse proxy network security appliance.

* + 1. **Application Virtualization**

Application virtualization is often confused with application-server virtualization. What it means is that applications operate on computers as if they reside naturally on the hard drive, but instead are running on a server. The ability to use RAM and CPU to run the programs while storing them centrally on a server, like through Microsoft Terminal Services and cloud-based software, improves how software security updates are pushed, and how software is rolled out.

* + 1. **Administrative Virtualization**

Administrative virtualization is one of the least-known forms of virtualization, likely due to the fact that it’s primarily used in data centers. The concept of administration, or ‘management,’ virtualization means segmented admin roles through group and user policies. For example, certain groups may have access to read specific servers, infrastructure, application files, and rules, but not to change them.

* + 1. **Network Virtualization**

Network virtualization involves virtually managing IPs, and is accomplished through tools like routing tables, NICs, switches, and VLAN tags.

* + 1. **Hardware Virtualization**

Hardware virtualization is one of the rarer forms of virtualization, and when simply explained it is similar to OS virtualization (it is, in fact, often required for OS virtualization). Except, instead of putting multiple software instances on a single machine, chunks of a machine are partitioned off to perform specific tasks.

* + 1. **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, and instead function more like worker bees in a hive.

## Virtualization challenges and how businesses can overcome them

1. Resource Distribution

At times, the way virtualization partitions systems can result in some that function really well and others that don't seem to have access to enough resources to meet their needs. Resource distribution challenges often occur early in the transition to virtualization and can be worked out with the provider's help to mitigate these issues moving forward.

1. VM Sprawl

The ability to create as many virtual machines as you want can lead to more VMs than are needed for the company to function. VM sprawl may seem harmless, but it can exacerbate resource distribution problems by diverting resources to VMs that aren't even being used while those that are used and needed see reduced functioning. Companies can avoid VM spawl by sticking to the number of VMs that are actually needed and adding more when the time comes.

1. Backward Compatibility

Many companies use legacy systems that can cause problems with newer virtualized software and programs. Compatibility issues can be time-consuming and difficult to resolve, but vendors may be aware of these difficulties and be able to suggest upgrades or workarounds to make sure everything functions the way it should.

1. Performance Monitoring

Virtualized systems don't lend themselves to the same kind of performance monitoring as hardware like mainframes and hard drives do. New tools like VMmark can create benchmarks that measure performance on virtual networks and make it possible to monitor performance and resource usage.

1. Backup

Since there is no actual hard drive on which data and systems can be backed up, frequent software updates can make it difficult to access backups at times. Software programs like Windows Server Backup and other tools can make this process easier and allow backups to be kept all in one place for easy tracking and access.

1. Security

Virtual systems can be compromised when users don't keep them secure and use best practices for passwords, downloads and other tasks. Security can be a problem for virtualization, but the isolation of each VM by the system can limit security problems and help keep virtual systems more secure than others might be.

1. Licensing Compliance

Using existing licensed software in a virtual environment can lead to compliance issues if more VMs are created than the company is licensed to use the software on. It's important to keep track of how licensed software is being used and to be sure compliance is maintained as the virtual environment grows.

# Application Virtualization

Application virtualization is a software technology that encapsulates computer programs from the underlying operating system on which they are executed.

## Benefits of Application Virtualization

* Allows the running of legacy apps (e.g., those developed for end-of-life OS platforms like Windows 7 and XP).
* Enables cross-platform operations (e.g., running Windows apps on iOS, Android, macOS, and Chrome OS).
* Prevents conflicts with other virtualized apps (e.g., conflicting anti-malware software).
* Permits users to run multiple app instances—if not virtualized, many apps can detect the running of an instance and won't allow new ones.

## Why Application Virtualization

To lower costs and improve productivity, organizations must evolve their digital workspace. This means migrating networking assets from on-premises to the cloud.

## Server without virtualization

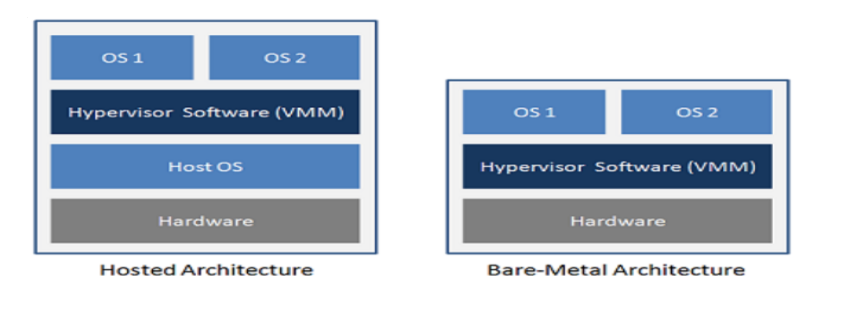
|  | * Only one OS can run at a time within a server. * Under utilization of resources. * Inflexible and costly infrastructure. * Hardware changes require manual effort and access to the physical server. |
| --- | --- |

## Server with Virtualization

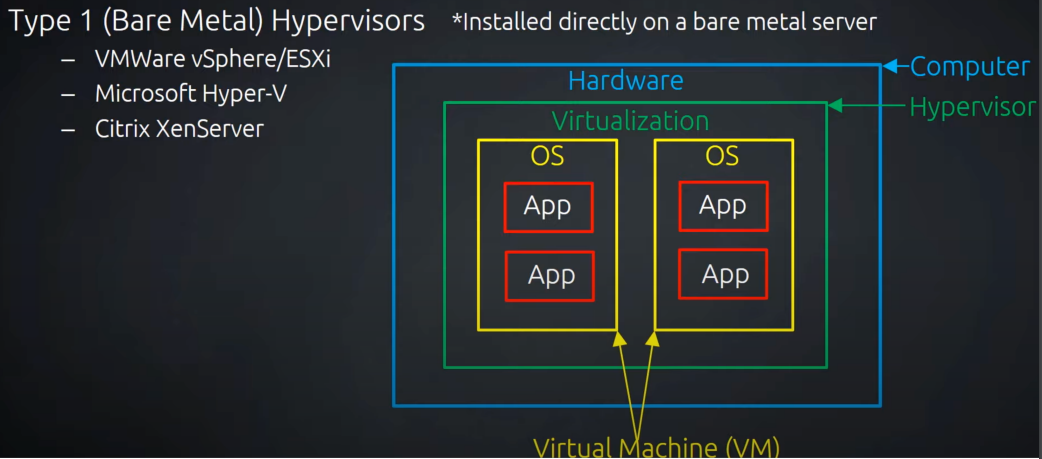
|  | * Can run multiple OS simultaneously. * Each OS can have different hardware * configuration. * Efficient utilization of hardware resources. * Each virtual machine is independent. * Save electricity, initial cost to buy servers, * space etc. * Easy to manage and monitor virtual * machines centrally. |
| --- | --- |

## Types of Virtualizations(Type - 1 & Type - 2)

1. Type -1 Virtualization : Native/Bare Metal Hypervisor
2. Type -2 Virtualization : Hosted Hypervisor



* + 1. **Type 1 Hypervisor**
* This is also known as Bare Metal or Embedded or Native Hypervisor.
* It works directly on the hardware of the host and can monitor operating systems that run above the Hypervisor.
* It is completely independent from the Operating System.
* The hypervisor is small as its main task is sharing and managing hardware resources between different operating systems.
* A major advantage is that any problems in one virtual machine or guest operating system do not affect the other guest operating systems running on the hypervisor.
* Examples: Vmware ESX, XEN,OVMM, Microsoft Hyper-V



* + 1. **Type 2 Hypervisor**
* This is also known as Hosted Hypervisor.
* In this case, the hypervisor is installed on an operating system and then supports other operating systems above it.
* It is completely dependent on host Operating System for its operations
* While having a base operating system allows better specification of policies, any problems in the base operating system affects the entire system as well even if the hypervisor running above the base OS is secure.
* Examples: VMware Workstation, Microsoft Virtual PC Oracle Virtual Box



# Hypervisor

A hypervisor, also known as a virtual machine monitor or VMM, is software that creates and runs virtual machines (VMs). A hypervisor allows one host computer to support multiple guest VMs by virtually sharing its resources, such as memory and processing.

Hypervisor plays an important role in the virtualization scenario by virtualization of hardware. It

provides support for running multiple operating systems concurrently in virtual servers created

within a physical server

The virtualization layer is the software responsible for hosting and managing all VMs. The

virtualization layer is a hypervisor running directly on the hardware.

Example: VMWare, Xen, KVM.

## Hypervisor reduces

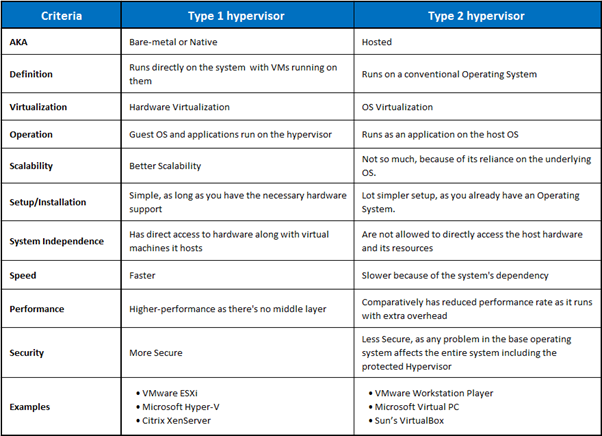
* Space
* Energy
* Maintenance requirements

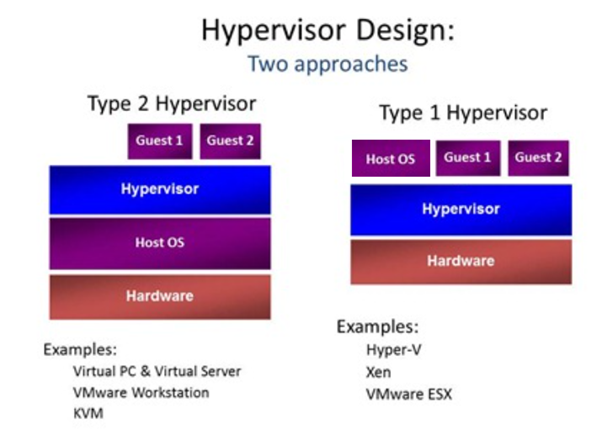
## Benefits of hypervisors

* **Speed:** Hypervisors allow virtual machines to be created instantly, unlike bare-metal servers. This makes it easier to provision resources as needed for dynamic workloads.
* **Efficiency**: Hypervisors that run several virtual machines on one physical machine’s resources also allow for more efficient utilization of one physical server. It is more cost- and energy-efficient to run several virtual machines on one physical machine than to run multiple underutilized physical machines for the same task.
* **Flexibility**: Bare-metal hypervisors allow operating systems and their associated applications to run on a variety of hardware types because the hypervisor separates the OS from the underlying hardware, so the software no longer relies on specific hardware devices or drivers.
* **Portability**: Hypervisors allow multiple operating systems to reside on the same physical server (host machine). Because the virtual machines that the hypervisor runs are independent from the physical machine, they are portable. IT teams can shift workloads and allocate networking, memory, storage and processing resources across multiple servers as needed, moving from machine to machine or platform to platform. When an application needs more processing power, the virtualization software allows it to seamlessly access additional machines.

## What are the goals of hypervisor

A hypervisor, also known as a virtual machine monitor or VMM, is software that creates and runs virtual machines (VMs). A hypervisor allows one host computer to support multiple guest VMs by virtually sharing its resources, such as memory and processing.







# MicroKernel Vs Monolithic Kernel

| **COMPARISON** | **MICROKERNEL** | **MONOLITHIC KERNEL** |
| --- | --- | --- |
| **Basic** | In microkernel user services and kernel, services are kept in separate address space. | In monolithic kernel, both user services and kernel services are kept in the same address space. |
| **Size** | Microkernel are smaller in size. | Monolithic kernel is larger than microkernel. |
| **Execution** | Slow execution. | Fast execution. |
| **Extendible** | The microkernel is easily extendible. | The monolithic kernel is hard to extend. |
| **Security** | If a service crashes, it does effect on working of microkernel. | If a service crashes, the whole system crashes in monolithic kernel. |
| **Code** | To write a microkernel, more code is required. | To write a monolithic kernel, less code is required. |
| **Communication** | It is a single static binary file | Servers communicate through IPC. |
| **Example** | QNX, Symbian, L4Linux, Singularity, K42, Mac OS X, Integrity, PikeOS, HURD, Minix, and Coyotos. | Linux, BSDs (FreeBSD, OpenBSD, NetBSD), Microsoft Windows (95,98,Me), Solaris, OS-9, AIX, HP-UX, DOS, OpenVMS, XTS-400 etc. |

## MONOLITHIC ARCHITECTURE

The monolithic architecture is considered to be a traditional way of building applications. A monolithic application is built as a single and indivisible unit. Usually, such a solution comprises a client-side user interface, a server side-application, and a database. It is unified and all the functions are managed and served in one place.

* + 1. **Advantages of the Monolithic Architecture**
* Less cross-cutting concerns. Cross-cutting concerns are the concerns that affect the whole application such as logging, handling, caching, and performance monitoring. In a monolithic application, this area of functionality concerns only one application so it is easier to handle it.
* Easier debugging and testing. In contrast to the microservices architecture, monolithic applications are much easier to debug and test. Since a monolithic app is a single indivisible unit, you can run end-to-end testing much faster.
* Simple to deploy. Another advantage associated with the simplicity of monolithic apps is easier deployment. When it comes to monolithic applications, you do not have to handle many deployments – just one file or directory.
* Simple to develop. As long as the monolithic approach is a standard way of building applications, any engineering team has the right knowledge and capabilities to develop a monolithic application.
  + 1. **Disadvantages of the Monolithic Architecture**
* Understanding. When a monolithic application scales up, it becomes too complicated to understand. Also, a complex system of code within one application is hard to manage.
* Making changes. It is harder to implement changes in such a large and complex application with highly tight coupling. Any code change affects the whole system so it has to be thoroughly coordinated. This makes the overall development process much longer.
* Scalability. You cannot scale components independently, only the whole application.
* New technology barriers. It is extremely problematic to apply a new technology in a monolithic application because then the entire application has to be rewritten.

## Microkernel

Microkernel being a kernel manages all system resources. But in a microkernel, the user services and the kernel services are implemented in different address space. The user services are kept in user address space, and kernel services are kept under kernel address space. This reduces the size of the kernel and further reduces the size of operating system.

* + 1. **Advantages of the Microkernel**
* Microkernel architecture is small and isolated therefore it can function better.
* Microkernels are secure because only those components are included that disrupt the functionality of the system otherwise.
* The expansion of the system is more accessible, so it can be added to the system application without disturbing the Kernel.
* Microkernels are modular, and the different modules can be replaced, reloaded, modified without even touching the Kernel.
* Fewer system crashes when compared with monolithic systems.
* Microkernel interface helps you to enforce a more modular system structure.
* Without recompiling, add new features
* Server malfunction is also isolated as any other user program’s malfunction.
* Microkernel system is flexible, so different strategies and APIs, implemented by different servers, which can coexist in the system.
* Increased security and stability will result in a decreased amount of code which runs on kernel mode
  + 1. **Disadvantages of the Microkernel**
* Providing services in a microkernel system are expensive compared to the normal monolithic system.
* Context switch or a function call needed when the drivers are implemented as procedures or processes, respectively.
* The performance of a microkernel system can be indifferent and may lead to some problems.

# Container vs hypervisor(VM)

Both involved in making applications faster and more efficient, but they achieve this in different ways.

## Hypervisors:

* Allow an operating system to run independently from the underlying hardware through the use of virtual machines.
* Share virtual computing, storage and memory resources.
* Can run multiple operating systems on top of one server (bare-metal hypervisor) or installed on top of one standard operating system and isolated from it (hosted hypervisor).

## Containers:

* Allow applications to run independently of an operating system.
* Can run on any operating system—all they need is a container engine to run.
* Are extremely portable since in a container, an application has everything it needs to run.

# Docker

Docker is a set of platform as a service products that use OS-level virtualization to deliver software in packages called containers.

## Some Docker vocabulary

1. Docker Image

The basis of a Docker container. Represents a full application

1. Docker Container

The basis of a Docker container. Represents a full application

1. Docker Engine

Creates, ships and runs Docker containers deployable on a physical or

virtual, host locally, in a datacenter or cloud service provide

1. Registry Service (Docker Hub(Public) or Docker Trusted Registry(Prívate))

Cloud or server based storage and distribution service for your images

## Advantages of Docker:

1. Return on Investment and Cost Savings

Dockers first advantage is ROI. Especially for large, established companies, which need to generate steady revenue over the long term, the solution is only better if it can drive down costs while raising profits.

1. Rapid Deployment

It can decrease deployment to seconds. It is because of the fact that it can create a container for every process and even does not boot an OS. So, even without worrying about the cost to bring it up again, it would be higher than what is affordable, Data can be created as well as destroyed.

1. Security

Docker makes sure that applications that are running on containers are completely segregated and isolated from each other, from a security point of view, by granting us complete control over traffic flow and management.

1. Simplicity and Faster Configurations

The way Docker simplifies the matters is one of the key benefits of it. It gives flexibility to users to take their own configuration, put that into the code, and further deploy it without any problems.

However, the requirements of the infrastructure are no longer linked with the environment of the application, as Docker can be used in a wide variety of environments.

1. CI Efficiency

With the help of a Docker, we can build a container image and can further use that same image over every step of the deployment process.

The advantage of it is the ability to separate non-dependent steps and also run them in parallel. In addition, the duration of time it takes from build to production may speed up notably.

1. Continuous Integration

While it comes to Continuous Integration, Docker works well as part of its pipelines along with tools such as Travis, Jenkins, and Wercker.

## Disadvantages of Docker:

1. Missing features

There are a ton of feature requests that are under progress, like container self-registration, and self-inspections, copying files from the host to the container, and many more.

1. Data in the container

There are times when a container goes down, so after that, it needs a backup and recovery strategy, although we have several solutions for that they are not automated or not very scalable yet.

1. Run applications as fast as a bare-metal serve

In comparison with the virtual machines, Docker containers have less overhead but not zero overhead. If we run an application directly on a bare-metal server we get true bare-metal speed even without using containers or virtual machines. However, Containers don’t run at bare-metal speeds.

1. Provide cross-platform compatibility

The one major issue is if an application is designed to run in a Docker container on Windows, then it can’t run on Linux or vice versa. However, Virtual machines are not subject to this limitation.

So, this limitation makes Docker less attractive in some highly heterogeneous environments which are composed of both Windows and Linux servers.

1. Run applications with graphical interfaces

In general, Docker is designed for hosting applications which run on the command line. Though we have a few ways (like X11 forwarding) by which we can make it possible to run a graphical interface inside a Docker container, however, this is clunky.

Hence we can say, for applications that require rich interfaces, Docker is not a good solution.

1. Solve all your security problems

In simple words, we need to evaluate the Docker-specific security risks and make sure we can handle them before moving workloads to Docker.

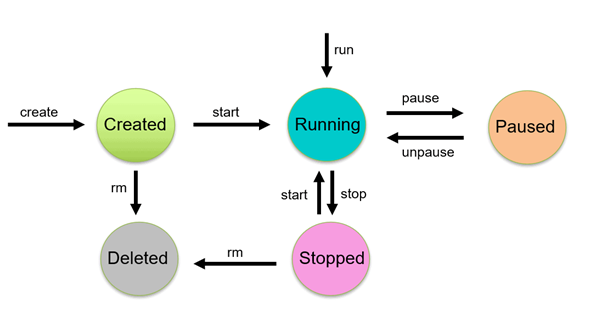
The reason behind it is that Docker creates new security challenges like the difficulty of monitoring multiple moving pieces within a large-scale, dynamic Docker environment.



## Docker Container Lifecycle:

There are different stages when we create a Docker container which is known as Docker Container Lifecycle:

* Created: A container that has been created but not started
* Running: A container running with all its processes
* Paused: A container whose processes have been paused
* Stopped: A container whose processes have been stopped
* Deleted: A container in a dead state
* Docker Container lifecycle



Start Container: $ docker start <container name>

Run Container: $ docker run -it --name <container name> <image name>

Pause Container: $ docker pause <container name>

Unpause Container: $ docker unpause <container name>

Stop Container: $ docker stop <container name>

Delete Container: $ docker stop <container name> $ docker rm <container name>

Kill Container: docker kill <container name>

## Dockerfile:

Docker can build images automatically by reading the instructions from a Dockerfile. A Dockerfile is a text document that contains all the commands a user could call on the command line to assemble an image. Using docker build users can create an automated build that executes several command-line instructions in succession.

# Container

Containers are isolated from one another and bundle their own software, libraries and configuration files; they can communicate with each other through well-defined channels.

## Advantages of Container:

1. Portability

As a part of a distributed system, containers are highly portable.

Because containers pack microservices and their dependencies in a small-sized package, it’s easy to move containers around, even across environments, such as the public cloud, private cloud, and hybrid cloud, as well as the multi-cloud and bare-metal environments.

1. Effective resource usage

Code packaged within containers share most of the dependencies needed to run the containers, including an operating system, libraries, and frameworks.

Unlike in virtual machines, there’s only one copy of necessary files in each hardware, leading to more effective resource usage. This also results in a lighter container, which means you can fit more containers within a physical server.

1. Easier to maintain

As containers use a microservices-based architecture, your code is broken down into manageable pieces that can be handled individually. Hence, you can update and maintain a container without worrying it will affect other parts of your application.

1. Highly Scalable

Container orchestration platforms are created to help you manage your containers. Container orchestrators, like Kubernetes or Docker Swarm, automate most of your container management process, including scaling, networking, and deployment.

Containerized applications are highly scalable, and they can scale up and down quickly by spinning up new nodes and/or pods when the needs call for it.

## Disadvantages of Container:

1. Lacking Security measures

Containers provide lightweight isolation from the host OS and containers within the same system. This leads to a weaker security boundary compared to virtual machines.

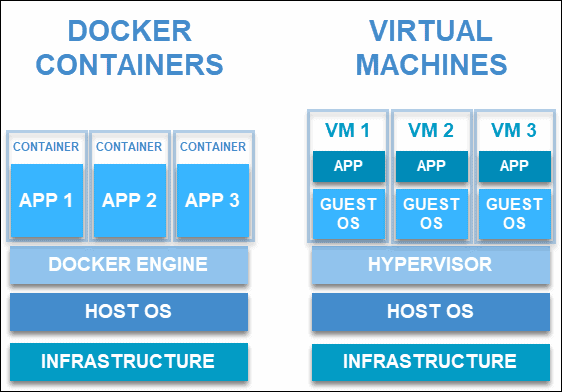
However, mature container users are paying more attention to security, as they try to improve collaboration between DevOps and Security, according to StackRox.

1. Runs only one OS

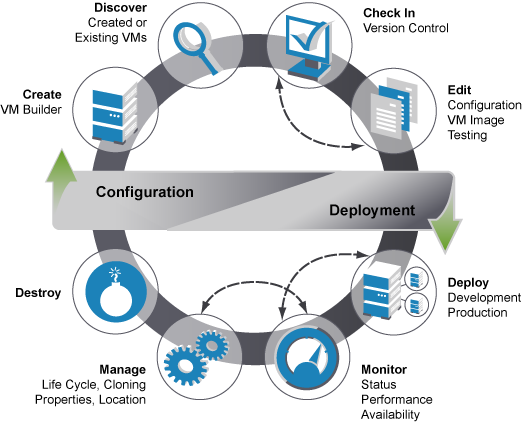
This can be a benefit if you only use one OS, but if you need to be able to use it across different OS’s this is a negative. You can run an earlier version of the same OS using lightweight virtual machines.

## Popular container providers:

* RedHat Openshift
* Google Kubernete Engine
* Docker
* Linux Containers (LXC, LXD, CGManager)
* Mesos
* Windows Server Containers



# VirtualMachineLifeCycle



Docker is a set of platform as a service products that use OS-level virtualization to deliver software in packages called containers.

# Kubernetes

Kubernetes is an open-source container-orchestration system for automating computer application deployment, scaling, and management. It was originally designed by Google and is now maintained by the Cloud Native Computing Foundation.

# Linux Containers - LXC and LXD

## LXC:

LXC—short for “Linux containers”, is a solution for virtualizing software at the operating system level within the Linux kernel. Unlike traditional hypervisors (think VMware, KVM and Hyper-V), LXC lets you run single applications in virtual environments, although you can also virtualize an entire operating system inside an LXC container

## LXD:

The simplest way to define LXD is to say it’s an extension of LXC. LXD also happens to be LXC’s main claim to fame, now that LXC has ceased to be important for Docker and CoreOS.

The more technical way to define LXD is to describe it as a REST API that connects to libxlc, the LXC software library. LXD, which is written in Go, creates a system daemon that apps can access locally using a Unix socket, or over the network via HTTPS.

# Cloud orchestration

Cloud orchestration is the end-to-end automation of the deployment of services in a cloud environment. More specifically, it is the automated arrangement, coordination, and management of complex computer systems, middleware, and services—all of which helps to accelerate the delivery of IT services while reducing costs. It is used to manage cloud infrastructure, which supplies and assigns required cloud resources to the customer like the creation of VMs, allocation of storage capacity, management of network resources, and granting access to cloud software. By using the appropriate orchestration mechanisms, users can deploy and start using services on servers or on any cloud platforms.

## Three aspects to cloud orchestration:

* Resource orchestration- where resources are allocated
* Workload orchestration- where workloads are shared between the resources
* Service orchestration- where services are deployed on servers or cloud environments

## Orchestration tools:

* Chef and Puppet
* OpenStack
* Heat
* Heat Orchestration Templates (HOT)
* Juju

## OpenStack components:

OpenStack is based on a modular architecture. There are six primary OpenStack components which handle compute, network and storage functions for on-demand VM provisioning. A bunch of other components enable additional features, such as dashboarding, bare metal provisioning, containers, secrets management and telemetry. In order to handle this complexity, organisations often use OpenStack Charms for fully automated OpenStack installation and post-deployment operations.

* Nova

Nova is the primary compute engine of OpenStack, responsible for instance scheduling, creation and termination. In order to ensure widespread interoperability, Nova supports a wide range of hypervisors, including QEMU/KVM, Hyper-V, VMware ESXi and Xen.

* Glance

Glance is an image service, responsible for uploading, managing and retrieving cloud images for instances running on OpenStack. Glance works across a variety of stores to provide the most convenient location of images for organisations.

* Neutron

Neutron provides network connectivity between OpenStack instances, enabling multi-VM deployments. For this purpose, Neutron uses various software defined networking (SDN) technologies, including Open Virtual Network (OVN), Open vSwitch (OVS), Juniper Contrail, Cisco ACI, etc.

* Cider

Cinder is a storage component that is responsible for provisioning, management and termination of persistent block devices. Those can be later attached to the instances running on OpenStack to enable persistent block storage.

* Swift

Swift is another storage component that provides a highly available and scalable object storage service similar to Amazon S3. It enables storing and retrieving unstructured data objects using a RESTful API for both OpenStack services and instances running on the cloud.

* Keystone

Keystone serves as an identity service, providing authentication and authorization functions for the users in order to enable multi-tenancy. Keystone can be easily integrated with external identity systems, such as lightweight directory access protocol (LDAP) or Active Directory.

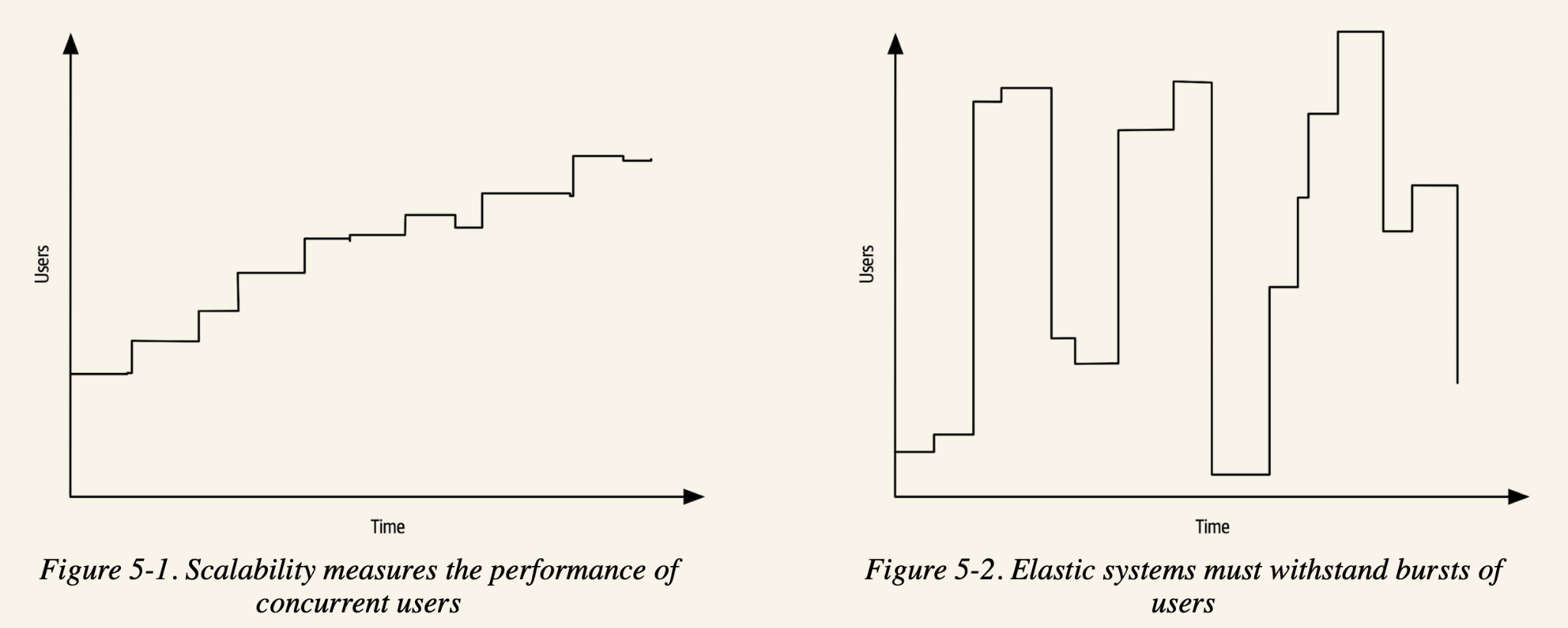
# Q n A

## 

**Q.1 (a) What is the difference between scalability and elasticity? Support your answer with a figure.**

Ans.

| **Scalability** | **Elasticity** |
| --- | --- |
| Increasing the capacity to meet the increasing workload | “Increasing or reducing” the capacity to meet the increasing or reducing workload |
| In a scaling environment, the available resources may exceed to meet the future demands | In elasticity environment the available resources matches the current demands |
| Scalability adapts only to the workload increase by provisioning the resources in an incremental manner | it adopts to both the workload increase & workload decrease in an automatic manner |
| Scalability enables a corporate to meet expected demands for services with long term strategic needs | Elasticity enables a corporate to meet the unexpected changes in the demand for service with “short-term”, tactical needs |
| Scalability is used to meet the static increase in the workload. | Elasticity is used just to meet the sudden up and down in the workload for a small period of time. |
| Scalability is always used to address the increase in workload in an organization. | Elasticity is used to meet dynamic changes, where the resources need can increase or decrease. |
| Scalability is used by giant companies whose customer circle persistently grows in order to do the operations efficiently. | Elasticity is commonly used by small companies whose workload and demand increases only for a specific period of time. |
| Scalability is a long term planning and adopted just to deal with an expected increase in demand. | It is a short term planning and adopted just to deal with an unexpected increase in demand or seasonal demands. |



**Q.1 (b) What are the disadvantages of virtualization in the context of cloud computing?**

Ans. The Disadvantages of Virtualization

**1. It can have a high cost of implementation.**

The cost for the average individual or business when virtualization is being considered will be quite low. For the providers of a virtualization environment, however, the implementation costs can be quite high. Hardware and software are required at some point and that means devices must either be developed, manufactured, or purchased for implementation.

**2. It still has limitations.**

Not every application or server is going to work within an environment of virtualization. That means an individual or corporation may require a hybrid system to function properly. This still saves time and money in the long run, but since not every vendor supports virtualization and some may stop supporting it after initially starting it, there is always a level of uncertainty when fully implementing this type of system.

**3. It creates a security risk.**

Information is our modern currency. If you have it, you can make money. If you don’t have it, you’ll be ignored. Because data is crucial to the success of a business, it is targeted frequently. The average cost of a data security breach in 2017, according to a report published by the Ponemon Institute, was $3.62 million. For perspective: the chances of being struck by lightning are about 1 in a million. The chances of experiencing a data breach while using virtualization? 1 in 4.

**4. It creates an availability issue.**

The primary concern that many have with virtualization is what will happen to their work should their assets not be available. If an organization cannot connect to their data for an extended period of time, they will struggle to compete in their industry. And, since availability is controlled by third-party providers, the ability to stay connected in not in one’s control with virtualization.

**5. It creates a scalability issue.**

Although you can grow a business or opportunity quickly because of virtualization, you may not be able to become as large as you’d like. You may also be required to be larger than you want to be when first starting out. Because many entities share the same resources, growth creates lag within a virtualization network. One large presence can take resources away from several smaller businesses and there would be nothing anyone could do about it.

**6. It requires several links in a chain that must work together cohesively.**

If you have local equipment, then you are in full control of what you can do. With virtualization, you lose that control because several links must work together to perform the same task. Let’s use the example of saving a document file. With a local storage device, like a flash drive or HDD, you can save the file immediately and even create a backup. Using virtualization, your ISP connection would need to be valid. Your LAN or Wi-Fi would need to be working. Your online storage option would need to be available. If any of those are not working, then you’re not saving that file.

**7. It takes time.**

Although you save time during the implementation phases of virtualization, it costs users time over the long-run when compared to local systems. That is because there are extra steps that must be followed to generate the desired result.

**Q.1 (c) What do you mean by Cloud Burst?**

Ans. In cloud computing, cloud bursting is a configuration which is set up between a private cloud and a public cloud to deal with peaks in IT demand. If an organisation using a private cloud reaches 100 percent of its resource capacity, the overflow traffic is directed to a public cloud so there is no interruption of services.

Cloud bursting is an application deployment technique in which an application runs in a private cloud or data center and bursts into a public cloud when the demand for computing capacity spikes. Once reduced to normal traffic levels, data is moved back to the private cloud.

In addition to flexibility and self-service functionality, the key advantage to cloud bursting is economical savings. You only pay for the additional resources when there is a demand for those resources - no more spending on extra capacity you are not using or trying to predict demand peaks and fluctuations. An application can be applied to the private cloud, then burst to the public cloud only when necessary to meet peak demands. Plus, cloud bursting can also be used to shoulder processing burdens by moving basic applications to the public cloud to free up local resources for business-critical applications. When using cloud bursting, you should consider security and compliance requirements, latency, load balancing and platform compatibility.

**Q.1 (d) \_\_\_\_\_\_\_\_\_\_ is DNS service in Amazon Cloud Computing. [2 + 2 + 1 + 1 = 6]**

Ans. Amazon Route 53 is DNS service in Amazon Cloud Computing.

Amazon Route 53 is a highly available and scalable cloud Domain Name System (DNS) web service. It is designed to give developers and businesses an extremely reliable and cost effective way to route end users to Internet applications by translating names like www.example.com into the numeric IP addresses like 192.0.2.1 that computers use to connect to each other. Amazon Route 53 is fully compliant with IPv6 as well.

Amazon Route 53 effectively connects user requests to infrastructure running in AWS – such as Amazon EC2 instances, Elastic Load Balancing load balancers, or Amazon S3 buckets – and can also be used to route users to infrastructure outside of AWS. You can use Amazon Route 53 to configure DNS health checks, then continuously monitor your applications’ ability to recover from failures and control application recovery with Route 53 Application Recovery Controller.

Amazon Route 53 Traffic Flow makes it easy for you to manage traffic globally through a variety of routing types, including Latency Based Routing, Geo DNS, Geo-Proximity, and Weighted Round Robin—all of which can be combined with DNS Failover in order to enable a variety of low-latency, fault-tolerant architectures. Using Amazon Route 53 Traffic Flow’s simple visual editor, you can easily manage how your end-users are routed to your application’s endpoints—whether in a single AWS region or distributed around the globe. Amazon Route 53 also offers Domain Name Registration – you can purchase and manage domain names such as example.com and Amazon Route 53 will automatically configure DNS settings for your domains.

Benefits:

* Highly available and reliable:

Amazon Route 53 is backed by the Amazon Route 53 Service Level Agreement.

* Flexible:

easily roll back to a previous version using the console or API

* Designed for use with other Amazon Web Services:

You can use Amazon Route 53 to map your zone apex (example.com versus www.example.com) to your Elastic Load Balancing instance, Amazon CloudFront distribution, AWS Elastic Beanstalk environment, API Gateway, VPC endpoint, or Amazon S3 website bucket using a feature called Alias record.

* Simple:

can configure your DNS settings with the AWS Management Console or our easy-to-use API.

* Fast:

It runs your application in multiple locations around the world and uses traffic policies to ensure your end users are routed to the closest healthy endpoint for your application.

* Cost-effective:

pay only for the resources you use

* Secure:

By integrating Amazon Route 53 with AWS Identity and Access Management (IAM), you can grant unique credentials and manage permissions for every user within your AWS account and specify who has access to which parts of the Amazon Route 53 service.

* Scalable:

Route 53 is designed to automatically scale to handle very large query volumes without any intervention from you.

* Simplify the hybrid cloud

Amazon Route 53 Resolver provides recursive DNS for your Amazon VPC and on-premises networks over AWS Direct Connect or AWS Managed VPN.

**Q.2 (a) An enterprise needs highly controlled storage and access to their databases as well as**

**managing the infrastructure for web frontends and other applications. They have a large**

**existing IT infrastructure and they are continually expanding the capabilities. Which**

**cloud computing model will satisfy all their current needs and enable them to reduce**

**cost? Justify your answer.**

Ans. The best cloud computing model for this scenario is IaaS (Infrastructure-as-a-service)

The reason for this decision is because of the following reasons:

* IaaS gives complete access and control to the cloud infrastructure.
* Here, the enterprise is continuously expanding capabilities in their infrastructure. Which means all their custom components in their architecture needs to be deployed. In cases of SaaS and PaaS, we need to depend on the cloud provider/vendor to introduce new features that we can leverage. Since all capabilities here seem to be custom, IaaS is the way to go
* IaaS is always cheaper than PaaS and SaaS, since the latter have a lot of out of the box features which will be charged, that we may not use. In the case of IaaS, it's controlled, so we don’t usually deploy anything that we don't need and are not charged extra.

**Q.2 (b) State True/ False and Justify your answer for the following statement. “EBS storage is persistent even if an instance is terminated.”**

Ans **“EBS storage is persistent even if an instance is terminated.” => False**

.

When an instance terminates, the data on any instance store volumes associated with that instance is deleted.

By default, Amazon EBS root device volumes are automatically deleted when the instance terminates. However, by default, any additional EBS volumes that we attach at launch, or any EBS volumes that you attach to an existing instance persist even after the instance terminates. This behavior is controlled by the volume's DeleteOnTermination attribute, which we can modify.

**Q.2 (c) How Keystone Play an Important Role in Openstack architecture?**

Ans. Keystone serves as an identity service, providing authentication and authorization functions for the users in order to enable multi-tenancy. Keystone can be easily integrated with external identity systems, such as lightweight directory access protocol (LDAP) or Active Directory.

In OpenStack, the keystone can play the role of a Certificate Authority (CA) using the keystone-manage utility or done by a third party.

The keystone module of open stack provides another type of identity services that include identification, cataloguing, management of policies, and dealing with tokens for authentication. The identity services offered by the keystone module provided as a set of services situated at more than one endpoint

**Q.2 (d) Justify how elastic IP is important for high availability of IaaS services. [2 + 2 + 5 + 3 = 12]**

Ans.

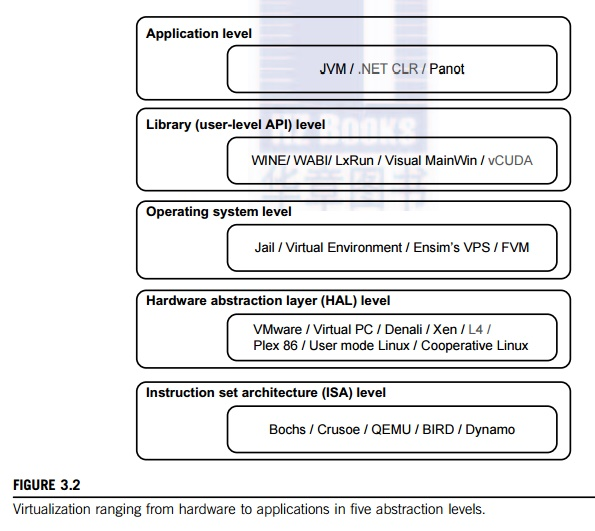
High availability handles Single point of failure and a reliable crossover of VMs across different availability zones. A typical architecture for High availability requires load balancers. Load balancers are important for the crossover between availability zones as well as distribution of traffic amongst HA enabled VMs and this is referred to as dynamic cloud computing.

An Elastic IP address is a static IPv4 address designed for dynamic cloud computing. With an Elastic IP address, you can mask the failure of an instance or software by rapidly remapping the address to another instance in your account, which can be tied to the load balancer.

**Q.3 (a) What is the performance of Virtualization at the ISA level? Justify your answer.**

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

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



**Q.3 (b) Differentiate between EBS and S3 Services.**

Ans.

1. **Amazon Elastic Block Store (Amazon EBS)**

Use Amazon EBS to provide storage for your virtual machine drives. It stores data in equally-sized blocks and organizes them into a hierarchy similar to a traditional file system. The volumes are provisioned in size and attached toEC2 instances in a way that’s similar to the local disk drive on a physical machine

Amazon EBS Benefits:

* Performance optimization: Increase throughput by devoting network capacity and minimizing the network contention between your instances and EBS.
* Low-latency performance: By using SSD EBS volumes, it offers reliable I/O performance scaled tomeet your workload needs. If your application requires high performance but nota large amount of storage, you can provision performance separately from storage capacity.
* Highly available and secure storage: EBS volumes offer redundancy within its Availability Zones while access control and encryption bolster security.
* Geographic interchangeability: With EBS, you can duplicate snapshots throughout AWS regions and place resources and data in multiple locations. This makes disaster recovery, data center migration, and geographical expansion simple.
* Easy data backup and restoration: Point-in-time volume snapshots safeguard data.
* Rapid up- or down-scaling: EBS can quickly scale volumes, ensuring you get the right performance and capacity for changing computing needs.
* Potential cost savings: Although you must pre-allocate your entire disk upfront, if you know how much space you need and only access your storage from one EC2 instance at a time, EBS costs about a third as much as EFS per GB.

Amazon EBS Use Cases:

* Testing and development: You can scale, archive, duplicate, or provision your testing, development, or production environments.
* NoSQL databases: EBS offers NoSQL databases the low-latency performance and dependability they need for peak performance.
* Relational databases: EBS scales to meet your changing storage needs. This makes it a great choice for deploying databases, including PostgreSQL, MySQL, Oracle, or Microsoft SQL Server.
* Business consistency: Copy EBS Snapshots and Amazon Machine Images (AMIs) to run applications in different AWS regions. This reduces data loss and speeds recovery time by backing up log files and data regularly, across geographies.
* Enterprise-wide applications: It can meet a variety of enterprise computing needs through powerful block storage that can support your most important applications, such as Microsoft Exchange, Oracle, or Microsoft SharePoint.

1. **Amazon Elastic File System (Amazon EFS)**

EFS is the best choice for running any application that has a high workload, requires scalable storage, and must produce output quickly. It scales automatically, even to meet the most abrupt workload spikes. After the period of high-volume storage demand has passed, EFS will automatically scale back down. EFS can be mounted to different AWS services and accessed from all your virtual machines. Use it for running shared volumes, or for big data analysis. You’ll always pay for the storage you actually use, rather than provisioning storage in advance that’s potentially wasted.

Amazon EFS Benefits:

* Performance that scales to support any workload: EFS offers the throughput changing workloads need. It can provide higher throughput in spurts that match sudden file system growth, even for workloads up to 500,000 IOPS or 10 GB per second.
* Energetic elasticity: Automatically scale your file system storage up or down. Remove or add files and never disturb applications. Once you make your EFS file system you can add files without worrying about storage provisioning.
* Accessible file storage: On-premises servers and EC2 instances can access shared file systems concurrently. EC2 instances can also access EFS file systems located in other AWS regions through VPC peering.
* Serverless architecture support:Unlike EBS, EFS works with AWS Lambda serverless functions to easily share data from function to function.Lambda functions can read large files from EFS, such as code libraries, and write output to EFS for storage and sharing.
* Comprehensive managed service: EFS is a complete managed service, meaning your firm will never have to patch, deploy, or maintain your file system.
* Cost savings: The only storage you’ll pay for is exactly what you use, as there’s no advance provisioning, up-front fees, or commitments. Moreover, you can use Lifecycle Management to transfer files that have been unused for a month to a more cost-effective storage class, which can lower expenses up to 85 percent. However, keep in mind that EFS costs about three times as much per GB as EBS.
* Tighter security and compliance: You can securely access the file system with your current security solution, or control access to EFS file systems using AWS Identity and Access Management (IAM), Amazon Virtual Private Cloud (Amazon VPC), or POSIX permissions. And, EFS can encrypt your data, whether it’s in transit or at rest. This gives you dependable security and makes regulatory compliance easier.

Amazon EFS Use Cases:

* Lift-and-shift application support: EFS is elastic, available, and scalable. It enables you to move enterprise applications easily and quickly without needing to re-architect them.
* Analytics for big data: It has the ability to run big data applications, which demand significant node throughput, low-latency file access, and read-after-write operations.
* Content management system and web server support: EFS is a robust throughput file system capable of enabling content management systems and web serving applications, such as archives, websites, or blogs.
* Application development and testing: Only EFS provides a shared file system needed to share code and files across multiple compute resources to facilitate auto-scaling workloads.

1. **Amazon S3**

Amazon S3 provides object storage. Each object has its own unique identifier or key, for access through web requests from any location. S3 also supports static web content hosting that can be accessed from the S3 bucket or from AWS CloudFront. And S3 is notably secure.

Amazon S3 Benefits:

* Robust performance, scalability, and availability: Amazon S3 scales storage resources free from resource procurement cycles or investments upfront.It keeps your data safe from errors, failures, and threats. It makes it available when you need it.
* Cost-saving storage classes: Storing data throughout S3 storage classes saves costs and maintains high levels of performance. Storage class analysis enables you to locate data that may be moved to a lower-cost storage class. Then you can make the transfer using an S3 Lifecycle policy. Finally, S3 Intelligent-Tiering enables the storage of data that has changing or unknown access patterns by tiering objects, which cuts storage costs.
* Easier security, compliance, and audit features: S3 can store data and protect it from unauthorized access using its powerful access management and encryption tools. S3 has features that make it easy to comply with regulatory requirements, and Amazon Macie can deny irregular access requests to your sensitive data. Plus, S3 works well with AWS’ many auditing features.
* Exacting data control: An array of management tools enables you to classify and report on data. S3 has storage class analysis that monitors access patterns, while S3 Lifecycle analyzes object transfers to lower-cost storage. S3 Object Lock assigns retention dates to objects to prevent deletion, and S3 Inventory offers visibility of stored objects and their encryption and metadata. Finally, S3 Batch Operations can run storage management maintenance for billions of objects while AWS Lambda can be used to automate workflows, define alerts and log activities without added management of infrastructure.

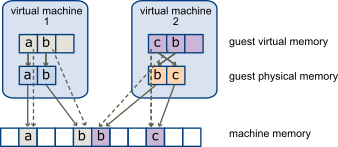
Amazon S3 Use Cases

* Data lake and big data analytics: S3 can create a data lake to hold raw data in its native format, then use machine learning tools, query-in-place, and analytics to draw insights. S3 works with AWS Lake Formation to create data lakes, then define governance, security, and auditing policies. Together, they can be scaled to meet your growing data stores, and you’ll never have to make an investment upfront.
* Backup and restoration: Secure, robust backup and restoration solutions are easy to build when you combine S3 with other AWS offerings, including EBS, EFS, or S3 Glacier. These offerings enhance your on-premises capabilities, while other offerings can help you meet compliance, recovery time, and recovery point objectives.
* Reliable disaster recovery: S3 storage, S3 Cross-Region Replication and additional AWS networking, computing, and database services make it easy to protect critical applications, data, and IT systems. It offers nimble recovery from outages, no matter if they are caused by system failures, natural disasters, or human error.
* Methodical archiving: S3 works seamlessly with other AWS offerings to provide methodical archiving capabilities. S3 Glacier and S3 Glacier Deep Archive enable you to archive data and retire physical infrastructure. There are three S3 storage classes you can use to retain objects for extended periods of time at their lowest rates. S3 Lifecycle policies can be created to archive objects at any point within their lifecycle, or you can upload objects to archival storage classes directly. S3 Object Lock meets compliance regulations by applying retention dates objects to avoid their deletion. And unlike a tape library, S3 Glacier can restore any archived object within minutes.

**Q.3 (c) In memory virtualization of cloud computing environment, shadow page tables are maintained by \_\_\_\_\_\_\_\_\_\_\_\_.**

Ans. In memory virtualization of cloud computing environment, shadow page tables are maintained by VMM

* The boxes represent pages, and the arrows show the different memory mappings.
* The arrows from guest virtual memory to guest physical memory show the mapping maintained by the page tables in the guest operating system. (The mapping from virtual memory to linear memory for x86-architecture processors is not shown.)
* The arrows from guest physical memory to machine memory show the mapping maintained by the VMM.
* The dashed arrows show the mapping from guest virtual memory to machine memory in the shadow page tables also maintained by the VMM. The underlying processor running the virtual machine uses the shadow page table mappings.



**Q.3 (d) Give at least two commercially available service providers that allow users to rent the computers to run and deploy their applications. [2 + 2 + 1 + 2 = 7]**

Ans. The scenario talks about “renting computers, to run and deploy applications”. The statement does not say how much control is required and just says “rent computers”. Hence, Assuming the statement talks about IaaS. The most relied upon and popular to cloud providers are:

1. Amazon Web Services (AWS)
2. Microsoft Azure

Though the above two providers are the leaders, there are others which are also popular like Digital Ocean

---------------------------------------------------------------------------------------------------------------------

* Salesforce.com
* Microsoft Office 365
* Box
* Google Apps
* Amazon Web Services
* Concur
* Zendesk
* DocuSign
* Dropbox
* Slack

1. Salesforce.com

Arguably the quintessential Software as a Service application, Salesforce remains at the vanguard of the cloud computing revolution it helped create. The customer relations management solution enables businesses to collect all information on customers, prospects and leads within a single online platform, enabling authorized employees to access critical data on any connected device at any time. Salesforce credits its tools for boosting customer sales an average of 37 percent as well as driving increased client loyalty and satisfaction.

2. Microsoft Office 365

Signature Microsoft productivity applications such as Word, Excel and PowerPoint are longtime staples of the workplace, but the cloud-based Microsoft Office 365 dramatically expands the Office suite’s parameters. Users now may create, edit and share content from any PC, Mac, iOS, Android or Windows device in real-time, connect with colleagues and customers across a range of tools from email to video conferencing and leverage a range of collaborative technologies supporting secure interactions both inside and outside of the organization.

3. Box

This online workspace enables professionals to collaborate with anyone, anywhere. Users can securely share large files via traditional link or custom URL, safeguarding data and documents via permissions and password protection. Box supports more than 120 file types, and users may preview content prior to downloading. All content sharing, editing, discussion and approval is confined to one centralized file, and users receive real-time notifications when edits are made. Box also automates tasks such as employee onboarding and contract approvals, reducing repetition and abbreviating review cycles.

4. Google Apps

Google long ago expanded beyond its search and advertising roots to offer businesses a comprehensive suite of productivity tools. Google Apps includes custom professional email (complete with spam protection), shared calendars and video meetings alongside Google Drive. A cloud-based document storage solution, Google Drive enables staffers to access files from any device and share them instantly with colleagues, in the process eliminating email attachments as well as the hassles of merging different versions.

5. Amazon Web Services

Amazon, too, has evolved beyond its core e-commerce platform to support the on-demand delivery of cloud-based IT resources and applications, bolstered by pay-as-you-go pricing options. Amazon Web Services currently encompasses more than 70 services in all, including computing, storage, networking, database, analytics, deployment, management and tools for the Internet of Things.

6. Concur

Business travel can pose headaches for on-the-go employees and finance departments alike. Concur streamlines the process by automating travel and expense management. Its web-based and mobile solutions enable staffers to book travel plans according to their own needs and preferences, while also making sure all bookings fall within company spending limits. Concur additionally reconciles expenses after travel is completed and delivers electronic airline, hotel and auto rental receipts directly into digital expense reports. This negates the need to collect, track and submit paper receipts.

7. Zendesk

This cloud-based customer service and support ticketing platform enables representatives to more efficiently tackle inbound client requests across any communications channel — email, web, social media, phone or chat. Features include Automatic Answers (a machine learning-powered tool for interpreting and solving customer questions and requests), Zopim (a real-time chat service) and Zendesk Voice (a cloud-based, built-in phone support solution). According to Zendesk, its business users experience positive ratings for more than 86 percent of their customer interactions.

8. DocuSign

Electronic signature technology and transaction management services platform DocuSign supports the exchange of digital contracts and other e-signed documents. Users may access, sign and send business documents from their office, their hotel room or anywhere else their job leads, guaranteeing approvals and agreements are executed in a matter of minutes, not days. DocuSign e-signatures are legally binding for most business and personal transactions in virtually every nation across the globe. The app supports more than 85 million users in 188

9. Dropbox

Keep your documents and files at your fingertips across all your devices using Dropbox. Anything added to Dropbox storage automatically shows up across all your desktop and mobile devices, enabling professionals to begin a project on their work PC, make edits on their smartphone during the evening commute home, and add the finishing touches from their home tablet. Then users can invite teammates to access any Dropbox folder or send them specific files and images accessible through password-protected links; there’s even a remote wipe option in case of emergency.

10. Slack

A real-time messaging, archiving and search solution, Slack is redefining business communication. Users may organize team conversations in open channels dedicated to specific topics or projects or limit more sensitive interactions to private, invite-only participants. Colleagues also may interact one-on-one using private, secure direct messages. Slack also enables users to share files, documents, spreadsheets and PDFs, complete with options for adding comments and highlighting for future reference; moreover, all messages, notifications and files are automatically indexed and archived.

Cloud application services, or Software as a Service (SaaS), represent the largest cloud market and are still growing quickly. SaaS uses the web to deliver applications that are managed by a third-party vendor and whose interface is accessed on the clients’ side. Most SaaS applications can be run directly from a web browser without any downloads or installations required, although some require plugins.

Because of the web delivery model, SaaS eliminates the need to install and run applications on individual computers. With SaaS, it’s easy for enterprises to streamline their maintenance and support, because everything can be managed by vendors: applications, runtime, data, middleware, OSes, virtualization, servers, storage and networking.

**Q.4 (a) Explain how live migration works. State two service providers of live migration**

Ans. Live Migration allows Hyper-V to overcome these limitations when moving VMs by removing the need for them to be temporarily suspended, thus removing downtime for applications running on the VM being moved. Live Migration uses a process to transfer memory pages from the current host to the destination host and then simply transfers ownership of the VM's virtual disks to the destination host.

eg:

* Accenture

Accenture uses a migration methodology that involves four phases: Inventory, High-Level Planning, Detailed Planning and Execution. The methodology relies on the company’s proprietary tool, known as Accelerate, that facilitates each step of the process. In the Inventory phase, it expedites the gathering of information to determine the current state of applications and infrastructure to define the scope. In the planning phases, Accelerate helps identify which workloads will be migrated, captures the dependencies across applications, and pinpoints technical limitations.

* Capgemini

Capgemini uses a cloud-agnostic migration methodology, called Capgemini Cloud Assessment, that focuses on low-risk, high-return business transformation.

Its methodology includes four phases: Plot, Scan, Craft and Solve. During these phases, it determines scope and execution strategy, performs analysis on the portfolio, applies filters to identify appropriate deployment for each application, and creates a road map with recommended steps for the cloud migration. Capgemini uses a proprietary ranking and scoring model to provide objective analysis.

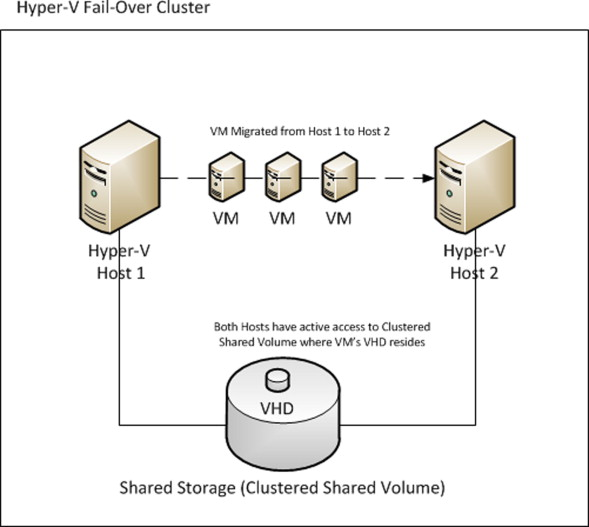
* Cognizant

Headquartered in New Jersey, Cognizant has more than 260,000 employees around the world. It helps customers assess and plan for secure cloud environments, set them up, and migrate workloads to the cloud using the Cognizant Cloud Steps Transformation Framework, a tool-based solution.

The framework is based on four steps: Assess and Plan, Set Up, Migrate, and Manage. It also offers prebuilt solution blueprints designed to accelerate cloud adoption for specific workloads. Blueprints for application migration include the Application Transformation Blueprint and the SAP Cloud Blueprint.

* Deloitte

Deloitte ATAmotion is an automated cloud migration engine that is managed through a central console. It is purpose-built for the enterprise and allows clients to migrate applications like SAP, SAP HANA and Oracle to private or public clouds.



**Q.4 (b) How AppFabric functions in Windows Azure?**

Ans. Azure AppFabric is a suite of middleware services and technologies to help you develop and manage services/applications that use Windows Azure. Middleware is typically defined as software that helps connect other pieces of software, and this definition is pretty accurate for the services appFabric provides.

You don't create an App Fabric per say. AppFabric services are used by your other applications as needed, so setup is typically configuring certain items in the Azure Portal, then implementing libraries of config entires in your web/worker roles that leverage the resources.

Essentially AppFabric provides certain resources that you need when composing complex applications as services, vs. you having to implement and maintain these resources yourself.

The basic offerings are:

Service Bus: A message relay that can be consumed by other .NET technologies (and others). SB helps you connect different cloud services as well as "hybrid" services. The hybrid is a big deal, as SB helps you easily connect on-premise web services with services you run in the cloud, w/o having to mess around with VPN, protocols, server setups, certificates, etc etc.

Access Control: An authentication and authorization service, helping you manage user-level access without having to extend/implement Active Directory, LDAP, and custom user authentication modules throughout Azure.

Caching: an in-memory distributed caching layer for your applications. This is typical to memcached or the Windows Server version of AppFabric

Integration: a PaaS service of EDI/transport technology like BizTalk server

Composite App: allows the composition of complex applications using a composition language versus just putting a bunch of code together. You basically define your application using a designer like you would a EF.Net data model or a Windows Workflow

So basically AppFabric provides you with a lot of services that you likely need, but the typical cloud developer may not want to "mess with" at least at first. This way you have these great building blocks to help you focus on your core logic/needs during development cycles while not limiting what your application can ultimately do. This "focus" is one of the core benefits to cloud computing, especially Platform as a Service, and is one area where Azure really shines compared to other offerings.

**Q.4 (c) A company has decided to leverage the web conferencing services provided by a cloud**

**provider and to pay for those services as they are used. The cloud provider manages the**

**infrastructure and any application upgrades. This is an example of what type of cloud**

**delivery model?**

Ans. Software As A Service (SAAS)

**Q.1 (a) List three reasons why a small business should choose cloud computing.**

Ans. Small businesses are finding the many attributes of cloud computing very attractive. Here are seven reasons why:

* Cloud computing can improve employee productivity. Using tools like Exchange Online and SharePoint Online, employees can easily access documents and emails while away from the office. Many small businesses have employees working remotely, who need both mobility and flexibility. Cloud computing is a straightforward and affordable way of addressing those needs, especially when you are using intuitive, familiar software that your employees already know.
* Cloud computing helps reduce upfront costs. Choosing a cloud-based solution for such things as email, document sharing, and web conferencing reduces the upfront cost associated with starting a business. There are no servers to buy. The cloud–based subscription model allows small businesses to easily increase or decrease their use of cloud services according to their needs.
* Cloud computing boosts collaboration. Employees everywhere can access and work with the same files in real time—no more emails back and forth with attachments to open. Better collaboration improves productivity and creativity.
* Cloud computing provides business resiliency. Numerous studies have shown that more than 50 percent of small businesses will go out of business within a year after a major data loss. In a cloud environment, if you lose your laptop—or worse, your whole office—you can get back to business in no time. All your business information and files are securely in place.
* Cloud computing is easy. The transfer of business information into the cloud is very straightforward—no need to install new hardware or software, no IT administration. It is often seamless to the user. Saving a document to a Microsoft OneDrive Pro follows the same process as saving it to a hard drive. No training is required.
* Cloud computing is accessible. Offering easy access and file sharing from any Internet connection, cloud computing allows employee access via smartphone or computer.
* Cloud computing can be more secure than a traditional IT infrastructure. Cloud- computing providers like Microsoft build multiple levels of security and redundancy into their data centers.

**Q.1 (b) Propose a novel cloud service that you would want for your start-up. Justify two important drivers how the stated service will address your business challenges.**

Ans.

**First Alternative Answer:**  
  
Assume I have a start up that runs E-Commerce website. Due to the COVID impacted shift in retail industry, customers are not going to physical stores to buy clothes. A novel cloud service that can used to create a Virtual Store is AWS Sumerian.  
  
How it addresses business challenges:

1. Removes the dependency of shopping experiences in exclusive physical stores completely powered by VR and AR experiences
2. Leverages other AWS services for NLP and Speech to create digital avatars that can help act as a concierge/shop assistant for FAQs and transactions

**Second Alternative Answer:**

Assume I have a start up that runs a job portal or website. A novel service I would probably use is an OCR service powered by AWS Textract and Comprehend  
How it addresses business challenges:

1. Enables auto extraction of text from uploaded CVs to reduce the number of fields filled by an applicant and thereby improves overall UX, and enhancing customer experience
2. Adds the Ability to extract keywords from the CV such as skillsets, and rank their CVs accordingly. This will help set up a recommendation engine in the future or roadmap.

**Third Alternative Answer:**

Assume I have a start up that provides IoT solutions. A novel service I would probably use is an AWS Sagemaker Autopilot.

How it addresses business challenges:

1. We can train a custom computer vision model to detect if a person is wearing a mask or not, and enhance the security feature in home IoT cameras.
2. Since we can detect people come over to homes are wearing masks or not, this data can be collated and contributed to organizations about percentage conversion of awareness of people wearing masks

**Q.1 (c) Differentiate between enterprise data centre and cloud service.**

| **Factor** | **Enterprise Data Centre** | **Cloud Service** |
| --- | --- | --- |
| **Demand** | Less in demand,  Enterprise Data Centre is costlier and technology often gets obsolete. | More in demand,  Cloud servers are much more affordable and practical compared to data centers |
| **Growth (Business)** | Comparatively steady growth | Comparatively more growing |
| **Users** | Tata Communications, Web Werks, Yotta, Nxtra Data Limited | Rapidshare, Moo, Vistaprint, Adobe creative cloud |
| **Applications** | * Data storage, management, backup and recovery * Productivity applications, such as email * High-volume e-commerce transactions * Powering online gaming communities * Big data, machine learning and artificial intelligence | * Art Applications(Moo, Vistaprint, and Adobe creative cloud) * File Storage Platform(Mediafire, Hotfile, Rapidshare) * Image Editing Applications(adobe Creative Cloud and Fotor) * Data Storage Applications(Box, mozy, jouks and Google Suite) * Antivirus Applications(sophos Endpoint Protection and Kaspersky Endpoint security cloud) * Entertainment Applications (Project Atlas and Google Stadia) * URL conversion Applications * Meeting Applications(GoToMeeting and Zoom) * Presentation Applications(Sliderocket) * Social Media Applications(Facebook, Twitter, yammer, Linkedin) * GPS Application (google maps, yahoo maps) * Accounting Application(Kash Flow and Zoho Books) * Management Application * e-Commerce Application(Amazon ) * Software as a Service (SaaS) Applications(FedEx) |

Ans.

**Demand**

Cloud servers are much more affordable and practical compared to data centers. Cloud servers are basically data centers that host data for different companies under one roof. They also supply various software services like office suites and applications to the companies for their operations

**Growth (Business)**

**Users**

**Applications**

Potential Disadvantages of Enterprise Data Centers

* Cost of Ownership. Building and equipping data center facilities (even ones that aren’t hyperscale) requires significant capital investments and ongoing expenditures of time, labor, and materials to maintain. This can put a heavy financial burden on a company—especially if it is a smaller organization that doesn’t have a large budget for IT infrastructure spending.
* Potentially Obsolete Technology. Many enterprise data centers represent legacy infrastructure, facilities that were often built during the first wave of internet growth before the era of cloud computing and extensive colocation solutions. And, because of the aforementioned expense of keeping privately-owned data centers up to date, companies are often unable to equip them with the technology needed to remain competitive with newer facilities.
* Communication Delays. The further a user or IoT device is from a data center, the greater the potential for latency or lag in communications. Additionally, with a single, centrally-located data center, there is a risk that peak loads may cause delays for all users. A centralized data center strategy may put organizations at increased risk for total failure if traffic exceeds the data center’s total capacity.

Advantages of Enterprise Data Centers

* Compatibility with Specific Software Applications. Enterprise data centers are well-suited for companies with unique network needs or those that do enough business to take advantage of economies of scale. They can be purpose-built for maximum compatibility with an organization’s enterprise applications and processes.
* IT Visibility. Another benefit of having dedicated enterprise data center infrastructure is that it is easy for organizations to ensure that their data centers have extensive monitoring tools installed. This gives the organization strong IT visibility to track things like bandwidth and power usage, which apps are seeing the most use, and so on. This makes it easier to estimate future data center needs so the organization can scale appropriately. It’s important to note that this benefit isn’t exclusive to on-prem data centers owned by the organization. Some outsourced data centers may also offer intelligent monitoring to ensure high IT visibility.

**Q.2 (a) Justify the need of API in cloud Applications. Differentiate between SOAP and REST?**

Ans.

Application developers use APIs to access cloud services through web-based communication. A developer writes code that calls the cloud provider's API, passes the requisite parameters and arguments in the context of the call, and then examines the response to confirm the operation.

A cloud API enables end users to access a cloud provider's application or service, such as compute infrastructure, storage resources or monitoring tools. APIs define the possible features and functions of that app or service, along with the details needed to execute them. APIs are typically based on REST or Simple Object Access Protocol (SOAP) communication schemes and rely on authorization schemes such as OAuth 2.0 to ensure user authentication and security.

| **SOAP** | **REST** |
| --- | --- |
| SOAP stands for Simple Object Access Protocol | REST stands for Representational State Transfer |
| SOAP is a protocol. SOAP was designed with a specification. It includes a WSDL file which has the required information on what the web service does in addition to the location of the web service. | REST is an Architectural style in which a web service can only be treated as a RESTful service if it follows the constraints of being  Client Server  Stateless  Cacheable  Layered System  Uniform Interface |
| SOAP cannot make use of REST since SOAP is a protocol and REST is an architectural pattern. | REST can make use of SOAP as the underlying protocol for web services, because in the end it is just an architectural pattern. |
| SOAP uses service interfaces to expose its functionality to client applications. In SOAP, the WSDL file provides the client with the necessary information which can be used to understand what services the web service can offer. | REST use Uniform Service locators to access to the components on the hardware device. For example, if there is an object which represents the data of an employee hosted on a URL as http://demo.guru99 , the below are some of URI that can exist to access them.  http://demo.guru99.com/Employee  http://demo.guru99.com/Employee/1 |
| SOAP requires more bandwidth for its usage. Since SOAP Messages contain a lot of information inside of it, the amount of data transfer using SOAP is generally a lot.  <?xml version="1.0"?>  <SOAP-ENV:Envelope  xmlns:SOAP-ENV  ="http://www.w3.org/2001/12/soap-envelope"  SOAP-ENV:encodingStyle  =" http://www.w3.org/2001/12/soap-encoding">  <soap:Body>  <Demo.guru99WebService  xmlns="http://tempuri.org/">  <EmployeeID>int</EmployeeID>  </Demo.guru99WebService>  </soap:Body>  </SOAP-ENV:Envelope> | REST does not need much bandwidth when requests are sent to the server. REST messages mostly just consist of JSON messages. Below is an example of a JSON message passed to a web server. You can see that the size of the message is comparatively smaller to SOAP.  {"city":"Mumbai","state":"Maharastra"} |
| SOAP can only work with XML format. As seen from SOAP messages, all data passed is in XML format. | REST permits different data format such as Plain text, HTML, XML, JSON, etc. But the most preferred format for transferring data is JSON. |

**Q.2 (b) State your justification on why you cannot make your cloud service available round the clock?**

Ans. Cloud services are on a pay-as-you-go model. We end up paying for what we use. But, unless it is a service deployed production that is used 24x7, we shouldn't make these services run round the clock.  
  
The reason is to get charged less when the cloud service is not used, especially around no load, zero user impact, or quiet hours.   
  
Example Scenario: We have dev/test or staging VMs that are not used during the night when the team is offline. It does not make sense to get charged for that time. Hence we shouldnt make it available round the clock unless its a production environment heavily used by customers/users.

**Q.2 (c) Computing Infrastructure as service: What does it mean and how is it possible?**

Ans.

Infrastructure as a service (IaaS) is a system of cloud computing that delivers virtualized computing resources over the internet. IaaS is one of the three main categories of cloud computing services, together with software as a service (SaaS) and platform as a service (PaaS).

IaaS quickly scales up or down with demand and helps avoid the need to procure physical servers and other data center infrastructure; each resource is offered as a distinct service component. A cloud computing service provider manages the infrastructure, while the user installs, configures, and manages software, including applications, middleware, and operating systems.

**Q.3 (a) Supposing you are fond of travelling and taking pictures on the go. Assuming that your smart phone may run out of space to store all the pics that you click, you decide to push them to amazon S3. What steps would be needed to store your pics in amazon S3.**

Ans.

Using the S3 console:

* Sign in to the AWS Management Console and open the Amazon S3 console
* Choose Create bucket
* In Bucket name, enter a DNS-compliant name for your bucket
* In Region, choose the AWS Region where you want the bucket to reside.
* In Bucket settings for Block Public Access, choose the Block Public Access settings that you want to apply to the bucket.
* Choose Create bucket.
* In the Amazon S3 console, choose the bucket where you want to upload an object, choose Upload, and then choose Add Files.
* In the file selection dialog box, find the file that you want to upload, choose it, choose Open, and then choose Start Upload.
* You can watch the progress of the upload in the Transfer pane.

**Q.3 (b) Supposing Mr. Modi seeks your advice on MeghRaj cloud. Suggest him the benefits of hybrid cloud? Suggest a design improvement for MeghRaj owned by the Indian Government to have storage support in case of Cloud bursting. Show your design through a clean sketch.**

Ans.

**MeghRaj cloud**

In order to utilise and harness the benefits of Cloud Computing, Government of India has embarked upon an ambitious initiative - "GI Cloud" which has been named as "MeghRaj". The focus of this initiative is to accelerate delivery of e-services in the country while optimizing ICT spending of the Government. This will ensure optimum utilization of the infrastructure and speed up the development and deployment of eGov applications. The architectural vision of GI Cloud encompasses a set of discrete cloud computing environments spread across multiple locations, built on existing or new (augmented) infrastructure, following a set of common protocols, guidelines and standards issued by the Government of India. Two Policy reports viz., "GI Cloud Strategic Direction Paper" and "GI Cloud Adoption and Implementation Roadmap" have been prepared by Meity.

**Advantages of Hybrid Cloud Solutions**

1. Meet High Service Demands with Cloud Bursting

Hybrid clouds help organizations satisfy periods of high demand for applications or services without the high costs of overprovisioning their own data centers for peak demand. Configure your applications that normally run on private cloud or on-prem to burst into the public cloud during times of excess demand, when additional computing resources are required to supply the necessary capacity and ensure customer satisfaction.

Cloud bursting is especially useful for industries and organizations that see regular fluctuations in their demand for data and application services. For example, financial applications experience surges in demand at the end of each quarter and during tax season. Another example is the retail industry, where seasonal events like Christmas and Black Friday result in greater volumes of web traffic and more purchases.

Cloud bursting is a great way for organizations in these and other industries to maintain high-quality customer experiences in periods of high application demand.

1. Make Data & Applications Accessible for Remote Workers

Cloud-hosted data and applications can be accessed from anywhere in the world by members of your organization with the appropriate authorization. In contrast, data and applications stored in on-premise data centers may be difficult to access or experience unacceptable lag times for other geographies.

Organizations with hybrid cloud solutions can leverage their infrastructure to provide employees with remote access to cloud-based and on-prem data and applications.

1. Comply with Data Privacy and Localization Laws

Organizations in some industries are subject to data security and privacy regulations such as the Health Insurance Portability and Accountability Act (HIPAA), the European General Data Privacy Regulation (GDPR), and the Payment Card Industry Data Security Standard (PCI DSS). Many countries have also passed data localization laws that require organizations to store personal customer data in the same country where it is collected.

With hybrid cloud solutions, organizations do not need to build data centers in each country they operate in. They can store locally gathered customer data in public cloud regions that comply with data localization requirements and protect their customers’ privacy as required by law.

1. Enhance Data Security & Protect Sensitive Data

Hybrid cloud solutions give organizations greater flexibility when it comes to securing their data. Businesses can store their most sensitive data in on-premise data centers where it is difficult for malicious actors to attack. At the same time, use public cloud storage for easily and quickly processing and analyzing less-sensitive data.

1. Increase Scalability of Operations

Public cloud services providers have made data storage and computing resources affordable and accessible at scale for businesses. Organizations can scale their operations on an as-needed basis and avoid the capital expenditures, management requirements, and technical overhead associated with building or expanding on-prem data centers.

1. Improve Disaster Recovery & Business Continuity

Organizations that rely on hybrid cloud are now developing portable applications that can run on either on-premise or public cloud infrastructure. Portable applications offer a significant competitive advantage when the organization experiences a service outage and must execute its disaster recovery strategy to maintain business continuity and avoid unplanned downtime that negatively impacts the customer experience.

1. Lower IT Spending & Operational Costs

Reducing IT spending and lowering total cost of ownership are key strategic objectives that have driven hybrid cloud adoption over the past five years.

Organizations that depend on on-prem infrastructure and data centers need enough servers to meet customer expectations during peak usage times. An organization might invest in 500 servers to accommodate peak demand, but require just 200 servers during normal or “average” demand periods. Buying for peak demand results in underutilization of IT resources and inefficient IT spending.

In contrast, an organization with hybrid cloud capabilities could invest in just 200 servers and rely on public cloud vendors to supply additional resources during times of peak demand. As a result, these organizations benefit from higher utilization rates for IT resources, lower costs, and more efficient IT spending.

1. Shift Capital Expenditures to Operational Expenditures in IT

Adopting hybrid cloud solutions can create tax advantages for organizations in Canada, the United States, and other jurisdictions.

Purchasing servers for a data center is considered a capital expenditure in the tax code. As a result, the servers are codified as a business asset and must be depreciated on a yearly basis until the end of their life cycle. If an organization spends $5,000 on a server that is expected to last five years, they would have to depreciate the asset by $1,000 per year for five years to claim the full tax benefit.

Purchasing cloud services, however, is considered an operating expenditure – not a capital expenditure. Operating expenditures are not subject to depreciation – the organization can claim the full amount of the expense as an income reduction in the same year that the costs were incurred. This means that an organization purchasing $5,000 in public cloud services could claim the entire tax benefit right away and reduce their tax liability accordingly.

1. Access Cutting-Edge Tools and Technologies

Organizations with hybrid cloud capabilities enjoy the maximum flexibility when it comes to discovering and implementing best-of-breed technologies to support every application workload. The ability to deploy any application to the computing environment which promises the best performance, lowest costs, and most reliable service can create significant competitive advantages. Orgs can expand their access to technology and avoid cloud vendor lock-in by integrating more than one public cloud into their hybrid cloud solution.

1. Drive Innovation

Hybrid cloud solutions are driving innovation, making it easier for firms to build new services, and empowering organizations to better meet the needs of their customers. Businesses can use public cloud resources to develop and test new applications, then deploy them into production as needed.

**Q.3 (c) Is it mandatory to opt for elastic IP? What will be the repercussions if an organization doesn’t opt for elastic IPs.**

Ans. It is not mandatory to opt for elastic IP.

An Elastic IP has the public IP address component, as you need to advertise your AWS instances to the public internet. As the AWS Elastic IP will advertise a public, static IP address that can compensate for fluctuations in your AWS infrastructure, then they are necessary for dynamic cloud computing.

If an organization doesn't use elastic IPs, then in case of instance failure the address will change in these fluctuations and the organization will have to keep mapping everytime in such events.

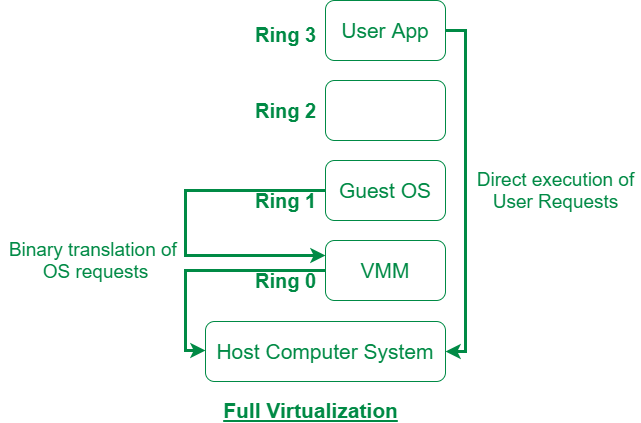
**Q.4 (a) Fill the below table with yes/no.**

| **Factors about Guest OS** | **Full Virtualization** | **Para-Virtualization or OS-Assisted Virtualization** | **Hardware assisted Virtualization** |
| --- | --- | --- | --- |
| **Has no role in virtualization** | No  Full virtualization is a virtualization technique used to provide a VME that completely simulates the underlying hardware. | Yes  Paravirtualization (PV) is an enhancement of virtualization technology in which a guest operating system (guest OS) is modified prior to installation inside a virtual machine (VM) in order to allow all guest OS within the system to share resources and successfully collaborate, rather than attempt to emulate an enti re hardware environment. | No  Hardware-assisted virtualization is a technology that allows for a CPU instruction set communication in which the VMM runs in a new root level mode below the OS kernel level. |
| **Remains unaware about virtualization** | No | Yes | No |
| **Normal version of OS can be used** | Yes  Normal Version (Microsoft and Parallels systems) | No  Guest OS codified to issue Hypercalls, so it cannot run on Native Hardware or other Hypervisors Poor compatibility; Not available on Windows OSes | Yes  Normal Version  (allow all guest OS) |
| **Performance is low** | Yes  Lower | No  Higher | No  Higher |

Ans.

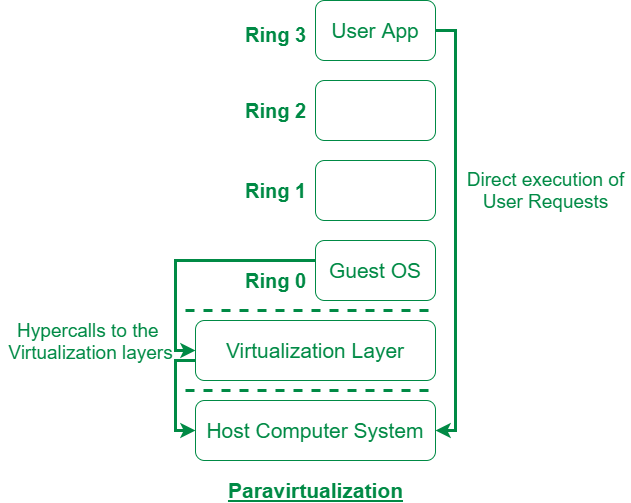
**Full Virtualization:**

It is the first software solution of server virtualization and uses binary translation and direct approach technique. In full virtualization, guest OS is completely isolated by the virtual machine from the virtualization layer and hardware. Microsoft and Parallels systems are examples of full virtualization.

****

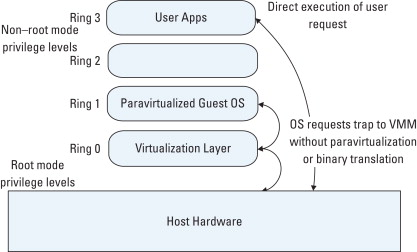
**Paravirtualization:**

Paravirtualization is the category of CPU virtualization which uses hypercalls for operations to handle instructions at compile time. In paravirtualization, guest OS is not completely isolated but it is partially isolated by the virtual machine from the virtualization layer and hardware. VMware and Xen are some examples of paravirtualization.



**Hardware assisted Virtualization**

Hardware-assisted virtualization is also called native virtualization, accelerated virtualization, or hardware VM, depending on the vendor. Hardware-assisted virtualization is a technology that allows for a CPU instruction set communication in which the VMM runs in a new root level mode below the OS kernel level.



|  | **Full Virtualization with Binary Translation** | **OS-Assisted Virtualization and Paravirtualization** | **Hardware-Assisted Virtualization** |
| --- | --- | --- | --- |
| Technique | Binary Translation and Direct Execution | Hypercalls | Exit to Root Mode on Privileged Instructions |
| Guest Modification and Compatibility | Unmodified Guest OS and Excellent compatibility | Guest OS codified to issue Hypercalls, so it cannot run on Native Hardware or other Hypervisors Poor compatibility; Not available on Windows OSes | Unmodified Guest OS and Excellent compatibility |
| Performance | Good | Better in certain cases | Fair Current performance lags Binary Translation virtualization on various workloads but improves over time |
| Used By | VMware, Microsoft, Parallels | VMware, Xen | VMware, Microsoft, Parallels, Xen |
| Guest OS Hypervisor Independent? | Yes | XenLinux runs only on Xen Hypervisor VMI-Linux is Hypervisor agnostic | Yes |

**Q.4 (b) What is the need of App Engine memory cache? Can you store application session data in memory cache? Justify your answer.**

Ans.

High performance scalable web applications often use a distributed in-memory data cache in front of or in place of robust persistent storage for some tasks. App Engine includes a memory cache service for this purpose.

We store application session data in memory cache, but values can expire from the memcache at any time, and can expire prior to the expiration deadline set for the value. For example, if the sudden absence of a user's session data would cause the session to malfunction, that data should probably be stored in the datastore in addition to the memcache.

**Q.1 (a) How the cloud services are measured in AWS? What are the resources that are measured?**

Ans.

Cloud technology is bringing in many benefits to the organizations and the services are measured in several ways.

* **Elasticity**: Cloud has the ability to create more resources to enhance performance for a single user or numerous users at a single point of time.
* **Instant provisioning**: Virtualization of infrastructure eliminates the need to deploy applications into devices or into on premises location.
* **Cost-effective**: Cloud adopts the policy of pay-as-you-use and there is no upfront investment to be made. This differs from the tradition policy of pay and use.
* **Better usage**: Consumers use the service without the need of planning in advance for what they may require and wasting them if not utilized.
* **Utilization**: The vital benefit is it removes the drawbacks of under-provisioning and over-provisioning that generally wastes a lot of money for organizations.
* **On demand**: Yet another feature of the cloud hosting is the on demand self-service which gives the user ability to automated services with just a few clicks.
* **Access flexibility**: Cloud network allows access to networks for a wide range of devices like smartphones, tablets, PCs, within a corporate firewall. It allows access to data and applications from any location at any time.
* **Control**: Users have the option to select and configure the applications and infrastructure that best suits their business needs.

Cloud offers high flexibility and better data security and recovery solutions and gives an edge over competition for businesses.

**Q.1 (b) Differentiate between EBS backed instances and S3 backed instances in AWS?**

Ans.



**Q.1 (c) What is private IP, Public IP and Elastic IP? What will be the problem if there were no**

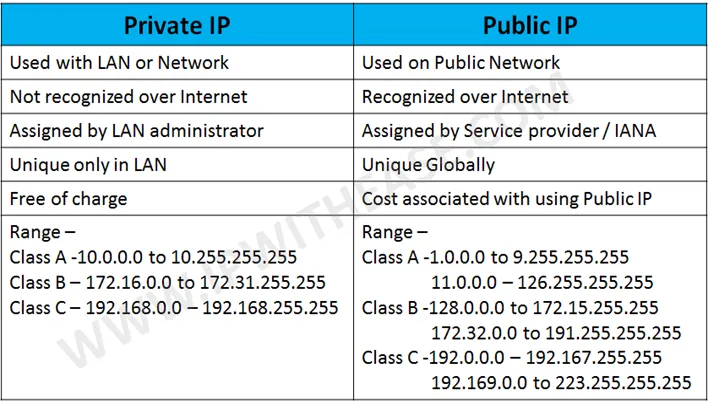
**provision of elastic IP?**

Ans.

**Private IP** addresses let devices connected to the same network communicate with one another without connecting to the entire internet. By making it more difficult for an external host or user to establish a connection, private IPs help bolster security within a specific network, like in your home or office.

**Public IP** address is an IP address that can be accessed directly over the internet and is assigned to your network router by your internet service provider (ISP). Your personal device also has a private IP that remains hidden when you connect to the internet through your router's public IP.

An **Elastic IP** address is a reserved public IP address that you can assign to any EC2 instance in a particular region, until you choose to release it. The Elastic IP address remains in place through events that normally cause the address to change, such as stopping or restarting the instance.



**Q.2 (a) Explain the difference between horizontal and vertical scaling with an example.**

Ans.

**Horizontal Scaling: Cassandra, MongoDB**

Horizontal scaling (aka scaling out) refers to adding additional nodes or machines to your infrastructure to cope with new demands. If you are hosting an application on a server and find that it no longer has the capacity or capabilities to handle traffic, adding a server may be your solution.

It is quite similar to delegating workload among several employees instead of one. However, the downside of this may be the added complexity of your operation. You must decide which machine does what and how your new machines work with your old machines.

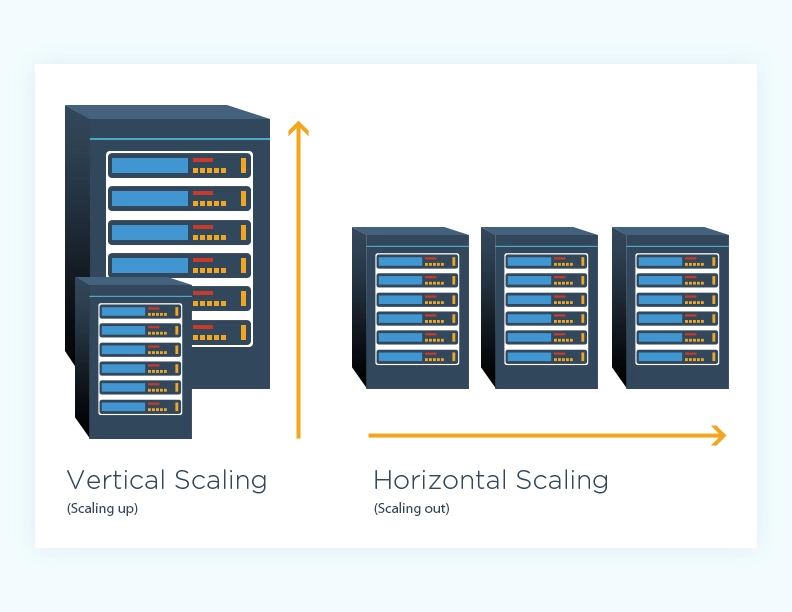
You can consider this the opposite of vertical scaling.

**Vertical Scaling: MySQL - Amazon RDS (The cloud version of MySQL)**

Vertical scaling (aka scaling up) describes adding additional resources to a system so that it meets demand. How is this different from horizontal scaling?

While horizontal scaling refers to adding additional nodes, vertical scaling describes adding more power to your current machines. For instance, if your server requires more processing power, vertical scaling would mean upgrading the CPUs. You can also vertically scale the memory, storage, or network speed.

Additionally, vertical scaling may also describe replacing a server entirely or moving a server’s workload to an upgraded one.



**Advantages of horizontal scaling**

* Scaling is easier from a hardware perspective - All horizontal scaling requires you to do is add additional machines to your current pool. It eliminates the need to analyze which system specifications you need to upgrade.
* Fewer periods of downtime - Because you’re adding a machine, you don’t have to switch the old machine off while scaling. If done effectively, there may never be a need for downtime and clients are less likely to be impacted.
* Increased resilience and fault tolerance - Relying on a single node for all your data and operations puts you at a high risk of losing it all when it fails. Distributing it among several nodes saves you from losing it all.
* Increased performance - If you are using horizontal scaling to manage your network traffic, it allows for more endpoints for connections, considering that the load will be delegated among multiple machines.

**Disadvantages of horizontal scaling**

* Increased complexity of maintenance and operation - Multiple servers are harder to maintain than a single server is. Additionally, you will need to add software for load balancing and possibly virtualization. Backing up your machines may also become a little more complex. You will need to ensure that nodes synchronize and communicate effectively.
* Increased Initial costs - Adding new servers is far more expensive than upgrading old ones.

**Advantages of vertical scaling**

* Cost-effective - Upgrading a pre-existing server costs less than purchasing a new one. Additionally, you are less likely to add new backup and virtualization software when scaling vertically. Maintenance costs may potentially remain the same too.
* Less complex process communication - When a single node handles all the layers of your services, it will not have to synchronize and communicate with other machines to work. This may result in faster responses.
* Less complicated maintenance - Not only is maintenance cheaper but it is less complex because of the number of nodes you will need to manage.
* Less need for software changes - You are less likely to change how the software on a server works or how it is implemented.

**Disadvantages of vertical scaling**

* Higher possibility for downtime - Unless you have a backup server that can handle operations and requests, you will need some considerable downtime to upgrade your machine.
* Single point of failure - Having all your operations on a single server increases the risk of losing all your data if a hardware or software failure was to occur.
* Upgrade limitations - There is a limitation to how much you can upgrade a machine. Every machine has its threshold for RAM, storage, and processing power.

**Which Should You Choose And When?**

Both horizontal and vertical scaling have their own benefits and limitations. Since there isn’t a one-size-fits-all solution for organizations, you need to scale according to your needs and resources. Here are a few factors to consider along with which type of scaling suits the situation best:

* **Cost** - Initial hardware costs for horizontal upgrades are higher. If you are working on a tight budget and need to add more resources to your infrastructure quickly and cheaply, then vertical scaling may be the best option for you.
* **Future-proofing** - Adding additional updated machines through horizontal scaling will increase the overall performance threshold of your organization. There is a limit to how much you can vertically scale a single node and it may not be able to handle the demands of the future.
* **Topographic distribution** - If you plan to have nationwide or global clients, it is unreasonable to expect them all to access your services from a single machine in a single location. In a situation like this, you’ll need to horizontally scale your resources to maintain your service level agreement (SLA).
* **Reliability** - Horizontal scaling may offer you a more reliable system. It increases redundancy and ensures that you are not relying on a single machine. If one machine fails, another may be able to pick up the slack temporarily.
* **Upgradability and flexibility** - If you are running your application’s tiers on individual machines, they are easier to decouple and upgrade without any downtime.
* **Performance and complexity** - Performance will depend on how your services work and how they are interconnected. Simple straightforward applications won’t benefit much from being run on multiple machines. In fact, it may degrade its quality. Sometimes it’s better to leave the application as is and upgrade the hardware to meet demand. Horizontal scaling may require you to rewrite the code or add a virtual machine that unifies all the servers.

**Q.2 (b) What is CAP principle? How is it relevant to Cloud Technology?**

Ans.

The CAP theorem, originally introduced as the CAP principle, can be used to explain some of the competing requirements in a distributed system with replication. It is a tool used to make system designers aware of the trade-offs while designing networked shared-data systems.

The three letters in CAP refer to three desirable properties of distributed systems with replicated data: consistency (among replicated copies), availability of the system for read and write operations) and partition tolerance in the face of the nodes in the system being partitioned by a network fault).

The CAP theorem states that it is not possible to guarantee all three of the desirable properties – consistency, availability, and partition tolerance at the same time in a distributed system with data replication.

The theorem states that networked shared-data systems can only strongly support two of the following three properties:

Consistency –

Consistency means that the nodes will have the same copies of a replicated data item visible for various transactions. A guarantee that every node in a distributed cluster returns the same, most recent, successful write. Consistency refers to every client having the same view of the data. There are various types of consistency models. Consistency in CAP refers to sequential consistency, a very strong form of consistency.

Availability –

Availability means that each read or write request for a data item will either be processed successfully or will receive a message that the operation cannot be completed. Every non-failing node returns a response for all read and write requests in a reasonable amount of time. The key word here is every. To be available, every node on (either side of a network partition) must be able to respond in a reasonable amount of time.

Partition Tolerant –

Partition tolerance means that the system can continue operating if the network connecting the nodes has a fault that results in two or more partitions, where the nodes in each partition can only communicate among each other. That means, the system continues to function and upholds its consistency guarantees in spite of network partitions. Network partitions are a fact of life. Distributed systems guaranteeing partition tolerance can gracefully recover from partitions once the partition heals.

**Q.2 (c) What is the importance of rings in OpenStack Swift architecture?**

Ans.

**Swift Architectural Overview:**

* **The Ring**

A ring represents a mapping between the names of entities stored on disk and their physical location. There are separate rings for accounts, containers, and one object ring per storage policy. When other components need to perform any operation on an object, container, or account, they need to interact with the appropriate ring to determine its location in the cluster.

* **Proxy Server**

The Proxy Server is responsible for tying together the rest of the Swift architecture. For each request, it will look up the location of the account, container, or object in the ring (see below) and route the request accordingly. For Erasure Code type policies, the Proxy Server is also responsible for encoding and decoding object data. See Erasure Code Support for complete information on Erasure Code support. The public API is also exposed through the Proxy Server.

* **Storage Policies**

Storage Policies provide a way for object storage providers to differentiate service levels, features and behaviors of a Swift deployment. Each Storage Policy configured in Swift is exposed to the client via an abstract name. Each device in the system is assigned to one or more Storage Policies. This is accomplished through the use of multiple object rings, where each Storage Policy has an independent object ring, which may include a subset of hardware implementing a particular differentiation.

* **Object Server**

The Object Server is a very simple blob storage server that can store, retrieve and delete objects stored on local devices. Objects are stored as binary files on the filesystem with metadata stored in the file’s extended attributes (xattrs). This requires that the underlying filesystem choice for object servers support xattrs on files. Some filesystems, like ext3, have xattrs turned off by default.

Each object is stored using a path derived from the object name’s hash and the operation’s timestamp. Last write always wins, and ensures that the latest object version will be served. A deletion is also treated as a version of the file (a 0 byte file ending with “.ts”, which stands for tombstone). This ensures that deleted files are replicated correctly and older versions don’t magically reappear due to failure scenarios.

* **Container Server**

The Container Server’s primary job is to handle listings of objects. It doesn’t know where those object’s are, just what objects are in a specific container. The listings are stored as sqlite database files, and replicated across the cluster similar to how objects are. Statistics are also tracked that include the total number of objects, and total storage usage for that container.

* **Account Server**

The Account Server is very similar to the Container Server, excepting that it is responsible for listings of containers rather than objects.

* **Replication**

Replication is designed to keep the system in a consistent state in the face of temporary error conditions like network outages or drive failures.

The replication processes compare local data with each remote copy to ensure they all contain the latest version. Object replication uses a hash list to quickly compare subsections of each partition, and container and account replication use a combination of hashes and shared high water marks.

Replication updates are push based. For object replication, updating is just a matter of rsyncing files to the peer. Account and container replication push missing records over HTTP or rsync whole database files.

The replicator also ensures that data is removed from the system. When an item (object, container, or account) is deleted, a tombstone is set as the latest version of the item. The replicator will see the tombstone and ensure that the item is removed from the entire system.

* **Reconstruction**

The reconstructor is used by Erasure Code policies and is analogous to the replicator for Replication type policies. See Erasure Code Support for complete information on both Erasure Code support as well as the reconstructor.

* **Updaters**

There are times when container or account data can not be immediately updated. This usually occurs during failure scenarios or periods of high load. If an update fails, the update is queued locally on the filesystem, and the updater will process the failed updates. This is where an eventual consistency window will most likely come in to play. For example, suppose a container server is under load and a new object is put in to the system. The object will be immediately available for reads as soon as the proxy server responds to the client with success. However, the container server did not update the object listing, and so the update would be queued for a later update. Container listings, therefore, may not immediately contain the object.

In practice, the consistency window is only as large as the frequency at which the updater runs and may not even be noticed as the proxy server will route listing requests to the first container server which responds. The server under load may not be the one that serves subsequent listing requests – one of the other two replicas may handle the listing.

* **Auditors**

Auditors crawl the local server checking the integrity of the objects, containers, and accounts. If corruption is found (in the case of bit rot, for example), the file is quarantined, and replication will replace the bad file from another replica. If other errors are found they are logged (for example, an object’s listing can’t be found on any container server it should be).

**Q.3 (a) Do you think authorizing someone with login and password credentials is enough to access cloud services? Justify your answer.**

Ans. Yes.

The most common way to access your cloud service is from any web browser; navigate to the cloud service website and log in, and there are your files. example: OneDrive even lets you preview and examine files online; you can edit Microsoft Office documents if you subscribe to the Office 365 service. Other examples: Dropbox, G Suite, Microsoft Office 365, Slack and Citrix Content Collaboration.

**Q.3 (b) Supposing you want to store your memorable marriage pics in Amazon S3. What steps will you have to follow?**

Ans.

Using the S3 console:

* Sign in to the AWS Management Console and open the Amazon S3 console
* Choose Create bucket
* In Bucket name, enter a DNS-compliant name for your bucket
* In Region, choose the AWS Region where you want the bucket to reside.
* In Bucket settings for Block Public Access, choose the Block Public Access settings that you want to apply to the bucket.
* Choose Create bucket.
* In the Amazon S3 console, choose the bucket where you want to upload an object, choose Upload, and then choose Add Files.
* In the file selection dialog box, find the file that you want to upload, choose it, choose Open, and then choose Start Upload.
* You can watch the progress of the upload in the Transfer pane.

**Q.3 (c) Can we do live migration if storage resources are not separate from computing**

**resources? Justify your answer.**

Ans.

Azure

**Q.4 (a) What are the different blob services supported in Microsoft Azure?**

Ans.

Azure Blob storage is a feature of Microsoft Azure. It allows users to store large amounts of unstructured data on Microsoft’s data storage platform. In this case, Blob stands for Binary Large Object, which includes objects such as images and multimedia files. These are known as unstructured data because they don’t follow any particular data model.

While Azure data storage is generally associated with data lakes and analytics, Azure Blob storage is more focused on common storage purposes. In other words, it includes objects which personal users would be used to storing, such as photos, videos, and documents. On an enterprise level, the amount of these files, as well as log files and backups is considerably higher, hence the need for Azure.

Blob storage for an organization, for example, would mean keeping backups of previous versions of a site and its content and images, which would need to be accessible from different locations. By using Azure data storage, a company can make this material easier and quicker to access compared to it being kept on a single-location server.

With Azure Blob storage, the files (photos, videos, training documents, etc.), which are known as blobs, are put in containers which function similar to directories. These are then linked to the storage account. When creating the address to give access to a file in Azure data storage, it will simply join the storage account and the location of the blob. The address will be in a .net format.

One of the big advantages for businesses is that Azure Blob storage allows them to collect all of their content assets in one place. These will then be available all across their different departments and internationally. The speed, scalability, ease of access, and security (both from accidents and criminals) make such cloud storage very attractive for all mid-sized and large organizations.

**Q.4 (b) Explain the working of Table service in Azure with a suitable example.**

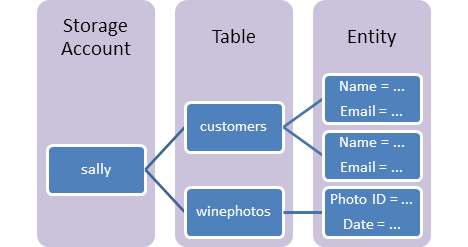
Ans.

Azure Table storage stores large amounts of structured data. The service is a NoSQL datastore which accepts authenticated calls from inside and outside the Azure cloud. Azure tables are ideal for storing structured, non-relational data. Common uses of Table storage include:

* Storing TBs of structured data capable of serving web scale applications
* Storing datasets that don't require complex joins, foreign keys, or stored procedures and can be denormalized for fast access
* Quickly querying data using a clustered index
* Accessing data using the OData protocol and LINQ queries with WCF Data Service .NET Libraries
* You can use Table storage to store and query huge sets of structured, non-relational data, and your tables will scale as demand increases.

**Table storage concepts**

Table storage contains the following components:



* URL format: Azure Table Storage accounts use this format: http://<storage account>.table.core.windows.net/<table>

Azure Cosmos DB Table API accounts use this format: http://<storage account>.table.cosmosdb.azure.com/<table>

You can address Azure tables directly using this address with the OData protocol. For more information, see OData.org.

* Accounts: All access to Azure Storage is done through a storage account. For more information about storage accounts, see Storage account overview.

All access to Azure Cosmos DB is done through a Table API account. See Create a Table API account for details creating a Table API account.

* Table: A table is a collection of entities. Tables don't enforce a schema on entities, which means a single table can contain entities that have different sets of properties.
* Entity: An entity is a set of properties, similar to a database row. An entity in Azure Storage can be up to 1MB in size. An entity in Azure Cosmos DB can be up to 2MB in size.
* Properties: A property is a name-value pair. Each entity can include up to 252 properties to store data. Each entity also has three system properties that specify a partition key, a row key, and a timestamp. Entities with the same partition key can be queried more quickly, and inserted/updated in atomic operations. An entity's row key is its unique identifier within a partition.

**Q.4 (c) What is the role of Memcache in Cloud computing?**

Ans.

High performance scalable web applications often use a distributed in-memory data cache in front of or in place of robust persistent storage for some tasks. App Engine includes a memory cache service for this purpose.

One use of a memory cache is to speed up common datastore queries. If many requests make the same query with the same parameters, and changes to the results do not need to appear on the web site right away, the application can cache the results in the memcache. Subsequent requests can check the memcache, and only perform the datastore query if the results are absent or expired. Session data, user preferences, and other data returned by queries for web pages are good candidates for caching.

Memcache can be useful for other temporary values. However, when considering whether to store a value solely in the memcache and not backed by other persistent storage, be sure that your application behaves acceptably when the value is suddenly not available. Values can expire from the memcache at any time, and can be expired prior to the expiration deadline set for the value. For example, if the sudden absence of a user's session data would cause the session to malfunction, that data should probably be stored in the datastore in addition to the memcache.

**Service levels**

App Engine supports two levels of the memcache service:

* Shared memcache is the free default for App Engine applications. It provides cache capacity on a best-effort basis and is subject to the overall demand of all the App Engine applications using the shared memcache service.
* Dedicated memcache provides a fixed cache capacity assigned exclusively to your application. It's billed by the GB-hour of cache size and requires billing to be enabled. Having control over cache size means your app can perform more predictably and with fewer reads from more costly durable storage.

Both memcache service levels use the same API.

**Limits**

The following limits apply to the use of the memcache service:

* The maximum size of a cached data value is 1 MB (10^6 bytes).
* A key cannot be larger than 250 bytes. In the Python runtime, keys that are strings longer than 250 bytes will be hashed. (Other runtimes behave differently.)
* The "multi" batch operations can have any number of elements. The total size of the call and the total size of the data fetched must not exceed 32 megabytes.
* A memcache key cannot contain a null byte.

**Q.1 (a) How would you know if an organization is IaaS or PaaS provider?**

Ans. SaaS, PaaS, and IaaS are simply three ways to describe how you can use the cloud for your business.

IaaS: cloud-based services, pay-as-you-go for services such as storage, networking, and virtualization.

PaaS: hardware and software tools available over the internet.

SaaS: software that’s available via a third-party over the internet.

On-premise: software that’s installed in the same building as your business.

**Q.1 (b) What could be the security disadvantages of virtualization in the context of cloud computing?**

Ans.

The Disadvantages of Virtualization

1. It can have a high cost of implementation.

The cost for the average individual or business when virtualization is being considered will be quite low. For the providers of a virtualization environment, however, the implementation costs can be quite high. Hardware and software are required at some point and that means devices must either be developed, manufactured, or purchased for implementation.

2. It still has limitations.

Not every application or server is going to work within an environment of virtualization. That means an individual or corporation may require a hybrid system to function properly. This still saves time and money in the long run, but since not every vendor supports virtualization and some may stop supporting it after initially starting it, there is always a level of uncertainty when fully implementing this type of system.

3. It creates a security risk.

Information is our modern currency. If you have it, you can make money. If you don’t have it, you’ll be ignored. Because data is crucial to the success of a business, it is targeted frequently. The average cost of a data security breach in 2017, according to a report published by the Ponemon Institute, was $3.62 million. For perspective: the chances of being struck by lightning are about 1 in a million. The chances of experiencing a data breach while using virtualization? 1 in 4.

4. It creates an availability issue.

The primary concern that many have with virtualization is what will happen to their work should their assets not be available. If an organization cannot connect to their data for an extended period of time, they will struggle to compete in their industry. And, since availability is controlled by third-party providers, the ability to stay connected in not in one’s control with virtualization.

5. It creates a scalability issue.

Although you can grow a business or opportunity quickly because of virtualization, you may not be able to become as large as you’d like. You may also be required to be larger than you want to be when first starting out. Because many entities share the same resources, growth creates lag within a virtualization network. One large presence can take resources away from several smaller businesses and there would be nothing anyone could do about it.

6. It requires several links in a chain that must work together cohesively.

If you have local equipment, then you are in full control of what you can do. With virtualization, you lose that control because several links must work together to perform the same task. Let’s use the example of saving a document file. With a local storage device, like a flash drive or HDD, you can save the file immediately and even create a backup. Using virtualization, your ISP connection would need to be valid. Your LAN or Wi-Fi would need to be working. Your online storage option would need to be available. If any of those are not working, then you’re not saving that file.

7. It takes time.

Although you save time during the implementation phases of virtualization, it costs users time over the long-run when compared to local systems. That is because there are extra steps that must be followed to generate the desired result.

**Q.1 (c) Which architecture model client-server or grid computing will be more convenient to move to the cloud? Justify your answer.**

Ans.

| **Cloud Computing** | **Grid Computing** |
| --- | --- |
| Cloud Computing follows client-server computing architecture. | Grid computing follows a distributed computing architecture. |
| Scalability is high. | Scalability is normal. |
| Cloud Computing is more flexible than grid computing. | Grid Computing is less flexible than cloud computing. |
| Cloud operates as a centralized management system. | Grid operates as a decentralized management system. |
| In cloud computing, cloud servers are owned by infrastructure providers. | In Grid computing, grids are owned and managed by the organization. |
| Cloud computing uses services like Iaas, PaaS, and SaaS. | Grid computing uses systems like distributed computing, distributed information, and distributed pervasive. |
| Cloud Computing is Service-oriented. | Grid Computing is Application-oriented. |
| It is accessible through standard web protocols. | It is accessible through grid middleware. |

In Grid Computing, resources are managed on collaboration pattern. **Cloud Computing is more flexible than Grid Computing.** Grid Computing is less flexible. Users pay for using the cloud computing resources.

**Q.1 (d) \_\_\_\_\_\_\_\_\_\_ provides a monitoring service in Amazon Cloud Computing.**

**Ans.**

CloudWatch provides a monitoring service in Amazon Cloud Computing.

Amazon CloudWatch is a monitoring and observability service built for DevOps engineers, developers, site reliability engineers (SREs), and IT managers. CloudWatch provides you with data and actionable insights to monitor your applications, respond to system-wide performance changes, optimize resource utilization, and get a unified view of operational health. CloudWatch collects monitoring and operational data in the form of logs, metrics, and events, providing you with a unified view of AWS resources, applications, and services that run on AWS and on-premises servers. You can use CloudWatch to detect anomalous behavior in your environments, set alarms, visualize logs and metrics side by side, take automated actions, troubleshoot issues, and discover insights to keep your applications

running smoothly.

**Q.2 (a) Identify three uses of APIs in realizing Amazon Cloud Computing.**

Ans.

An application program interface (API) allows software programs to communicate, making them more functional. An AWS user creates, manages and maintains APIs within the Amazon API Gateway.

An API gateway accepts and processes concurrent API calls, which happen when APIs submit requests to a server. It manages traffic, authorizes end users and monitors performance. Using Amazon API Gateway, a developer can connect with services such as:

* AWS Elastic Beanstalk;
* Elastic Compute Cloud instances;
* AWS Lambda event-driven code; and
* AWS Step Functions state machines.

**Q.2 (b) What happens to the EBS storage after the termination of the instance? Justify your answer.**

Ans. By default, Amazon EBS root device volumes are automatically deleted when the instance terminates. However, by default, any additional EBS volumes that you attach at launch, or any EBS volumes that you attach to an existing instance persist even after the instance terminates.

**Q.2 (c) Why vertical partitioning will not be suitable for cloud storage?**

Ans.

**Horizontal partitioning** involves putting different rows into different tables. Perhaps customers with ZIP codes less than 50000 are stored in CustomersEast, while customers with ZIP codes greater than or equal to 50000 are stored in CustomersWest. The two partition tables are then CustomersEast and CustomersWest, while a view with a union might be created over both of them to provide a complete view of all customers.

**Vertical partitioning** involves creating tables with fewer columns and using additional tables to store the remaining columns. Normalization also involves this splitting of columns across tables, but vertical partitioning goes beyond that and partitions columns even when already normalized.

**Q.3 (a) Supposing you want to host an online quiz test platform. For utmost security you want to**

**provision as many VMs as the number of candidates. What steps will you follow to**

**provision the VMs?**

Ans. Azure

**Q.3 (b) Supposing while the online test is going on, one of the VM faces some technical issues, then how will you migrate the VM.**

Ans. Azure

**Q.3 (c) What are web and worker roles in Azure working model?**

Ans. There are two types of Azure Cloud Services roles. The only difference between the two is how your role is hosted on the VMs:

Web role: Automatically deploys and hosts your app through IIS.

Worker role: Does not use IIS, and runs your app standalone.

**Q.4 (a) “Cloud computing will enhance e-Governance”. State two reasons in support and two in against of the above statement.**

Ans.

**Cloud based e-governance**

E-governance with cloud computing offers integration management with automated problem resolution, manages security end-to-end, and helps budget based on actual usage of data. At a global level, cloud architectures can benefit government to reduce duplicate efforts and increase utilisation of resources. This helps the government going green, reducing pollution and effective waste management.

Enterprises and SMBs are already reaping the benefits of cloud by using the “pay-as-you-use” service model, its massive scalability and readily availability. Since government requires a massive infrastructure it is important for government to use cloud computing on long-term basis.

No one should be deluded by the complexity and scale of services and hurdles to overcome to implement such a large-scale programme for a country like India because of its size and distribution. With proper planning, execution, training and good management could reduce overall costs to a great extent and help in more efficient utilisations of taxpayers’ money. Though there are initiatives already being planned but it is time for e-governance to take a giant leap.

E-governance is a process of reform in the way governments work, share information, engage citizens and deliver services to external and internal clients for the benefit of both government and the clients that they serve. Government has innumerable applications that can be automated. Spending on IT would increase the productivity of the government and help in decision-making and policy enforcement.

**Key challenges**

There are several challenges we need to address on our way to build e-governance systems that are successfully deployed and actually end up benefiting its users. E-governance databases should be scalable, to deal with large data over the years for a range of applications. Cloud databases available for deployment offer unprecedented level of scaling without compromising on the performance. Cloud databases, therefore, must be considered if the foremost concern is on-demand, high-end scalability—that is, large scale, distributed scalability, the kind that can’t be achieved simply by scaling up.

Corruption in government organisations can be controlled by using IT services, by keeping the providers of the services accountable. Process audits, security audits must be done periodically to ensure the security of the system. Cloud can help in analysing huge volumes of data and detecting any fraud.

Natural disasters like floods, earthquakes, wars and internal disturbances could cause the e-governance applications not only loose data, but also make services unavailable. Multiple installations in geographically separated locations with complete backup and recovery solutions must exist. Cloud virtualisation technologies allow backups and restoring. It offers application migration seamlessly compared to traditional data centre.

**Q.4 (b) When do you use HTTP get and when do you use HTTP post?**

Ans.

GET is used for viewing something, without changing it.

The GET method requests a representation of the specified resource. Requests using GET should only retrieve data. The HEAD method asks for a response identical to that of a GET request, but without the response body.

POST is used for changing something, For example, a search page should use GET to get data while a form that changes your password should use POST . Essentially GET is used to retrieve remote data, and POST is used to insert/update remote data.

For example, a search page should use GET to get data while a form that changes your password should use POST . Essentially GET is used to retrieve remote data, and POST is used to insert/update remote data.

**Q.4 (c) What is the main purpose of App Engine memory cache? Do you think storing session data in memory cache is a good idea? Justify your answer.**

Ans.

One use of a memory cache is to speed up common datastore queries. If many requests make the same query with the same parameters, and changes to the results do not need to appear on the web site right away, the application can cache the results in the memcache. Subsequent requests can check the memcache, and only perform the datastore query if the results are absent or expired. Session data, user preferences, and other data returned by queries for web pages are good candidates for caching.

Memcache can be useful for other temporary values. However, when considering whether to store a value solely in the memcache and not backed by other persistent storage, be sure that your application behaves acceptably when the value is suddenly not available. Values can expire from the memcache at any time, and can be expired prior to the expiration deadline set for the value.

**session data storing in memory cache is not a good idea.**

For example, if the sudden absence of a user's session data would cause the session to malfunction, that data should probably be stored in the datastore in addition to the memcache.

**Elastic Load Balancing** automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, Lambda functions, and virtual appliances. It can handle the varying load of your application traffic in a single Availability Zone or across multiple Availability Zones. Elastic Load Balancing offers four types of load balancers that all feature the high availability, automatic scaling, and robust security necessary to make your applications fault tolerant.

In Computer Science, the ordered protection domains are referred to as **Protection Rings**. These mechanisms help in improving fault tolerance and provide Computer Security. Operating Systems provide different levels to access resources. Rings are hierarchically arranged from most privileged to least privileged.

**Use of Protection Ring :**

Use of Protection Rings provides logical space for the levels of permissions and execution. Two important uses of Protection Rings are :

1. Improving Fault Tolerance

2. Provide Computer Security

**Levels of Protection Ring :**

There are basically 4 levels ranging from 0 which is the most privileged to 3 which is least privileged. Most Operating Systems use level 0 as the kernel or executive and use level 3 for application programs. A resource that is accessible to level n is also accessible to levels 0 to n and the privilege levels are rings.

*Protection Rings*

**Modes of Protection Ring :**

There are basically two modes : Supervisor Mode, and Hypervisor Mode. These are explained as following below in brief.

1. **Supervisor Mode :**

Supervisor Mode is an execution mode in some of processors which allows execution of all instructions including privileged instructions. It also gives access to different address space, to memory management hardware, and to other peripherals. Usually, Operating System runs in this mode.

2. **Hypervisor Mode :**

Modern CPUs offer x86 virtualization instructions for hypervisor to control “Ring 0” hardware access. In order to help virtualization, VT and Pacifica insert new privilege level below “Ring 0” and Both these add nine new “machine code” instructions that only work on Ring −1 and intended to be used by hypervisor.

**Implementation :**

Protection Rings are combined with processor modes against rule of slave in some systems. Operating system running on hardware that supports such rules can use both methods of protecting or only one of them. Efficient use of architecture of Protection Rings requires close interaction between hardware and operating system.

The operating system is designed such that they have worked on a lot of platforms and may have different implementation mechanism rings on each platform. The security model is generally simplified to two levels of access – first is level of “core” and second is level of “user”, even if hardware providing greater granularity of performance levels.

**Features of Protection Ring :**

· Protection Ring follows hierarchy.

· Protection Ring provides layered architecture.

· Protection Ring provides Computer Security.

· Protection Ring provides good Fault Tolerance.

Attention reader! Don’t stop learning now. Get hold of all the important CS Theory concepts for SDE interviews with the [**CS Theory Course**](https://practice.geeksforgeeks.org/courses/SDE-theory?vC=1) at a student-friendly price and become industry ready.

# 1. Platform as a Service and SaaS

Software as a service is a software licensing and delivery model in which software is licensed on a subscription basis and is centrally hosted. SaaS is also known as "on-demand software" and Web-based/Web-hosted software

## 1.1 Introduction to PaaS

Platform as a service (PaaS) or application platform as a service (aPaaS) or platform-based service is a category of cloud computing services that allows customers to provision, instantiate, run, and manage a modular bundle comprising a computing platform and one or more applications, without the complexity of building and maintaining the infrastructure typically associated with developing and launching the application(s); and to allow developers to create, develop, and package such software bundles

## 1.2 PaaS examples

PaaS examples: AWS Elastic Beanstalk, Heroku, Windows Azure (mostly used as PaaS), Force.com, OpenShift, Apache Stratos, Magento Commerce Cloud.

## 1.3 Windows Azure

**What is Microsoft Azure used for?**

At its core, Azure is a public cloud computing platform—with solutions including Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) that can be used for services such as analytics, virtual computing, storage, networking, and much more.

**What is Windows Azure in cloud computing?**

Microsoft Azure, formerly known as Windows Azure, is Microsoft's public cloud computing platform. It provides a range of cloud services, including compute, analytics, storage and networking. The Azure platform aims to help businesses manage challenges and meet their organizational goals.

**What is Microsoft Azure and how it works?**

Azure is Microsoft's public cloud platform. Azure offers a large collection of services including platform as a service (PaaS), infrastructure as a service (IaaS), and managed database service capabilities. Azure, like other cloud platforms, relies on a technology known as virtualization.

**What are the benefits of Azure?**

The 5 major benefits of Microsoft Azure

· Speed of service.

· Enhanced flexibility.

· Integrated delivery pipeline.

· Disaster recovery.

· Security.

**AWS vs Azure?**

| **Parameters** | **AWS** | **Azure** |
| --- | --- | --- |
| Launched Date | Launched in 2006. | Launched In 2010. |
| Market Share | 31% Share of the global computing market | 11% Share from the worldwide market. |
| Availability Zone | 61 Availability Zone | 140 Availability Zones |
| Storage services | * S3 * Buckets * EBS * SDB * domains * Easy to use * SQS * CloudFront * AWS Import/Export | * Blob Storage * Containers * Azure Drive * Table Storage * Tables * Storage Stats |
| Databases Services | * MySQL * Oracle * DynamoDB | * MS SQL * SQL Sync |
| Deployment Services | * Amazon Web Services * Amazon Machine Instance (AMI) * Traditional Deployment Models * Fine-grained updates * Elastic Beanstalk * Cloud Formation | Cspkg (fancy zip file)  Upload via portal or API via blob storage  Course-grained updates  “click to scale.”  More magic |
| Networking Services | * IP/Elastic IP/ELB * Virtual Private Cloud * Route 53 * ELB * Firewall heavily configurable | * Automatic IP assignment * Load-balancing * Azure Connect * Balancing * Endpoints defined in csdef/cscfg |
| Price | Per hour- rounded up | On-demand reserved spot. |
| Customers | Adobe, Airbnb, Expedia, Yelp, Nokia, Netflix, Novartis. | Pearson, 3M, Towers Watson, NBC, Essar, Serko, etc. |
| Type of Cloud | Virtual Private Cloud (VPC) | Virtual Network |
| Connection type | Direct Connect | ExpressRoute |
| Pricing models | * Free Tier * Per Hour * Free Trial Per Minute * No change for stopped * Pay for EBS volume | * Free Trial * Per Minute |
| Government Cloud | AWS has an edge as far as government cloud offerings. | Limited reach for government cloud offerings. |
| Support for Hybrid cloud | Does not offers the best of hybrid cloud support. | With Hybrid Cloud, organizations can integrate onsite servers with Cloud instances. |
| Ecosystem | AWS has a software marketplace with an extensive partner ecosystem. | With very few Linux options, Azure doesn’t’ have a big ecosystem. |
| Support for Big Data | EBS storage is ideal for handling big data. | Standard storage has many issues for big data, and therefore you need premium storage. |
| Maturity | More mature cloud environment for big data. | The less mature environment for big data. |
| Machine access | In AWS machine can be accessed separately. | Machines are grouped into cloud service and respond to the same domain name with various ports. |
| Salary | The average salary for “AWD engineer” is approximately $141,757 per year for Software Architect. | The average salary for “Microsoft Azure” ranges from approximately $113,582 per year. |
| Key features | Zero setups, Detail Monitoring, Auto-scaling groups. | Startup friendly, High performance, Low cost. |
| Long term data archiving | Allows long term data archiving and retrieval. | Does not offer any long term data archiving and retrieval option. |
| Security | Security is provided using defined roles with permission control feature. | Provides security by offering permissions on the whole account. |

**Advantages of AWS**

Here, are significant advantages of adopting AWS cloud services:

· Compute Cloud allows you to increase or decrease storage according to the need of your organization

· AWS enables you to select an operating system, programming language, database of your choice.

· Broad & deep service offerings

· Robust partner ecosystem

· Trusted by high-profile customers

· High Transfer Stability

· Minimal information is lost during server and storage transfer

· Offers more data centers for availability and low latency

· Better DevOps support

· Simpler licensing method

· Stronger support for Bl and analytics

**Advantages of Azure**

Here, are some major advantages of using Azure cloud services:

· Capability for developers and users to create, maintain and deploy applications

· Fully scalable cloud computing platform offers open access across multiple languages, frameworks, and tools

· Total support for Microsoft legacy apps

· Greater awareness of enterprise needs

· Easy one-click migrations in many cases

· Conversion of on-prem licenses to the cloud

· Support for mixed Linux/Windows environments

· Offers inbuilt tool like Azure stack to help the organization deliver Azure service from the own data center

**Disadvantages of AWS**

Here, are few drawbacks of Amazon Web Services:

· Less hybrid- cloud-friendly

· AWS elastic load balancer is not equipped to handle as many requests as it receives

· AWS lacks customer support, so it more suitable for a technically savvy group of consumers and those companies who have their inbuild tech support team

· The number of choices offered by AWS is confusing to those who may not speak the language of technology.

· Incompatible and Weak Hybrid Strategy

· AWS is a less open private cloud. This makes it an unpopular storage option for sensitive industries like banking

· AWS has too many products which makes the selection process much harder

**Disadvantages of Azure**

The major Drawbacks of Azure cloud services are:

· Customer service is not transparent, and data is hosted globally. So, if you have data restrictions where it must be stored in a specific country, at that time you need to verify/specify with Microsoft

· You will be charged extra for paying as you go

· Azure cloud-based services are full of glitches. To fix these bugs, you will need to spend additional money

· Less flexibility about non-Windows server platforms, when compared to AWS

**Which one is better AWS or Azure?**

Now, let’s see AWS vs Azure which is better for your needs?

Microsoft Azure has increased its market share in the last couple of years, but not to an extent where there is a real contest between the two companies at least for the near future.

Moreover, both companies introduce new products, new integrations, and new pricing structures. Therefore, the final selection will be depend on the need of your organization.

**How do I install Microsoft Azure?**

Installing and Configuring Microsoft Azure Service Broker

· Step 1: Set up Azure.

· Step 2: Create a Service Broker Database.

· Step 3: Install and Configure Microsoft Azure Service Broker. Azure Config. Broker Config. SQL Database Config. ...

· Step 4: Confirm Installation.

· Step 5: Confirm Service and Plan Access.

## 1.4 5 Principles of UI Design - AWS PaaS

Some key design principles of the AWS Cloud include **scalability, disposable resources, automation, loose coupling managed services instead of servers, and flexible data storage options**.

**What are the 5 pillars of AWS well-architected framework?**

Based on five pillars — **operational excellence, security, reliability, performance efficiency, and cost optimization** — AWS Well-Architected provides a consistent approach for customers and partners to evaluate architectures, and implement designs that can scale over time.

**What are the 3 areas of operational excellence in the cloud?**

Operational Excellence in the Cloud

The guidelines and design principles we discussed earlier allow the AWS environment (and the systems it supports) to be seen from three major points of view: **preparation, operation, and evaluation**.

## 1.5 Introduction to SaaS

Software as a service is a software licensing and delivery model in which software is licensed on a subscription basis and is centrally hosted. SaaS is also known as "on-demand software" and Web-based/Web-hosted software.

## 1.6 Pros and Cons of SaaS model and applications

### i. The pros

**Cost**

For businesses, subscription-based software licensing make it simpler to understand and allocate costs for separate business units or departments. Plus, it can be easier to account for a steady expenditure rather than one, large cost every few years.

Publishers using the as a service model may also have multiple tiers of pricing, allowing businesses to pay less in exchange for access to fewer features of the application. This has created a lower buying threshold, giving smaller businesses access to software they might otherwise have been unable to afford.

**Maintenance**

Perhaps the best benefit of SaaS is the automatic access to patches and updates.

With perpetual software licensing, businesses used one iteration of an application until it absolutely *had* to be updated to the latest version — either for security reasons or to access new features.

A subscription-based model means the publisher will automatically update your licenses as new versions are released. Your employees won’t be using outdated tools and the business doesn’t have to spring for a whole new application.

**Mobility**

Today’s employees are looking for flexibility in their working lives — and workplace mobility plays a big part in that. As a result, businesses are embracing remote working policies.

This trend means that the software your employees depend on needs to be accessible from anywhere — it’s another reason SaaS is gaining in popularity. With no disc to be installed, SaaS-based applications can be used anywhere there’s a network connection. It’s helping the mobile workplace become more attainable.

### ii. The cons

**Security**

Publishers aren’t responsible for ensuring data security when using the SaaS model for their applications. Instead, it’s the business’s responsibility to make sure there are appropriate security safeguards and protocols in place.

The growing popularity of mobility in the workplace has created a somewhat unique challenge for IT departments as they struggle to secure mobile devices in remote locations. Endpoint security is now vital in order to safeguard important data in an increasingly mobile world.

The rise of SaaS has contributed to [the need for strong endpoint security](https://www.insight.com/en_US/content-and-resources/brands/adobe/security-concerns-overcome-moving-to-saas.html). Employees will want — and need — to take workplace software applications on the go. And businesses need to accommodate this trend without opening themselves to malicious attacks.

**Contractual obligations**

Compliance is a huge concern for businesses partaking in SaaS offerings. SaaS contracts can be difficult to understand and the penalties for overusing licenses are often steep. Companies [found to be out of compliance](https://www.insight.com/en_US/content-and-resources/2020/how-to-survive-a-publisher-software-audit.html) by the publisher may end up having to pay a large, lump sum in order to get up to date quickly.

This means that Software Asset Management (SAM) has become an important part of the IT department’s job. The company needs to know how many licenses each business unit or department is using — as well as how many the business is entitled to — in order to avoid over- and under-usage.

**Loss of control**

With the perpetual software sales model, applications were largely controlled by the business that used them.

The SaaS model turns much of that control over to the publisher.

Now, that *can* be a good thing. For instance, instead of having to gauge when to upgrade to the latest version of an application and going through the process of installing that new version, the publisher rolls out automatic updates.

However, it also means publishers keep a close eye on their clients’ usage. And they won’t hesitate to audit a company they suspect is out of compliance with its contract. The resulting process is largely controlled by the publisher and, if not prepared for it, businesses can find themselves struggling to navigate it.

Consider having a knowledgeable third party available to help your organization get through the software audit with the least amount of financial repercussions

# 2. Capacity management and Scheduling in cloud computing

### i. Cloud capacity management tools

Cloud capacity management is a complex, multifaceted process, and there is no single tool that will meet all of your capacity planning needs. A variety of tool types can assist in the process, including:

· **Monitoring and log management.** Data collected by monitoring and logging tools such as AWS CloudWatch, Azure Monitor and third-party monitoring platforms can help you keep track of performance trends and alert you to changing capacity needs.

· **Infrastructure as code.** [Infrastructure-as-code](https://searchitoperations.techtarget.com/definition/Infrastructure-as-Code-IAC) tools automate infrastructure setup and resource allocation, so it is much easier and faster to reconfigure allocations in response to capacity changes.

· **Cost calculators.** To manage the financial aspects of capacity planning, the cost-prediction tools that cloud providers offer are useful. They can help evaluate the costs associated with different resource allocations or workload types.

· **Right-sizing and cost management.** Cloud providers offer tools designed to help predict capacity requirements. [AWS has a cost management tool](https://aws.amazon.com/aws-cost-management/), [as does Microsoft Azure](https://azure.microsoft.com/en-us/services/cost-management/). Some third-party application performance management (APM) tools also offer right-sizing features.

### ii. Steps to manage cloud capacity

The nature of cloud architectures and services varies widely, so there is no single or simple way to approach cloud capacity. In general, however, an effective cloud capacity management strategy will involve several key steps.

#### 1. Assess baseline capacity requirements

First, determine how many cloud servers, application instances, databases and so on your team requires on average to maintain adequate performance. You'll need to know how many CPU, memory and storage resources each workload requires -- these are your baseline capacity requirements. It's important to remember that you shouldn't use that baseline to make resource allocations, especially if demands placed on the workloads often fluctuate. Still, knowing your baseline provides a starting point for capacity planning.

#### 2. Assess scalability needs

Once you know the baseline requirements for each workload that you run in the cloud, examine the scalability they'll require. Evaluate how much variation occurs to workload demand between different times of day, days of the week or seasons of the year. Some of your cloud workloads will have higher scalability requirements than others. For instance, a website with a globally dispersed user base probably won't see as much fluctuation in usage in a full day as a website that caters to users in a specific geographic location, which likely will see most demand during that locale's daytime hours. Likewise, a website for a meal-delivery service will probably experience higher load during mealtimes than at other times of day.

#### 3. Make initial resource allocations

For workloads that don't already run in the cloud, you'll need to set initial resource allocations before you start them. Plan to allocate 20% more resources to each workload than the baseline requirements dictate. This provides a healthy buffer [in case demand unexpectedly jumps](https://searchcloudcomputing.techtarget.com/tip/A-guide-to-cloud-capacity-planning-for-unexpected-spikes) but doesn't unreasonably overprovision your environment.

#### 4. Set up autoscaling policies

Mainstream [public cloud](https://searchcloudcomputing.techtarget.com/definition/public-cloud) providers allow you to create so-called [autoscaling](https://searchcloudcomputing.techtarget.com/definition/autoscaling) policies. With these policies in place, the cloud platform automatically increases or decreases the resource allocations assigned to your workloads based on the traffic thresholds you configure in the policies. You can apply autoscaling policies to most types of cloud workloads, including virtual machine instances, databases, containers and serverless functions. However, certain niche categories of cloud workloads, such as IoT devices, typically can't be managed using autoscaling.

#### 5. Collect and analyze capacity data

Whether or not you configure autoscaling for your workloads, it's important to constantly assess how well the allocations work and adjust accordingly. Consider these metrics and factors:

· How often do your autoscaling policies trigger? If they are rarely applied because your workloads never reach the minimum thresholds for autoscaling, the workloads are likely overprovisioned. It may be time to reconfigure your thresholds.

· How do your actual cloud costs, as reflected in monthly bills, compare to your anticipated costs? Beating cost expectations is one sign that you are managing capacity well; when you find [cloud expenses are too high](https://searchcloudcomputing.techtarget.com/feature/5-ways-to-reduce-cloud-costs), you could probably do a better job at capacity management.

· How often do you experience disruptions or downtime related to capacity or resource allocation?

· How often does your team intervene manually to correct a capacity issue? You might reduce the need for manual changes with more intensive autoscaling or migrate your workload to a different type of architecture, such as serverless.

· Do the baseline workload requirements and the anticipated scalability needs that you identified for each workload remain consistent with actual performance?

## 2.1 Capacity management and Scheduling

**Why the cloud needs capacity management?**

Consider a cloud server that hosts several web applications. Proper capacity management ensures that the server runs on a virtual server instance with enough CPU, memory and storage resources to support the applications, but not so many resources that a significant portion goes unused.

Another aspect of capacity management is to determine how many servers to include in a cluster that shares responsibility for hosting an application. In this case, the IT team must be sure to include enough servers to handle the load placed on the application and also keep sufficient backup systems in place to guarantee the application remains available in the event some servers crash.

This balancing act is the key to capacity management. An organization wants to avoid both under provisioning workloads in such a way that they cannot perform adequately, and overprovisioning them by allocating resources they do not need.

**Effective capacity management:**

**Provide insight into long-term IT planning**. For example, capacity management can help determine which workloads to move to the cloud. Workloads with fast-changing capacities are ideal candidates for the cloud, where resource allocations can be easily scaled up and down.

**Determine which infrastructural and application architectures align with your needs**. For instance, if you have a virtual server with routinely fluctuating capacity demands, you might find that serverless functions would be a better way to host that workload. Serverless functions allow you to allocate large amounts of resources for short periods in a more cost-effective and easy-to-manage way than is possible with virtual servers.

**Arrange the right people and tools**. This is a step beyond your team knowing how many resources to allocate to workloads. It's important to find out if you have the organizational resources necessary to assign those resources. You'll need staff on hand to perform the necessary provisioning, and those workers should have the requisite skills to work with the tools you use to manage resource allocation.

**Avoid disruptions to users.** Wrong-sized workloads can create problems for the people who expect a specific application to be ready for them when they need it. When your workload capacities are well managed, you minimize your risk of having applications or servers fail.

While it has been a part of IT workflows for decades, capacity management has become especially important since the emergence of cloud computing. This is because scalability is a crucial factor in an organization's decision to migrate to the cloud. To capitalize fully on that scalability, however, IT teams must manage resource utilization effectively and continuously. If they can't, they miss one of the chief advantages of cloud architecture. Such companies might do better to stick with on-premises architectures.

## 2.2 Distributed management of virtual machines

Cloud orchestration is the end-to-end automation of the deployment of services in a cloud environment. More specifically, it is the automated arrangement, coordination, and management of complex computer systems, middleware, and services—all of which helps to accelerate the delivery of IT services while reducing costs. It is used to manage cloud infrastructure, which supplies and assigns required cloud resources to the customer like the creation of VMs, allocation of storage capacity, management of network resources, and granting access to cloud software. By using the appropriate orchestration mechanisms, users can deploy and start using services on servers or on any cloud platforms.

## 2.3 Reservation-based provisioning of virtualized resource

**What are three resource provisioning methods in cloud computing?**

The three models are **advanced provisioning, dynamic provisioning and user self-provisioning.**

**WHAT IS resource provisioning in cloud computing?**

Resource Provisioning means **the selection, deployment, and run-time management of software** (e.g., database server management systems, load balancers) and hardware resources (e.g., CPU, storage, and network) for ensuring guaranteed performance for applications.

**What is reservation in cloud computing?**

An IT resource reservation system is **established to protect cloud service consumers** (“tenants”) sharing the same underlying IT resources from each other.

**What is a provisioning resource?**

Overview. Provisioning is the **process of setting up IT infrastructure**. It can also refer to the steps required to manage access to data and resources, and make them available to users and systems. Provisioning is not the same thing as configuration, but they are both steps in the deployment process.

**What are the benefits of resource reservation architecture?**

In terms of benefits added, Resource Reservation Service will: **improve visibility of cloud resources consumption** (current and planned for future); enable cloud resource planning based on current and future demand from end users; automate the processes of resource allocation and reclaiming.

**What are the benefits of resource reservation architecture?**

In terms of benefits added, Resource Reservation Service will: **improve visibility of cloud resources consumption** (current and planned for future); enable cloud resource planning based on current and future demand from end users; automate the processes of resource allocation and reclaiming.

## 1.1 Provisioning to meet SLA commitments

IaaS clouds can be used to deploy services that will be consumed by users other than the one that has deployed the services.

• There is a distinction between the cloud consumer (i.e., the service owner; for instance, the company that develops and manages the applications) and the end users of the resources provisioned on the cloud (i.e., the service user; for instance, the users that access the applications).

• Furthermore, service owners will enter into service-level agreements (SLAs) with their end users, covering guarantees such as the timeliness with which these services will respond.

• Requirements are formalized in infrastructure SLAs between the service owner and cloud provider, separate from the high-level SLAs between the service owner and its end users.

• In many cases, either the service owner is not resourceful enough to perform an exact service sizing or service workloads are hard to anticipate in advance.

• Therefore, to protect high-level SLAs, the cloud provider should cater for elasticity on demand.

• Scaling and de-scaling of an application is best managed by the application itself. The reason is that in many cases, resource allocation decisions are application-specific and are being driven by the application level metrics.

## 1.2 Stages of VM life cycle within OpenNebula

The life-cycle of a Virtual Machine within OpenNebula includes the following stages:

| **Short state** | **State** | **Meaning** |
| --- | --- | --- |
| pend | Pending | By default a VM starts in the pending state, waiting for a resource to run on. It will stay in this state until the scheduler decides to deploy it, or the user deploys it using the onevm deploy command. |
| hold | Hold | The owner has held the VM and it will not be scheduled until it is released. It can be, however, deployed manually. |
| clon | Cloning | The VM is waiting for one or more disk images to finish the initial copy to the repository (image state still in lock) |
| prol | Prolog | The system is transferring the VM files (disk images and the recovery file) to the host in which the virtual machine will be running. |
| boot | Boot | OpenNebula is waiting for the hypervisor to create the VM. |
| runn | Running | The VM is running (note that this stage includes the internal virtualized machine booting and shutting down phases). In this state, the virtualization driver will periodically monitor it. |
| migr | Migrate | The VM is migrating from one resource to another. This can be a life migration or cold migration (the VM is saved, powered-off or powered-off hard and VM files are transferred to the new resource). |
| hotp | Hotplug | A disk attach/detach, nic attach/detach operation is in process. |
| snap | Snapshot | A system snapshot is being taken. |
| save | Save | The system is saving the VM files after a migration, stop or suspend operation. |
| epil | Epilog | In this phase the system cleans up the Host used to virtualize the VM, and additionally disk images to be saved are copied back to the system datastore. |
| shut | Shutdown | OpenNebula has sent the VM the shutdown ACPI signal, and is waiting for it to complete the shutdown process. If after a timeout period the VM does not disappear, OpenNebula will assume that the guest OS ignored the ACPI signal and the VM state will be changed to **running**, instead of **done**. |
| stop | Stopped | The VM is stopped. VM state has been saved and it has been transferred back along with the disk images to the system datastore. |
| susp | Suspended | Same as stopped, but the files are left in the host to later resume the VM there (i.e. there is no need to re-schedule the VM). |
| poff | PowerOff | Same as suspended, but no checkpoint file is generated. Note that the files are left in the host to later boot the VM there.  When the VM guest is shutdown, OpenNebula will put the VM in this state. |
| unde | Undeployed | The VM is shut down. The VM disks are transferred to the system datastore. The VM can be resumed ater. |
| fail | Failed | The VM failed. |
| unkn | Unknown | The VM couldn’t be reached, it is in an unknown state. |
| clea | Cleanup-resubmit | The VM is waiting for the drivers to clean the host after a onevm recover --recreate action |
| done | Done | The VM is done. VMs in this state won’t be shown with onevm list but are kept in the database for accounting purposes. You can still get their information with the onevm show command. |

## 1.3 6.6. Network model for OpenNebula

OpenNebula allows the creation of Virtual Networks by mapping them on top of the physical ones.

**What is the use of OpenNebula?**

The two primary uses of the OpenNebula platform are **data center virtualization and cloud deployments based** on the KVM hypervisor, LXD system containers, and AWS Firecracker microVMs. The platform is also capable of offering the cloud infrastructure necessary to operate a cloud on top of existing VMware infrastructure.

**How is OpenNebula beneficial?**

OpenNebula is the cloud management tool which **helps to synchronize the storage, network and virtual techniques**, and also helps the users to deploy and manage virtual machines on physical resources according to the allocation strategies at data centers and remote cloud resources dynamically.

**What is hypervisor used for?**

A hypervisor, also known as a virtual machine monitor or VMM, is **software that creates and runs virtual machines (VMs)**. A hypervisor allows one host computer to support multiple guest VMs by virtually sharing its resources, such as memory and processing.

**Who uses Opennebula?**

We have data on 518 companies that use OpenNebula. The companies using OpenNebula are most often found in United States and in the Computer Software industry.

| Company | University of California-Los Angeles |
| --- | --- |

**Is Docker a hypervisor?**

Docker originally used Linux Containers (LXC) and was designed for Linux kernel only. ... In the case of Windows, Docker uses Hyper-V which is in-built virtualization technology provided by Windows. Docker **uses Hypervisor framework in the case of MacOs for virtualization**.

**What is difference hypervisor and Docker?**

The most significant difference between hypervisors and Dockers is **the way they boot up and consume resources**. Hypervisors are of two types – the bare metal works directly on the hardware while type two hypervisor works on top of the operating system. Docker, on the other hand, works on the host kernel itself.

**What is difference hypervisor and Docker?**

The most significant difference between hypervisors and Dockers is **the way they boot up and consume resources**. Hypervisors are of two types – the bare metal works directly on the hardware while type two hypervisor works on top of the operating system. Docker, on the other hand, works on the host kernel itself.

# 2. Issues and Challenges : Availability, Multi-Tenancy, Security and SLA

Cloud orchestration is the end-to-end automation of the deployment of services in a cloud environment. More specifically, it is the automated arrangement, coordination, and management of complex computer systems, middleware, and services—all of which helps to accelerate the delivery of IT services while reducing costs. It is used to manage cloud infrastructure, which supplies and assigns required cloud resources to the customer like the creation of VMs, allocation of storage capacity, management of network resources, and granting access to cloud software. By using the appropriate orchestration mechanisms, users can deploy and start using services on servers or on any cloud platforms.

## 2.1 Multi-Tenancy, 4 levels of multi tenancy

**THE CHALLENGES OF MULTI-TENANCY:**

Regarded as one of the most important features of cloud computing, multi-tenancy is a key common attribute of both public and private spaces. It applies to all three layers of a cloud (IaaS, PaaS and SaaS) and refers to a software architecture design in which a single instance of a software application serves multiple customers.

**CAPACITY OPTIMIZATION**

Database administrators need the tools and the knowledge to understand which tenant should be deployed on which network in order to maximise capacity and reduce costs. This is process is further complicated by the need to continuously align capacity with business demand and requires providers to manage the actual and forecasted resource utilization for all their servers.

**SERVICE DELIVERY AND HIGH AVAILABILITY**

When failures occur or when certain services generate abnormal loads the service delivery can be interrupted – yet business clients will often request high-availability, typically 99.999 percent. Therefore, monitoring the service delivery and its availability is critical to ensure that the service is properly delivered and meeting SLAs. Without effective monitoring problems are hard to locate and downtimes are increased – often leading to lost revenue.

**MONITORING**

According to Librato CTO and co-founder Joseph Ruscio, “modern IT environments are incredibly dynamic and their operators require sophisticated alerting capabilities”. He believes effective monitoring can be the solution for successfully managing the ever changing IT landscape and thus many of the challenges of multi-tenancy.

## 2.2 Multi-tenant models for cloud

Multi-tenant architecture, commonly referred to as multitenancy, is a software architecture in which multiple single instances of software run on a single physical server. The server then serves multiple tenants. Additionally, multitenant architecture is used to enable multiple users to use a single application, for instance a database.

Multi-tenant architecture is often used in [cloud computing](https://www.datamation.com/cloud/what-is-cloud-computing/), to offer shared tenancy on [public cloud](https://www.datamation.com/cloud/public-cloud-computing-providers/) providers like [Amazon Web Services](https://www.datamation.com/cloud/amazon-web-services/), [Microsoft Azure](https://www.datamation.com/cloud/azure-cloud-migration-services-product-overview-and-insight/) and [Google Cloud](https://www.datamation.com/cloud/google-cloud-platform/). Additionally, multitenancy is a key part of another cloud model, [software as a service](https://www.datamation.com/cloud/what-is-software-as-a-service-saas/), and so is deployed by many [SaaS companies](https://www.datamation.com/cloud/saas-companies/) as well as virtually every [cloud company](https://www.datamation.com/cloud/top-16-cloud-computing-companies-2020/).

“Tenants” is a term for a group of users or software applications that all share access to the hardware through the underlying software. Multiple tenants on a server all share the memory, which is dynamically allocated and cleaned up as needed. They also share access to system resources, such as the network controller.

This is the opposite of single-tenancy, in which the server runs one instance of the operating system and one application. This one application could be something simple like file and print apps, complex like Web or application servers, or mission-critical, such as a database.

## Single-Tenant Vs. Multi-Tenant

Single-tenancy is largely seen as the “deluxe” option, in that a client operates in a solely dedicated environment. In a multi-tenant environment, each customer shares the software application along with a single database, so multiple people from the same company can access the database. Still, even in multi-tenant, each tenant is isolated from other tenants.

### i. Advantages Of Single-Tenant Architecture

Single-tenant hosting gives clients more control because there is no sharing of resources. This manifests in a number of ways:

· **Greater customization:** Since they have only one client, single tenants can customize the software for their needs, whereas multi-tenant tends to be one-size-fits-all.

· **Greater isolation from security risks:** You control the environment and (hopefully) what goes in and out of it.

· **Faster recovery:** Restoring one client is faster and easier than many.

· **Better control:** Single-tenant can be choosier about accepting software changes and updates and decide what add-ons they want to use.

· **Avoiding “noisy neighbor” syndrome:** Since you are sharing resources in multi-tenant scenarios, someone else who is really heavily using the system might slow you down.

### ii. Disadvantages Of Single-Tenant Architecture

Though some companies prefer it, there are downsides to single-tenancy.

· **Cost:** There is no cost sharing for things like balancing, services, system monitoring, and deployment.

· **Client responsibility:** Clients are responsible for software updates, patches, backup, restore, and disaster recovery.

· **Less efficient:** Single-tenant systems can be less efficient if they don’t run at full capacity or if they are over-provisioned.

### iii. Advantages Of Multi-Tenant Architecture

The chief advantage of multi-tenant hosting is that it is less expensive. The resource pooling greatly reduces the cost since you only pay for what you need. And since multi-tenant is part of a SaaS provider, you are not paying for on-premises hardware. Functions like system monitoring and servicing the deployment become shared among all of the customers, which makes it less expensive as the cost is spread around.

There are other advantages as well.

· **Simplified hosting:** It’s not your hardware to manage any more, reducing a lot of time and expense.

· **Better protection of systems:** With less interaction with the outside world, exposure to malicious software is reduced.

· **Upgrading software is no longer your problem:** You always get the latest version of software pushed out to you by the provider.

### iv. Disadvantages Of Multi-Tenant Architecture

Despite its cost advantage, multi-tenant environments have some downsides.

· **They have their own security risks:** For starters, you need strict authentication and access controls to make sure the right people get access. Second, data corruption could possibly spread from one user to all, though precautions guard against this.

· **Downtime:** Outages can be nationwide, and often make the news when they happen. SaaS providers tend to build enough redundancy into the system to minimize this.

· **Noisy neighbors:** As mentioned earlier, someone else on your CPU might be consuming cycles and slowing you down. Capacity is supposed to be elastic and expand as needed but that’s not always the case.

## 2.3 Introduction to cloud security

**MULTI TENANCY CLOUD SECURITY**

Software providers will naturally argue that their software is protected with the highest level of security available and that a company’s data is more secure than ever on their servers. Nonetheless, there is a scope for human error, where a database administrator accidentally grants access to an unauthorized person or contravenes the security policy of an organisation.

## 2.4 Cloud security Issues

## Multi-tenancy issues in cloud computing:

Multi-tenancy issues in cloud computing are a growing concern, especially as the industry expands. And big business enterprises have shifted their workload to the cloud. Cloud computing provides different services on the internet. Including giving users access to resources via the internet, such as servers and databases.

Cloud computing lets you work remotely with networking and software.

There is no need to be at a specific place to store data. Information or data can be available on the internet. One can work from wherever he wants. Cloud computing brings many benefits for its users or tenants, like flexibility and scalability. Tenants can expand and shrink their resources according to the needs of their workload. Tenants or users do not need to worry about the maintenance of the cloud.

Tenants need to pay for only the services they use. Still, there are some multi-tenancy issues in cloud computing that you must look out for:

**Security:** This is one of the most challenging and risky issues in multi-tenancy cloud computing. There is always a risk of data loss, data theft, and hacking. The database administrator can grant access to an unauthorized person accidentally. Despite software and cloud computing companies saying that [client data is safer](https://rivalime.com/web-application-security-testing-checklist/) than ever on their servers, there are still security risks.

There is a [potential for security](https://www.eescorporation.com/best-cloud-security-tools/) threats when information is stored using remote servers and accessed via the internet. There is always a risk of hacking with cloud computing. No matter how secure encryption is, someone can always decrypt it with the proper knowledge. A hacker getting access to a multitenant cloud system can gather data from many businesses and use it to his advantage. Businesses need high-level trust when putting data on remote servers and using resources provided by the cloud company to run the software.

The multi-tenancy model has many new security challenges and vulnerabilities. These new security challenges and vulnerabilities require new techniques and solutions. For example, a tenant gaining access to someone else’s data and it’s returned to the wrong tenant, or a tenant affecting another in terms of resource sharing.

**Performance:** SaaS applications are at different places, and it affects the response time. SaaS applications usually take longer to respond and are much slower than server applications. This slowness affects the overall performance of the systems and makes them less efficient. In the competitive and growing world of cloud computing, lack of performance pushes the cloud service providers down. It is significant for multi-tenancy cloud service providers to enhance their performance.

**Less Powerful:** Many cloud services run on web 2.0, with new user interfaces and the latest templates, but they lack many essential features. Without the necessary and adequate features, multi-tenancy cloud computing services can be a nuisance for clients.

**Noisy Neighbor Effect:** If a tenant uses a lot of the computing resources, other tenants may suffer because of their low computing power. However, this is a rare case and only happens if the cloud architecture and infrastructure are inappropriate.

**Interoperability:** users remain restricted by their cloud service providers. Users can not go beyond the limitations set by the cloud service providers to optimize their systems. For example, users can not interact with other vendors and service providers and can’t even communicate with the local applications.

This prohibits the users from optimizing their system by integrating with other service providers and local applications. Organizations can not even integrate with their existing systems like the on-premise data centers.

**Monitoring:** constant monitoring is vital for cloud service providers to check if there is an issue in the multi-tenancy cloud system. Multi-tenancy cloud systems require continuous monitoring, as computing resources get shared with many users simultaneously. If any problem arises, it must get solved immediately not to disturb the system’s efficiency.

However, monitoring a multi-tenancy cloud system is very difficult as it is tough to find flaws in the system and adjust accordingly.

**Capacity Optimization:** Before giving users access, database administrators must know which tenant to place on what network. The tools applied should be modern and latest that offer the correct allocation of tenants. Capacity must get generated, or else the multi-tenancy cloud system will have increased costs. As the demands keep on changing, multi-tenancy cloud systems must keep on upgrading and providing sufficient capacity in the cloud system.

## 2.5 Threat Model

Threat modeling serves to identify threats and preventive measures for a system or application. However, threat modeling is one security methodology that has not matched the general rate of cloud adoption, due to a gap in guidance, expertise, and applicability of the practice. Threat modeling for cloud systems expands on standard threat modeling to account for unique cloud services. It allows organizations to further security discussions and assess their security controls and mitigation decisions.

This document from the [Top Threats Working Group](https://cloudsecurityalliance.org/research/working-groups/top-threats/) attempts to bridge the gap between threat modeling and the cloud. To that end, this publication provides crucial guidance to help identify threat modeling security objectives, set the scope of assessments, decompose systems, identify threats, identify design vulnerabilities, develop mitigations and controls, and communicate a call-to-action. Central lessons include the benefits of threat modeling, the unique knowledge and considerations required when threat modeling in the cloud, and how to create a cloud threat model. Example threat modeling cards are provided and can be used by your team for a more gamified approach.

**Key Takeaways:**

· The baseline threat modeling processes taken from various standards and best practices

· The differences between standard threat modeling and cloud threat modeling

· How to create a cloud threat model from scratch

· A basic cloud threat model reference

· What should be included in a detailed security design report

· Example cloud threat modeling cards

## 2.6 Top 5 cloud security threats

## 1. Unauthorized Access to Data

It is the biggest risk to cloud security. According to a new cloud security spotlight report, 53% of respondents see unauthorized access via improper access controls and misuse of employee credentials as their biggest cloud security threat.

Unauthorized access involves individuals accessing enterprise data, networks, endpoints, devices, or applications, without having proper permissions. The good news is that poor access control can be tackled through security solutions in combinations with access management policies. Indusface’s Web Application Firewall allows blocking of access to cloud applications based on IP, countries, GEO location, and many more. It provides complete tracking, monitoring as well as reporting of app access, enabling enterprises to comply with data security regulations.

**Tips to prevent Poor Access Management**

* Develop a data governance framework for all user accounts. All user accounts should be connected directly to the central directory services like Active Directory that can monitor and revoke access privileges.
* You can use third-party security tools to regularly pull lists of users, privileges, groups, and roles from cloud service environments. Then your security team can sort and analyze it.
* You should also keep logging and event monitoring mechanisms in place to detect unauthorized changes and unusual activity.

## 2. Distributed Denial of Service (DDoS) Attacks

Another most common form of attack on the cloud, which proves extremely damaging. DDoS (Distributed Denial of Attack) is a kind of attack, which involves denying access to online service for legitimate users by flooding them with malicious connection requests.

**Tips to tackle DDoS attacks in the cloud**

* Have an excess of bandwidth on your enterprise’s internet connection. The more bandwidth you have, the more hackers must do to flood its connection.
* Discover vulnerabilities in your system – scan your network and system to determine vulnerabilities with [web application scanning](https://www.indusface.com/web-application-scanning.php) tools to find vulnerabilities, which can be exploited to execute DDoS attacks. Implement security controls to fix the detected security issues.
* Keep a backup internet connection – a backup connection with a separate pool of IP addresses provides an alternate path in case the primary circuit is flooded with requests.
* Configure WAF rules to filter out the malicious IPs – Configure your WAF firewall with custom rules to monitor and filter out traffic based on your requirements.

## 3. Cloud Misconfiguration

Three-quarters of all enterprises on the cloud are suffering from some sort of cloud misconfiguration, which affects security. Common weaknesses include default passwords, inadequate access restrictions, mismanaged permission controls, inactive data encryption, and many more. Many of these vulnerabilities result from insider threats and a lack of security awareness.

Another way company introduces vulnerabilities is by attempting to personalize their cloud usage by setting changes or plug-ins. These ad-hoc changes can cause configuration drift, which creates availability, management, and security problems.

**Tips to Overcome Cloud Misconfiguration Error**

* Get to know your Cloud – Learn all the services, settings, and permissions of your cloud services, and never forgot to leverage the benefits of integrated security features.
* Modify credentials and permissions – thoroughly check the default credentials and set up multi-factor authentication to ensure an extra layer of security.
* Regularly audit your cloud asset – Don’t assume that properly configured cloud settings will remain the same for a long time. Proper auditing and monitoring can help you to identify the sign of misconfigurations.
* Choose the right security solutions – The [best cloud security service providers](https://www.indusface.com/) like Indusface can provide a complete package of features, which includes security management, threat detection, and intrusion prevention.

## 4. Data Leaks and Data Breaches

The largest and critical cloud computing threat for organizations today is the loss of personal and sensitive information and data – both inadvertently and deliberately. The risk of data breaches increases as more companies allow their employees to use personal devices for work without implementing a robust security policy in place. Using personal devices to access storage services like One Drive or Dropbox increases security risks, especially when older OS versions are used. Another way in which sensitive information can be leaked is due to insider threats. Storing sensitive data and passwords in a plain text file can mean it is susceptible if the attackers get their hands on it.

Especially this is high risks in the cloud since it is a shared environment, a single vulnerability on the cloud opens the whole environment to be compromised leading to data breaches and loss.

**Tips to avoid the risk of Data Leaks**

* Encrypt Data – Sensitive data should not be in your cloud environment without being encrypted.
* Change Password – Store all your passwords in a safer place. Be smarter while you choose a password and increase the frequency in which they’ve changed.
* Set Permissions – Not all employees need the same level of access to your sensitive files. Assign permissions based on a ‘need to know’ basis to prevent the wrong people from accessing.
* Educate your staff – Train your staff to prevent them from inadvertently leaking sensitive information.

## 5. Insecure API

The adoption of APIs is advantageous for businesses, but it is a nightmare for the security team.

Though APIs are meant to streamline cloud computing processes, they are not always black & white. There is a gray area where APIs if left unsecured can allow hackers to exploit private details. Insufficient API security is one of the major causes of cloud data breaches. Gartner predicts that by 2022, APIs will be the most common vector used frequently in cyber-attacks.

**Best Practices for API security**

* Comprehensive authentication & authorization policies – APIs should be designed with tokens, signatures, quotas, encryption, API gateways, etc., to ensure API security.
* Web Application Firewalls – applies web-based vulnerability exploit defense to APIs in the cloud
* Choose standard API framework – relay only on APIs which are designed with security in mind. Examine its security aspects and decide whether it is secure enough to integrate 3rd party apps.

## 2.7 who is responsible for managing security

**Who manages cloud computing?**

The short answer is that you own the data you create, but the cloud service provider has ultimate control over it. This is reflected in many providers' terms of service which state that they can hold on to the data to comply with legal regulations.

**What are cloud security controls?**

Cloud security control is a set of security controls that protects cloud environments against vulnerabilities and reduces the effects of malicious attacks. A wide-ranging term, cloud security control includes all of the best practices, procedures, and guidelines that have to be implemented to secure cloud environments

## 2.8 Service License Agreements: Lifecycle and Management

**What is the life cycle of SLA?**

You use the service level agreement (SLA) lifecycle to govern a service level agreement from being initially identified, through to being activated, and, eventually, terminated when it is no longer required.

**The main elements of a good SLA:**

1.Overall objectives

The SLA should set out the overall objectives for the services to be provided. For example, if the purpose of having an external provider is to improve performance, save costs or provide access to skills and/or technologies which cannot be provided internally, then the SLA should say so. This will help the customer craft the service levels in order to meet these objectives and should leave the service provider in no doubt as to what is required and why.

2.Description of the Services

The SLA should include a detailed description of the services. Each individual service should be defined i.e. there should be a description of what the service is, where it is to be provided, to whom it is to be provided and when it is required. For example, if one of the services is the delivery of a specific report, the relevant provision of the SLA should describe the report, state what it should include, state its format (perhaps referring to a specific template), how it is to be delivered (e.g. by email), to whom, when and at what frequency (e.g. to the finance team daily by 10 am each weekday morning).

3.Performance Standards

Then, taking each individual service in turn, the customer should state the expected standards of performance. This will vary depending on the service. Using the "reporting" example referred to above, a possible service level could be 99.5%. However this has to be considered carefully. Often a customer will want performance standards at the highest level. Whilst understandable, in practice this might prove to be impossible, unnecessary or very expensive to achieve. On the other hand, the service provider may well argue that service levels should be set deliberately low in order to guarantee that the service can be provided at a competitive price. It is all a matter of judgement and the customer will need to consider each service level carefully – it is often the case that individual services will be weighted differently depending on their business importance. Performance standards for availability of an on-line service are generally high as it is vital for the customer to ensure constant availability of the service. Other individual services may be less important and the service levels for these can be set at a lower level.

The service provider and the customer will also need to set these performance standards in the context of anticipated workloads and the service levels may need to vary in the light of any changes to these workloads during the course of the contract. All this can be built into the SLA so that the cost implications of a change in workload can be factored in.

4.Compensation/Service Credits

In order for the SLA to have any "bite", failure to achieve the service levels needs to have a financial consequence for the service provider. This is most often achieved through the inclusion of a service credit regime. In essence, where the service provider fails to achieve the agreed performance standards, the service provider will pay or credit the customer an agreed amount which should act as an incentive for improved performance. These service credits can be measured in a number of ways. For example, if the 99.5% level for reporting is not achieved, the SLA could include a service credit that a specified reduction in price is given for each 0.5% shortfall in performance in each week. Alternatively, service credits can be given where there are, say, three or more failures to meet a service level in any specified period. Again, each service level needs to be looked at individually and a sensible level of credit agreed between the service provider and the customer for failure to achieve the agreed level over a specified period of time. The important thing is to ensure that the service credits are reasonable and incentivise the service provider to do better, and that they kick in early enough to make a difference.

5.Critical Failure

Service credits are useful in getting the service provider to improve its performance, but what happens when service performance falls well below the expected level? If the SLA only included a service credit regime then, unless the service provided was so bad as to constitute a material breach of the contract as a whole, the customer could find itself in the position of having to pay (albeit at a reduced rate) for an unsatisfactory overall performance. The solution is to include a right for the customer to terminate the agreement if service delivery becomes unacceptably bad. So the SLA should include a level of critical service level failure, below which the service provider has this termination right (and the right to sue for damages). For example, if service credits kick in if a service level failure has occurred twice in a particular period, the SLA could state that the customer has the right to terminate the agreement for material breach if the service level has not been achieved, say, eight times in the same period. Again, as with service credits, each service level has to be looked at individually and weighted according to business importance. With an on-line service, availability of that service is crucial so you might expect the right to terminate to arise earlier than for a failure to provide routine reports on time. In addition, the SLA could group certain service levels together for the purpose of calculating service credits and the right to terminate for critical failure; SLAs sometimes include aggregate point scoring systems for these purposes.

**Other SLA provisions:**

Whilst service levels, service credits and the right to terminate are the main provisions in a Service Level Agreement, depending on the structure of the entire agreement, an SLA can include other matters, such as the following:

1.Changes to pricing

Pricing may need to vary depending on a number of factors and the SLA may therefore include a pricing review mechanism or provisions dealing with the sharing of cost savings.

2.Contract Management

In longer term contracts, the parties will need to keep performance of the services under review. Provisions dealing with reporting, meetings, information provision and escalation procedures for disputes are sometimes included in the SLA rather than in the main body of the agreement. Unfortunately these types of provisions are often overlooked, but for a contract for services to be successful, it is vital that contract management procedures are agreed and are actually followed.

3.Change Control

Often the SLA will include a change control procedure, which sets out a mechanism for agreeing and recording changes to the agreement or to the services to be provided. In an agreement of any length or complexity, it is inevitable that changes will be made to the services (which will have a knock on effect on the service levels) and an agreed and properly implemented change control procedure is vital.

Common mistakes

Unfortunately many Service Level Agreements leave a lot to be desired and do not actually provide the protection the customer requires. Here are some of the more common mistakes:

* Don’t let the service provider draft the service level agreement. It is vital for customers to go through the process of deciding what services need to be provided, how they are to be provided, when, where and to whom.
* Do not allow the service level agreement to become a marketing document. This can happen when the service provider prepares the SLA. In such a situation the first draft SLA contains large chunks of material taken from marketing documents which do little more than provide a high level overview of the services without any detail. These statements are often not measurable and therefore difficult to enforce. Subjectivity should be kept to a minimum.
* Do not leave preparation of the Service Level Agreement until the last minute. The SLA should take priority and should be worked up at the start of the process. A last minute introduction of a strict SLA might lead your service provider to revise their pricing upwards.
* Do not aim for absolute perfection. Instead, prioritise which service levels are most important. When drafting, do not expect an unrealistically high level of service performance as this may (a) not be necessary and (b) lead to an increase in the price. There must be enough in the contract for the service provider otherwise the relationship will not work from the outset. Remember that the best outcome is a long lasting and mutually beneficial relationship with the service provider.
* Don’t overdo it! Remember that someone from the customer side needs to monitor service levels and compliance by the service provider. If you include too many detailed service levels which you cannot effectively monitor (due to a lack of people/expertise etc.) then the efficacy of the SLA is reduced.
* Don’t have service levels without a compensation regime of some sort. Include service credits: there must be an incentive for the service provider to do better.
* Don’t have overly long service level measurement periods. Contracts where performance is measured over a number of months before any right to service credits arises, may lead to the customer having to put up with sub-standard performance for a long period of time without any right to a remedy under the agreement.
* Take care if you want to base service levels with an external service provider on non-binding internal service levels. Sometimes organisations have their own internal service levels which are not legally enforceable but are used as a guideline for the expected level of service to be provided by one part of an organisation to another (e.g. the IT or HR function). Don’t be tempted just to apply these to an external service provider without consideration, particularly, of the business objectives for the services.
* Don’t lose sight of your objectives. Ensure the service levels reflect the overall objectives for the contract. If the overwhelming business objective is to save money, ensure that the service levels are focussed on achieving this.
* Always include a right to terminate for critical failure and define carefully what "critical failure" actually means. You might not want to use this right, but if service performance is poor, you will be glad that you have it in the contract.

## 2.9 Traditional approaches to SLO management

Traditionally, load balancing techniques and admission control mechanisms have been used to provide guaranteed quality of service (QoS) for hosted web applications. These mechanisms can be viewed as the first attempt towards managing the SLOs. The existing approaches for load balancing and admission control for ensuring QoS are:

**1.Load Balancing**

The objective of load balancing is to distribute the incoming requests onto a set of physical machines, each hosting a replica of an application, so that the load on the machines is equally distributed. The load balancing algorithm executes on a physical machine that interfaces with the clients.

**2.Admission Control**

The objective of load balancing is to distribute the incoming requests onto a set of physical machines, each hosting a replica of an application, so that the load on the machines is equally distributed. The load balancing algorithm executes on a physical machine that interfaces with the clients.

## 2.10 SLA Management in Cloud

SLA management of applications hosted on cloud platforms involves five phases.

1. Feasibility

2. On-boarding

3. Pre-production

4. Production

5. Termination

### i. What Is SLA Management?

Broadly defined, SLA management is the ongoing process of ensuring all provided services and processes—including the underlying contracts—are in alignment with the agreed-upon service level targets stipulated by the contract. From the creation of help desk tickets to retrospective reporting and regular customer feedback, [SLA monitoring helps to protect your business](https://www.n-able.com/blog/what-is-sla-service-level-agreement-for-msp)—and ensure your customers are satisfied.

Because strong SLAs will specify the measurement criteria for the agreed upon services and responsibilities, proper SLA management also involves remaining attentive to those metrics.

### ii. Common Service Level Agreement Metrics

The details of individual SLAs will of course vary depending on the type of services a customer requires—and the metrics used to measure how well the customer and provider are meeting their service targets will vary accordingly. SLA metrics are associated with specific SLA objectives, which are essentially the reason why each metric is important. Here are a few of the most common metrics used to measure how service provider performance and quality is meeting customer expectations:

* **Abandonment Rate:** the percentage of queued calls customers abandon while waiting for an answer.
* **Availability** or **Uptime:** also referred to as system reliability. This is usually measured by the percentage of time a device has been working, or the percentage of time that provided services are operational and accessible to the customer.
* **Average Speed of Answer (ASA):** the average amount of time required for the service desk to answer a call.
* **Business Results:** the use of key performance indicators to calculate how the contributions of service providers affect business performance.
* **Defect Rate:** the percentage of errors in deliverables. This can include everything from coding errors to missed deadlines.
* **First-call Resolution (FCR):** the percentage of incoming calls resolved without the use of a help desk callback to finish resolving the case.
* **Mean Time to Recovery (MTTR):** the time required to recover following a service outage.
* **Security:** the number of antivirus updates or patches installed. Even if an incident occurs, MSPs can demonstrate they’ve taken preventative measures.
* **Time Service Factor (TSF):** the percentage of queued calls answered within a defined time frame.
* **Turnaround Time (TAT):** the time required to resolve a specific task or issue once the service provider receives it.

## 2.11 Automated Policy based management

There are two policies namely Operational policies and Provisioning Policies on part of the on-boarding activity. The policies specify the sequence of actions to be performed under different circumstances. resources, and granting access to cloud software. By using the appropriate orchestration mechanisms, users can deploy and start using services on servers or on any cloud platforms.

Some of the parameters used to prioritize action and perform resource contention are:

Ø The SLA class to which the application belongs to.

Ø The amount of penalty associated with SLA breach.

Ø Whether the application is at the threshold of breaching the SLA.

Ø Whether the application has already breached the SLA.

Ø The number of applications belonging to the same customer that has breached SLA.

Ø The number of applications belonging to the same customer about to breach SLA.

Ø The type of action to be performed to rectify the situation.

The basic functionality of these components is described below:

1. Prioritization Engine

2. Provisioning Engine

3. Rules Engine

4. Monitoring System

5. Auditing:

6. Accounting/Billing System

## 2.12 Managing Clouds: Services and Infrastructure

**What are the levels of managed cloud infrastructure services?**

Managed cloud services include, at the infrastructure level: **Architecture guidance**. **System administration and operations (Ops)** **System monitoring, alerting, and reporting**.

# 3. Distributed File System (DFS) and Hadoop

**What is Distributed File System?**

In computing, a distributed file system (DFS) or network file system is **any file system that allows access to files from multiple hosts sharing via a computer network**. This makes it possible for multiple users on multiple machines to share files and storage resources.

**What is a distributed file system examples?**

A distributed file system (DFS) is **a file system with data stored on a server**. The data is accessed and processed as if it was stored on the local client machine. The DFS makes it convenient to share information and files among users on a network in a controlled and authorized way.

**What are the types of distributed file system?**

**Applications :**

· NFS – NFS stands for Network File System. ...

· CIFS – CIFS stands for Common Internet File System. ...

· SMB – SMB stands for Server Message Block. ...

· Hadoop – Hadoop is a group of open-source software services. ...

· NetWare – NetWare is an abandon computer network operating system developed by Novell, Inc.

**How does DFS work?**

DFS uses the **Windows Server file replication service to copy changes between replicated targets**. Users can modify files stored on one target, and the file replication service propagates the changes to the other designated targets. The service preserves the most recent change to a document or files.

**What are the requirements for DFS?**

**There are no additional hardware or software requirements** for running DFS Management or using DFS Namespaces. A namespace server is a domain controller or member server that hosts a namespace. The number of namespaces you can host on a server is determined by the operating system running on the namespace server.

## 3.1 Introduction to Distributed File System (DFS)

Distributed File System is a set of client and server services that allow an organization using Microsoft Windows servers to organize many distributed SMB file shares into a distributed file system. DFS has two components to its service: Location transparency and Redundancy. resources, and granting access to cloud software. By using the appropriate orchestration mechanisms, users can deploy and start using services on servers or on any cloud platforms.

## 3.2 Case Study HDFS

**What HDFS means?**

Hadoop Distributed File System

**Hadoop Distributed File System** (HDFS for short) is the primary data storage system under Hadoop applications. It is a distributed file system and provides high-throughput access to application data.

**What is HDFS example?**

HDFS is a **distributed file system that provides access to data across Hadoop clusters**. A cluster is a group of computers that work together. Like other Hadoop-related technologies, HDFS is a key tool that manages and supports analysis of very large volumes; petabytes and zettabytes of data.

**What is Hadoop case study?**

The Apache Hadoop software library is a framework that **allows for the distributed processing of large data sets across clusters of computers using simple programming models**. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage.

## 3.3 Hadoop components and importance of MapReduce

Cloud orchestration is the end-to-end automation of the deployment of services in a cloud environment. More specifically, it is the automated arrangement, coordination, and management of complex computer systems, middleware, and services—all of which helps to accelerate the delivery of IT services while reducing costs. It is used to manage cloud infrastructure, which supplies and assigns required cloud resources to the customer like the creation of VMs, allocation of storage capacity, management of network resources, and granting access to cloud software. By using the appropriate orchestration mechanisms, users can deploy and start using services on servers or on any cloud platforms.

## 3.4 Setting started - Amazon EMR

Step 1: Plan and configure an Amazon EMR cluster

• Prepare storage for Amazon EMR: When you use Amazon EMR, you can choose from a variety of file systems to store input data, output data, and log files. In this tutorial, you use EMRFS to store data in an S3 bucket. EMRFS is an implementation of the Hadoop file system that lets you read and write regular files to Amazon S3.

• Prepare an application with input data for Amazon EMR: The most common way to prepare an application for Amazon EMR is to upload the application and its input data to Amazon S3. Then, when you submit work to your cluster you specify the Amazon S3 locations for your script and data. In this step, you upload a sample PySpark script to your Amazon S3 bucket.

• Launch an Amazon EMR cluster: After you prepare a storage location and your application, you can launch a sample Amazon EMR cluster. In this step, you launch an Apache Spark cluster using the latest Amazon EMR release version.

Step 2: Manage your Amazon EMR cluster

• Submit work to Amazon EMR: After you launch a cluster, you can submit work to the running cluster to process and analyze data. You submit work to an Amazon EMR cluster as a step. A step is a unit of work made up of one or more actions.

• View results: After a step runs successfully, you can view its output results in your Amazon S3 output folder.

• (Optional) Connect to your running Amazon EMR cluster

Step 3: Clean up your Amazon EMR resources

• Terminate your cluster

• Delete S3 resources

Next steps: You have now launched your first Amazon EMR cluster from start to finish.

## 3.5 Amazon EMR - Plan and Configure clusters (# only for CSI)

### Plan and configure clusters:

This section explains configuration options and instructions for planning, configuring, and launching clusters using Amazon EMR. Before you launch a cluster, you make choices about your system based on the data that you're processing and your requirements for cost, speed, capacity, availability, security, and manageability. Your choices include:

· What region to run a cluster in, where and how to store data, and how to output results. See [Configure cluster location and data storage](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-cluster-location-data-storage.html).

· Whether you are running Amazon EMR clusters on Outposts or Local Zones. See [EMR clusters on AWS Outposts](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-outposts.html) or [EMR clusters on AWS Local Zones](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-localzones.html).

· Whether a cluster is long-running or transient, and what software it runs. See [Configuring a cluster to continue or terminate after step execution](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-longrunning-transient.html) and [Configure cluster software](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-software.html).

· Whether a cluster has a single master node or three master nodes. See [Plan and configure master nodes](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-ha.html).

· The hardware and networking options that optimize cost, performance, and availability for your application. See [Configure cluster hardware and networking](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-instances.html).

· How to set up clusters so you can manage them more easily, and monitor activity, performance, and health. See [Configure cluster logging and debugging](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-debugging.html) and [Tag clusters](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-tags.html).

· How to authenticate and authorize access to cluster resources, and how to encrypt data. See [Security in Amazon EMR](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-security.html).

· How to integrate with other software and services. See [Drivers and third-party application integration](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-plan-third-party.html).

## 3.6 AMazon EMC - Manage Clusters (# only for CSI)

**What is cluster manager in EMR?**

Cluster resource management

The resource management layer is **responsible for managing cluster resources and scheduling the jobs for processing data**. ... Amazon EMR also has an agent on each node that administers YARN components, keeps the cluster healthy, and communicates with Amazon EMR.

## 3.7 Understanding MapReduce (\* Not for CSI)

**Why is MapReduce important?**

MapReduce programming **enables companies to access new sources of data**. It enables companies to operate on different types of data. It allows enterprises to access structured as well as unstructured data, and derive significant value by gaining insights from the multiple sources of data.

## 3.8 Explore word count Java program (\* Not for CSI)

In Hadoop, [MapReduce](https://dzone.com/articles/mapreduce-design-patterns-1) is a computation that decomposes large manipulation jobs into individual tasks that can be executed in parallel across a cluster of servers. The results of tasks can be joined together to compute final results.

MapReduce consists of 2 steps:

* **Map Function –** It takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (Key-Value pair).

**Example –** (Map function in Word Count)

| **Input** | Set of data | Bus, Car, bus, car, train, car, bus, car, train, bus, TRAIN,BUS, buS, caR, CAR, car, BUS, TRAIN |
| --- | --- | --- |
| **Output** | Convert into another set of data  (Key,Value) | (Bus,1), (Car,1), (bus,1), (car,1), (train,1),  (car,1), (bus,1), (car,1), (train,1), (bus,1),  (TRAIN,1),(BUS,1), (buS,1), (caR,1), (CAR,1),  (car,1), (BUS,1), (TRAIN,1) |

* **Reduce Function –** Takes the output from Map as an input and combines those data tuples into a smaller set of tuples.

**Example –** (Reduce function in Word Count)

| **Input**  **(output of Map function)** | Set of Tuples | (Bus,1), (Car,1), (bus,1), (car,1), (train,1),  (car,1), (bus,1), (car,1), (train,1), (bus,1),  (TRAIN,1),(BUS,1), (buS,1), (caR,1), (CAR,1),  (car,1), (BUS,1), (TRAIN,1) |
| --- | --- | --- |
| **Output** | Converts into smaller set of tuples | (BUS,7),  (CAR,7),  (TRAIN,4) |

## Work Flow of the Program

Workflow of MapReduce consists of 5 steps:

1. **Splitting** – The splitting parameter can be anything, e.g. splitting by space, comma, semicolon, or even by a new line (‘\n’).

2. **Mapping** – as explained above.

3. **Intermediate splitting** – the entire process in parallel on different clusters. In order to group them in “Reduce Phase” the similar KEY data should be on the same cluster.

4. **Reduce** – it is nothing but mostly group by phase.

5. **Combining** – The last phase where all the data (individual result set from each cluster) is combined together to form a result.

## Now Let’s See the Word Count Program in Java

Fortunately, we don’t have to write all of the above steps, we only need to write the splitting parameter, Map function logic, and Reduce function logic. The rest of the remaining steps will execute automatically.

Make sure that Hadoop is installed on your system with the Java SDK.

### iii. Steps

1. Open Eclipse> File > New > Java Project >( Name it – MRProgramsDemo) > Finish.

2. Right Click > New > Package ( Name it - PackageDemo) > Finish.

3. Right Click on Package > New > Class (Name it - WordCount).

4. Add Following Reference Libraries:

1. Right Click on Project > Build Path> Add External

1. */usr/lib/hadoop-0.20/***hadoop-core.jar**

2. *Usr/lib/hadoop-0.20/lib/***Commons-cli-1.2.jar**

5. Type the code.

6. Make a jar file.

7. Run the jar file.

8. Open the result.

## 3.9 MapReduce Facts (\* Not for CSI)

**MapReduce without Combiner:**

**MapReduce with Combiner:**

## How does MapReduce Work?

Generally, MapReduce consists of two (sometimes three) phases: i.e. Mapping, Combining (optional) and Reducing.

1. Mapping phase: Filters and prepares the input for the next phase that may be Combining or Reducing.

2. Reduction phase: Takes care of the aggregation and compilation of the final result.

3. Combining phase: Responsible for reduction local to the node, before sending the input to the Reducers. Combine phase optimizes performance as it minimizes the network traffic between Mapper and Reducers by sending the output to the Reducer in chunks.

Similarly, NCache MapReduce has three phases: Map, Combine, and Reduce. Only the Mapper is necessary to implement, Reducer and Combiner implementations are optional. NCache MapReduce will execute its default reducer if the user does not implement Reducer. Default reducer merges output omitted by Mapper into an array.

The Mapper, Combiner and Reducer are executed simultaneously during an NCache MapReduce task on the NCache cluster. Mapper output is individually sent to the Combiner. When Combiner’s output reaches the specified chunk size, it is then sent to the Reducer, which finalizes and persists the output.

In order to monitor the submitted task, a traceable object is provided to the user.

Number of tasks to be executed simultaneously and Mapper’s output chunk is configurable. Mapper’s output is sent to combiner or reducer once output chunk reaches the configured chunk size. See NCache Administrator’s Guide.

A typical MapReduce task has the following components:

**Mapper:** Processes the initial input and enables user to emit the output into a dictionary to be used as an input for the combiner or reducer.

**Combiner Factory:** creates and manages combiners for each key emitted into output by the mapper.

**Combiner:** Works as local reducer to the node where Mapper’s output is combined to minimize traffic between Mapper and Reducer.

**Reducer Factory:** create and manages reducers for each key emitted into output by the mapper or combiner.

**Reducer:** Processes all those intermediate key-value pairs generated by Mapper or combined by Combiner to aggregate, perform calculations or apply different operations to produce the reduced output.

**Key Filter:** Key Filter, as the name indicates, allows the user to filter cache data based on its keys before sent to the Mapper. The KeyFilter is called during Mapper phase. If it returns true, the Map will be executed on the key. If it returns false, Mapper will skip the key and move to next one from the Cache.

**TrackerTask:** This component lets you keep track of the progress of the task and its status as the task is executed. And lets you fetch the output of the task and enumerate it.

**Output:** The output is stored in-memory, on the server side. It can be enumerated using the TrackableTask instance on the client application.