3 Virtualization in Cloud Computing

Syllabus

At the end of this unit, you should be able to understand and comprehend the following syllabus topics :

- Introduction
  - Definition of Virtualization
  - Adopting Virtualization
  - Types of Virtualization
  - Virtualization Architecture and Software
  - Virtual Clustering
  - Virtualization Application
  - Pitfalls of Virtualization
  - Virtual infrastructures
  - CPU Virtualization
  - Network and Storage Virtualization
- Grid, Cloud and Virtualization
  - Virtualization in Grid
  - Virtualization in Cloud
    - Virtualization and Cloud Security
    - Virtualization and Cloud Computing
  - Anatomy of Cloud Infrastructure

.1 Introduction to Virtualization

University Question

Q. What is virtualization? **SPPU - Oct. 16 (In Sem.), 2 Marks**

Virtualization is one of the biggest breakthroughs in the history of computers. It has made cloud computing possible and has been the cornerstone of modern technological development. I briefly touched upon it in the Chapter 1. Let's review the understanding of virtualization from Chapter 1 and proceed our discussion further.

The dictionary meaning of virtual is "something that is unreal". It gives you a sense of existence but in reality does not exist of its own. Let's take an example. You might have seen a mirage – a natural occurring phenomenon in deserts or during very hot summer days where you appear to see water due to optical illusion. It does not actually exist, but it gives you an impression that it is present.

- Similarly, in computing, virtualization provides a sense of existence of computing resources in a way that may not be real.

***Definition** : Virtualization is an abstraction technique where the finer details of the hardware layout are hidden from the upper layers of computing such as an operating system or application.*

- For example, you might already know about virtual memory. The operating system uses space on the hard disk during the high demands of primary memory, RAM. To the running applications, it gives a sense that it is using the memory space in RAM but in reality, the memory address is mapped to a region on the hard disk. The application is totally unaware of how the operating system is managing its memory and where.

3.2 Core Components of Virtualization (Virtual Infrastructures)

There are some common terms and components that you must understand to make sense of virtualization technology.

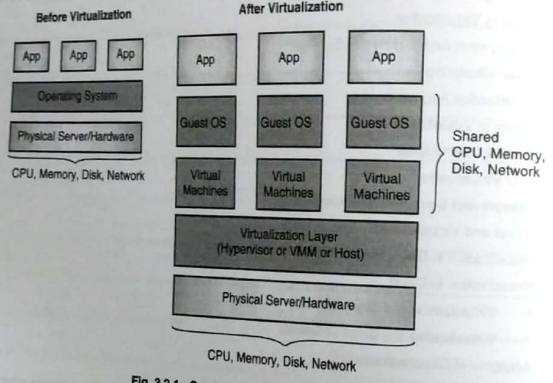


Fig. 3.2.1 : Components of Virtualization

3.2.1 Physical Server / Hardware

This is the group of real hardware resources such as CPU, memory, disk and network. These resources are actually consumed (in a shared way) by the virtualization layer and the systems running on top of it. As usual, you need to buy these resources, assemble them, maintain them and ensure that they are operational.

Relating this to cloud computing, the physical hardware is owned by the cloud service provider. It is housed in its datacentre and is maintained and operated by its administrators.

.2 Virtualization Layer

iversity Questions

What is hypervisor?

Define hypervisor.

SPPU - Aug. 15 (In sem.), 2 Marks

SPPU - Dec. 1[illegible] Marks

ight No. - L86236/2019)

- This is the layer where 'magic' happens. The virtualization layer runs software on the physical hardware and consumes the physical resources and makes them available to the systems running on top it. Which software is run at this layer, depends on the type of virtualization (as you will learn later on) that you are referring to. But, mostly it runs a software that is called hypervisor or Virtual Machine Monitor (VMM).

***Definition** : Hypervisor or **VMM** is the software that runs the virtual machines and serves as a bridge between the virtual machines and the physical hardware.*

- The hypervisor carries out several tasks on behalf of virtual machines or users. These tasks could be,
  - Passing on the instructions from the virtual machines to the actual hardware,
  - Routing network traffic,
  - Isolating one virtual machine from another,
  - Creating new virtual machines,
  - Deleting virtual machines and several others.
- There are quite a few hypervisors that are predominantly used in the industry today. They are VMware ESXi, KVM, Xen and Microsoft Hyper-V.
- Hypervisors, once installed, typically create a machine called host. In general working terms, host is the hypervisor that runs the virtual software. It is just a simple way to visualise that the hypervisor is installed on physical server.
- The Fig. 3.2.2 shows snapshot for VMware ESXi (one of the hypervisors).

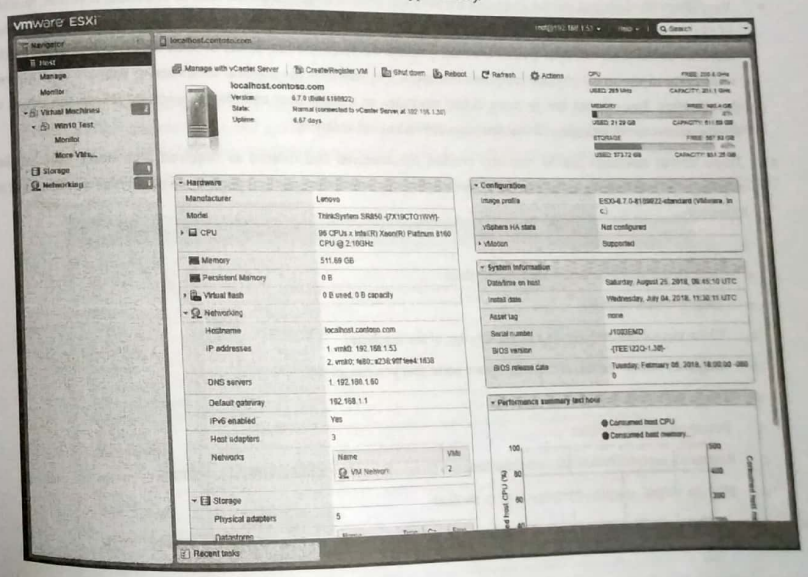


Fig. 3.2.2 : VMware ESXi

- As you see, the host typically takes all the physical hardware and logically shows that it is the 'server' that would run the virtual machines.
- Relating this to cloud computing, the virtualization layer is owned by the cloud service provider. The cloud service provider can choose to use a hypervisor based on its requirements. The Table 3.2.1 lists the hypervisors used by the various cloud service providers.

Table 3.2.1 : List of Hypervisor used by Cloud Service Provider

Cloud Service Provider	Hypervisor Used
AWS	Xen
Azure	Hyper-V
Google Cloud	KVM
Alibaba	Xen and KVM

3.2.3 Virtual Machines (VM)

***Definition** : A Virtual Machine (VM) is the artificial computer system created by the host (hypervisor).*

- Very much like how you choose a physical machine and provide its specifications such as RAM size, disk size, number of CPUs, type of network and other desired hardware features, you also need to specify the configuration of a virtual machine to the host. The host creates a virtual machine for you according to the desired specification based on the availability of the actual physical hardware resources. The host uses the physical hardware to support your desired specification. You can run one or more virtual machines on a single host and that is precisely where the power of virtualization comes into play. Virtual machines are also called as Guests.
- These virtual machines can be remotely created, administered and deleted as required. You don't have similar challenges as maintaining actual machines. Some of the common operations that you can do on virtual machines are as following :
  - Create virtual machine
  - Edit virtual machine hardware specifications
  - Delete virtual machine
  - Clone virtual machine (make a duplicate copy of the machine)
  - Snapshot virtual machine (create a restore point that you can go to)
  - Power off virtual machine
  - Power on virtual machine
  - Create a template from virtual machine that can be used to create similar virtual machines
  - Migrate virtual machine from one host to another
- The Fig. 3.2.3, snapshot gives a view of how several machines are created and managed by a single host.

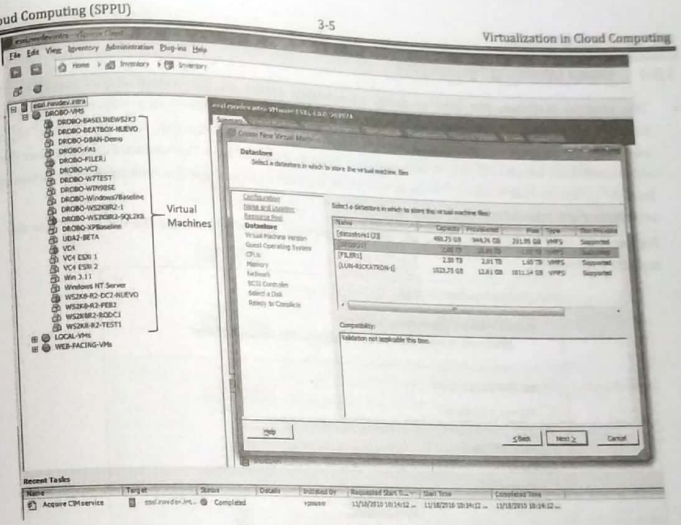


Fig. 3.2.3 : Virtual Machines on a Host

- Relating this to cloud computing, the virtual machines are created, managed and owned by the IaaS users. The cloud service provider has mechanisms to carry out several operations on such virtual machines in the cloud environment. In case of PaaS and SaaS users, the virtual machines are created, managed and owned by the cloud service providers themselves.
- For example, the Fig. 3.2.4, snapshot from AWS shows your running virtual machines (AWS calls it EC2 instances) and option to create a new one.

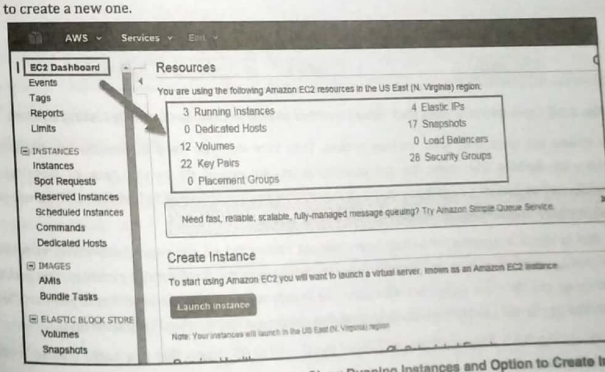


Fig. 3.2.4 : Screenshot of EC2 Service Console Show Running Instances and Option to Create Instance

3.2.4 Guest Operating System (OS)

- Like physical machines, you need to install an operating system on the created virtual machines (Guests). Consider that creating a virtual machine is as good as buying a physical machine from a vendor. It is not usable until you install the OS on it and configure it. Most of the OS vendors such as Microsoft, Red Hat, Ubuntu, etc. support virtualization and have no difficulty in installing their OS on a virtual machine. The hardware resources are presented from the virtual machine to the OS. The OS, in most cases, does not know whether it is running inside a virtual machine or a physical machine and continues to work normally.
- For example, the Fig. 3.2.5, snapshot from VMware hypervisor asks you to choose the desired OS for the virtual machine you created.

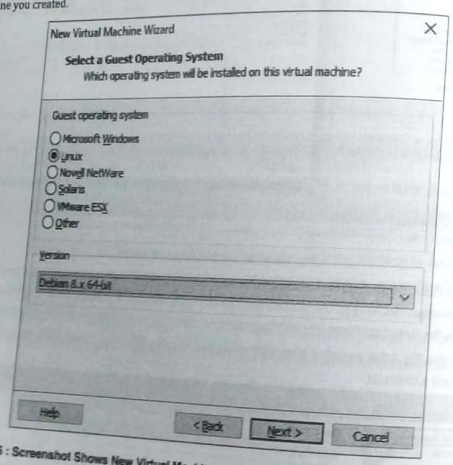


Fig. 3.2.5 : Screenshot Shows New Virtual Machine Wizard to Choose Guest Operating System

- You can choose the desired OS from various options. Note here that you need to provide installation discs and appropriate OS licenses and carry out the installation as you would do in the case of physical machines. Virtual machines just present you with the installation workflow and you still need to install the OS before you can use the virtual machine. Just creating the virtual machine is not enough.
- Relating this to cloud computing, as an IaaS user, you can choose the OS for your cloud-based virtual machines. You don't require installation discs and licenses. Those are managed by the cloud service provider. You just choose the desired OS image and the cloud service provider gives you a ready-to-use virtual machine that is pre-installed with the desired OS. You pay for the use of virtual machine as well as the license cost for the OS installed on it.
- For example, the Fig. 3.2.6, snapshot from AWS shows you the list of OS images that you might choose from for your cloud-based virtual machine.

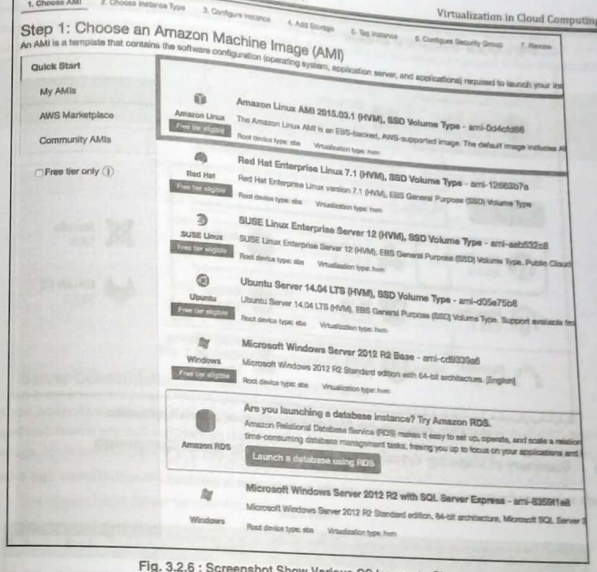


Fig. 3.2.6 : Screenshot Show Various OS Images to Choose

- Based on your selection, the cloud-based virtual machine is created with the OS.
- In the case of PaaS and SaaS, the guest OS is chosen, managed and operated by the cloud service provider along with the virtual machine required to deliver those services.

3.2.5 Applications (App)

- Finally, you run the applications on top of the OS. There could be various apps such as Excel, Word, CRM, games etc. that can run without any issues irrespective of whether the underlying environment is virtualized or not. The hypervisor takes care of everything to ensure that the underlying environment details are completely abstracted (hidden) from the app.
- Relating this to cloud computing, you can directly create app-based virtual machine instances. The OS as well as the respective app is deployed for you.
- For example, in the Fig. 3.2.7, snapshot from AWS, you can choose the OS and the app to create a virtual machine instance.

3.2.4 Guest Operating System (OS)

- Like physical machines, you need to install an operating system on the created virtual machines (Guests). Consider that creating a virtual machine is as good as buying a physical machine from a vendor. It is not usable until you install the OS on it and configure it. Most of the OS vendors such as Microsoft, Red Hat, Ubuntu, etc. support virtualization and have no difficulty in installing their OS on a virtual machine. The hardware resources are presented from the virtual machine to the OS. The OS, in most cases, does not know whether it is running inside a virtual machine or a physical machine and continues to work normally.
- For example, the Fig. 3.2.5, snapshot from VMware hypervisor asks you to choose the desired OS for the virtual machine you created.

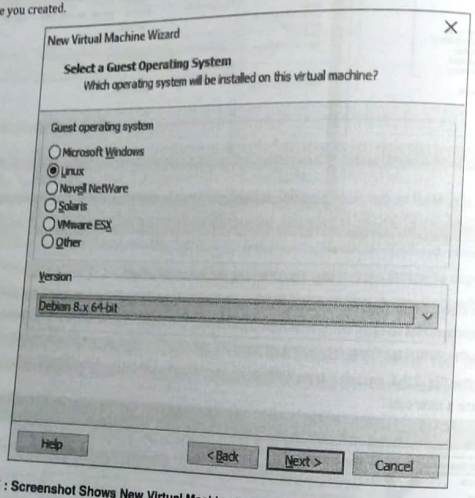


Fig. 3.2.5 : Screenshot Shows New Virtual Machine Wizard to Choose Guest Operating System

- You can choose the desired OS from various options. Note here that you need to provide installation discs and appropriate OS licenses and carry out the installation as you would do in the case of physical machines. Virtual machines just present you with the installation workflow and you still need to install the OS before you can use the virtual machine. Just creating the virtual machine is not enough.
- Relating this to cloud computing, as an IaaS user, you can choose the OS for your cloud-based virtual machines. You don't require installation discs and licenses. Those are managed by the cloud service provider. You just choose the desired OS image and the cloud service provider gives you a ready-to-use virtual machine that is pre-installed with the desired OS. You pay for the use of virtual machine as well as the license cost for the OS installed on it.
- For example, the Fig. 3.2.6, snapshot from AWS shows you the list of OS images that you might choose from for your cloud-based virtual machine.

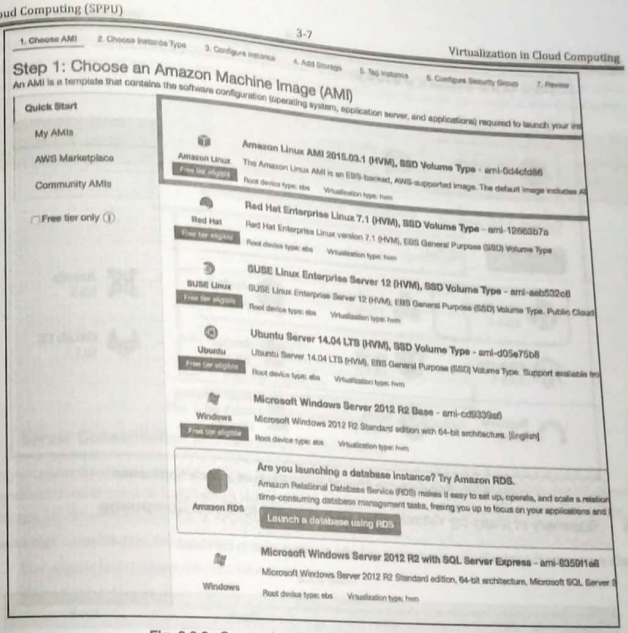


Fig. 3.2.6 : Screenshot Show Various OS Images to Choose

- Based on your selection, the cloud-based virtual machine is created with the OS.
- In the case of PaaS and SaaS, the guest OS is chosen, managed and operated by the cloud service provider along with the virtual machine required to deliver those services.

3.2.5 Applications (App)

- Finally, you run the applications on top of the OS. There could be various apps such as Excel, Word, CRM, games etc. that can run without any issues irrespective of whether the underlying environment is virtualized or not. The hypervisor takes care of everything to ensure that the underlying environment details are completely abstracted (hidden) from the app.
- Relating this to cloud computing, you can directly create app-based virtual machine instances. The OS as well as the respective app is deployed for you.
- For example, in the Fig. 3.2.7, snapshot from AWS, you can choose the OS and the app to create a virtual machine instance.

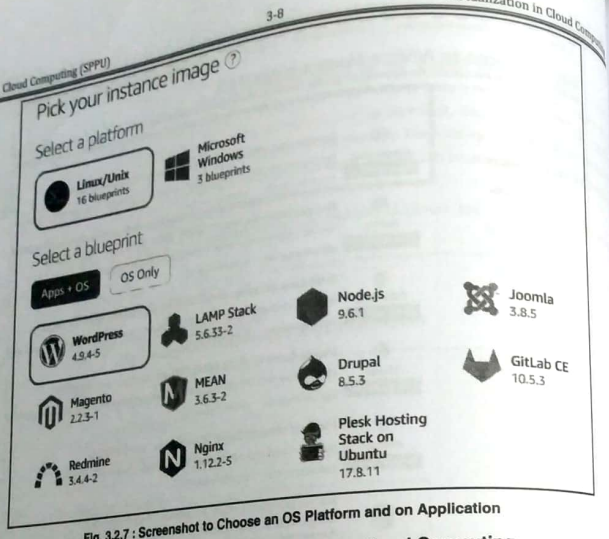


Fig. 3.2.7 : Screenshot to Choose an OS Platform and on Application

3.2.6 Summary of Mapping Virtualization Components to Cloud Computing

Let's summarise the virtualization components and who manages what in the cloud.

Table 3.2.2

Virtualization Component	IaaS	PaaS	SaaS
Physical Hardware	Cloud Service Provider	Cloud Service Provider	Cloud Service Provider
Virtualization Layer	Cloud Service Provider	Cloud Service Provider	Cloud Service Provider
Virtual Machine	User	Cloud Service Provider	Cloud Service Provider
Guest OS	User	Cloud Service Provider	Cloud Service Provider
Application	User	User	Cloud Service Provider

3.3 Advantages /Needs / Applications / Goals of Virtualization (Adopting Virtualization)

University Questions

Q. Explain the need of virtualization. SPPU - Aug. 15 (In Sem.), 4 Marks

Q. Explain applications of virtualization. SPPU - Dec. 15, Oct. 16 (In Sem.), 4 Marks

Virtualization offers several advantages over traditional way of managing physical machines. It is widely used and is the industry accepted way of running modern workloads. Let's understand what the major advantages is that virtualization provides and what are some of the needs and goals that it accomplishes.

Note : I have listed the core and different advantages that the virtualization provides instead of making a long list of interrelated points. I personally don't like repetitions. Also, VMware has published a report at *https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/infographic/vmw-top5-reasons-infographic.pdf* that we would refer to for quantifying the benefits of virtualization.

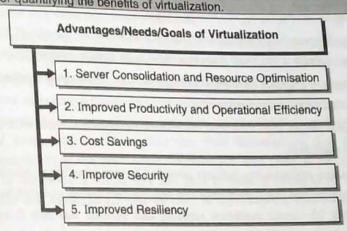


Fig. 3.3.1 : Advantages/Needs/Goals of Virtualization

3.3.1 Server Consolidation and Resource Optimisation

- As you understand, you can run several virtual machines on one host. Virtualization allows you to consolidate (combine) several physical machines on lesser number of physical servers. As per the report mentioned earlier, on an average, 16 virtual machines run on each physical server.
- Such a high consolidation ratio means,
  - You require lesser power for your datacentre.
  - You require lesser physical space to place your machines.
  - You require lesser cooling (which in turn saves more power) and is better for environment.
  - You require lesser number of physical machine purchases.
  - You require lesser number of skilled staffs to manage the physical machines.
- Server consolidation also results in resource optimisation.
- For example, take a look at the performance chart of your computer (If you are on a Windows® machine, you can go to the task manager or if you are on a Linux machine you can use the *top* command). Does it always use 100% of CPU and RAM all the time? Perhaps no. While buying the computer, I am sure you would have tried to buy the best hardware suiting your budget. But, as you see, you don't even use perhaps 50% of it all the time. So, why waste the resources and not utilise them fully? Virtualization helps you to run several machines on the same hardware and thus maximally utilise the hardware without wasting CPU cycles or RAM or other hardware resources.
- Relating this to cloud computing, the cloud service provider enjoys massive economies of scale (buying in bulk at cheap). It invests in the best possible hardware at the cheapest possible price and then uses that hardware to provide services to multiple tenants using it to the maximum. The spare (extra) hardware capacity is also sold out at discounted rates to ensure maximum hardware utilisation and high return on investment.

3.3.2 Improved Productivity and Operational Efficiency

- As per the report mentioned earlier, virtualization can typically provide productivity improvement of up to 67%.
- With virtualization,
  - You don't need to physically handle every machine. All operations can be remotely done.
  - You can quickly create and delete machines as per your requirement. Consider that with respect to
    physical machine and then deploying and configuring it.
  - You can program and automate various operations such as power on, power off, taking regular backup, or
    the virtual machines quickly if you need more machines with similar configuration and data.
  - You don't need to handle multiple hardware systems such as networking equipment, networking cables,
    supply, disks, monitors, keyboards, mouse, etc.
  - You don't need several skilled staff. One staff can handle multiple machines with automation and gaining
    control of the virtual machines.
  - You can also choose other optimisation features provided by virtualization products such as workload
    where the virtual machine load is automatically and equally distributed amongst hosts in an environment.
    example, if a host has 20 virtual machines running and another one has 5 virtual machines running
    automated workload placement feature can move a few machines from the first host to the second host to
    distribute the load.
- Such a high productivity enhancement has made it obvious that virtualization is the way to handle modern
  Relating this to cloud computing, virtualization has enabled cloud service providers as well as consumers to
  several tasks such as automatically scaling out and scaling in based on demands. The improved productivity
  operational efficiency have enabled the better utilisation of skilled workforce and focusing only on key aspects
  managing workloads while automating several other repetitive tasks.

3.3.3 Cost Savings

- By now, it must have been obvious that virtualization has the potential for huge cost savings with respect to
  a datacentre that relies only on physical machines. As per the report mentioned earlier, virtualization can
  to 50% cost savings compared to the traditional way of managing datacentres with physical machines.
- The major cost benefits arise out of the following changes that the virtualization brings in,
  - Savings from lesser number of machines to purchase,
  - Savings from lesser physical space required,
  - Savings from lesser power consumption,
  - Savings from fewer skilled staffs required,
  - Savings from better utilisation of hardware,
  - Savings from lesser business downtimes and higher resiliency (ability to recover quickly).
- All these smaller savings add up to give a significant saving over traditional datacentre management approach.
- Relating this to cloud computing, the cloud service providers mostly use standard hardware, across their
  that they buy in bulk. They pass on the cost benefits to users arising out of using virtualization to
  hardware and run workloads.

3.3.4 Improved Security

- Virtualization is a proven technology that provides improved security. The security improvements come from the following changes that the virtualization brings in.
  - You can enforce the security policies uniformly across all the virtual machines.
  - You can automate security tasks such as placing the virtual machines on a given network.
  - You can enforce hardened security baselines (OS and App configured with the right security settings) by allowing virtual machine creation only from approved templates.
  - You can provide granular access control to the datacentre environment. For example, a network administrator may not be allowed to configure storage or virtual machines. Similarly, a virtual machine administrator may not be allowed to login to the OS running inside the virtual machine.
  - You can quickly prove compliance during security audits by demonstrating adequate security controls placed in the virtualized environment. For example, you can create separate virtual machine clusters for development and testing and actual production workload. You can then demonstrate that the development and testing environment is separate from the production environment and there is no network path between the two separate clusters. In the physical world, you will have to make a listing of all the physical machines that run production workload and demonstrate that for each of them there is no network path from development and testing machines.
- Such security improvements enable you focus on key aspects of your business without spending too much of time in managing datacentre security.
- Relating this to cloud computing, the cloud service providers provide several security mechanisms to protect your cloud-based virtual machines. Some of the controls could be firewall configuration, automating security patch installation, allowing only certain ports to be opened and managing who can access those virtual machines. The cloud service providers also have a demonstrable way to prove various compliance requirements for their virtualized environment.

3.3.5 Improved Resiliency

- Resiliency means the ability to recover after a shock or a disaster. According to the report mentioned earlier, virtualization can lead to 36% less downtime and 26% less time spent in troubleshooting.
- Virtualization provides several features that help in improving resiliency.
  - It provides ability to configure a highly available cluster that can automatically move the virtual machines from one host to another host in case of a host failure.
  - It can distribute the virtual machine load evenly amongst several hosts in an environment.
  - It provides a mechanism to configure fault tolerance on a virtual machine where a secondary virtual machine automatically picks up the load if the primary virtual machine fails.
  - You can create virtual recovery sites where the virtual machines can be migrated in case of a disaster.
  - You can automate jobs such as backup to ensure that your data is protected and can be restored in case of a disaster.
  - You can create several network paths for routing virtual machine traffic and the best network path can automatically be chosen based on the load or network congestion or link failure.
  - You can dynamically add virtual hardware such as RAM, CPU, Disk etc., as and when required, without requiring a significant downtime.

3.3.2 Improved Productivity and Operational Efficiency

- As per the report mentioned earlier, virtualization can typically provide productivity improvement of up to 67%.
- With virtualization,
  - You don't need to physically handle every machine. All operations can be remotely done.
  - You can quickly create and delete machines as per your requirement. Consider that with respect to buying a physical machine and then deploying and configuring it.
  - You can program and automate various operations such as power on, power off, taking regular backup, or cloning the virtual machines quickly if you need more machines with similar configuration and data.
  - You don't need to handle multiple hardware systems such as networking equipment, networking cables, power supply, disks, monitors, keyboards, mouse, etc.
  - You don't need several skilled staff. One staff can handle multiple machines with automation and gaining better control of the virtual machines.
  - You can also choose other optimisation features provided by virtualization products such as workload placement where the virtual machine load is automatically and equally distributed amongst hosts in an environment. So, for example, if a host has 20 virtual machines running and another one has 5 virtual machines running, the automated workload placement feature can move a few machines from the first host to the second host to equally distribute the load.
- Such a high productivity enhancement has made it obvious that virtualization is the way to handle modern workloads. Relating this to cloud computing, virtualization has enabled cloud service providers as well as consumers to automate several tasks such as automatically scaling out and scaling in based on demands. The improved productivity and operational efficiency have enabled the better utilisation of skilled workforce and focusing only on key aspects of managing workloads while automating several other repetitive tasks.

3.3.3 Cost Savings

- By now, it must have been obvious that virtualization has the potential for huge cost savings with respect to managing a datacentre that relies only on physical machines. As per the report mentioned earlier, virtualization can provide up to 50% cost savings compared to the traditional way of managing datacentres with physical machines.
- The major cost benefits arise out of the following changes that the virtualization brings in,
  - Savings from lesser number of machines to purchase,
  - Savings from lesser physical space required,
  - Savings from lesser power consumption,
  - Savings from fewer skilled staffs required,
  - Savings from better utilisation of hardware,
  - Savings from lesser business downtimes and higher resiliency (ability to recover quickly).
- All these smaller savings add up to give a significant saving over traditional datacentre management approach.
- Relating this to cloud computing, the cloud service providers mostly use standard hardware, across their datacentres that they buy in bulk. They pass on the cost benefits to users arising out of using virtualization to manage the hardware and run workloads.

3.3.4 Improved Security

- Virtualization is a proven technology that provides improved security. The security improvements come from the following changes that the virtualization brings in.
  - You can enforce the security policies uniformly across all the virtual machines.
  - You can automate security tasks such as placing the virtual machines on a given network.
  - You can enforce hardened security baselines (OS and App configured with the right security settings) by allowing virtual machine creation only from approved templates.
  - You can provide granular access control to the datacentre environment. For example, a network administrator may not be allowed to configure storage or virtual machines. Similarly, a virtual machine administrator may not be allowed to login to the OS running inside the virtual machine.
  - You can quickly prove compliance during security audits by demonstrating adequate security controls placed in the virtualized environment. For example, you can create separate virtual machine clusters for development and testing and actual production workload. You can then demonstrate that the development and testing environment is separate from the production environment and there is no network path between the two separate clusters. In the physical world, you will have to make a listing of all the physical machines that run production workload and demonstrate that for each of them there is no network path from development and testing machines.
- Such security improvements enable you focus on key aspects of your business without spending too much of time in managing datacentre security.
- Relating this to cloud computing, the cloud service providers provide several security mechanisms to protect your cloud-based virtual machines. Some of the controls could be firewall configuration, automating security patch installation, allowing only certain ports to be opened and managing who can access those virtual machines. The cloud service providers also have a demonstrable way to prove various compliance requirements for their virtualized environment.

3.3.5 Improved Resiliency

- Resiliency means the ability to recover after a shock or a disaster. According to the report mentioned earlier, virtualization can lead to 36% less downtime and 26% less time spent in troubleshooting.
- Virtualization provides several features that help in improving resiliency.
  - It provides ability to configure a highly available cluster that can automatically move the virtual machines from one host to another host in case of a host failure.
  - It can distribute the virtual machine load evenly amongst several hosts in an environment.
  - It provides a mechanism to configure fault tolerance on a virtual machine where a secondary virtual machine automatically picks up the load if the primary virtual machine fails.
  - You can create virtual recovery sites where the virtual machines can be migrated in case of a disaster.
  - You can automate jobs such as backup to ensure that your data is protected and can be restored in case of a disaster.
  - You can create several network paths for routing virtual machine traffic and the best network path can automatically be chosen based on the load or network congestion or link failure.
  - You can dynamically add virtual hardware such as RAM, CPU, Disk etc., as and when required, without requiring a significant downtime.

- Several such high resiliency features are either absent from the traditional datacentre or involve complex, not expansive solutions. Virtualization provides such features seamlessly and improves the overall resiliency of your datacentre.
- Relating this to cloud computing, the cloud service providers provide a robust environment for running your workloads. Several of such virtualization features are automated to ensure service uptime and maximum availability for your cloud environment.

3.4 Challenges / Limitations / Pitfalls of Virtualization

University Question

Q. Discuss undesirable effects of virtualization. SPPU - Oct. 16 (In Sem.) 8 Marks

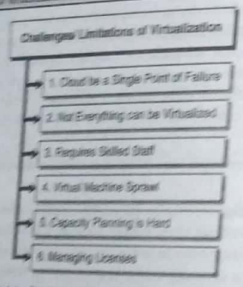


Fig. 3.4.1 : Challenges/ Limitation of Virtualization

Note : Despite any challenges or limitations of virtualization, it is widely used and is the industry-accepted way of running modern workloads. The points highlighted here are for your general reference only and they do not really impact the adoption of the virtualization technology and its usage. So, just know about what challenges could possibly be with virtualization, but do not get overly paranoid about any.

4.1 Could be a Single Point of Failure

Several physical machines could be consolidated on a host as virtual machines. But, if this host was to go down for any reason, you are likely to lose access to the virtual machines hosted on it. The host could be a single point of failure that could bring down your virtual machines along with it. As a precaution, it is recommended to have multiple hosts and some additional capacity reserved for any unforeseen issues. The virtual machines can then be migrated to other hosts as and when desired either manually or automatically.

Not Everything can be Virtualized

Some applications are hardware dependent and require specific hardware specification to be present for running or accessing them. For example, firewall applications might use ASICs (Application-Specific Integrated Circuits) for controlling the malicious traffic. Similarly, there could be other applications, such as a USB flashing software or the network or Bluetooth dongle based application that might require attaching a physical device to the machine for using it. Such dependency on the hardware may prevent the applications from working in a virtualized environment.

- There might also be extreme performance requirements that may not be met due to any overhead in the virtualized environments. These could be gaming applications, drawing and architecture applications or other applications requiring high performance from the hardware.
- Some application vendors also put installation restrictions in virtualized environments. The licenses are difficult to consume and account for in the virtualized environments.
- Hence, it might be better to run these applications on a physical machine.

3.4.3 Requires Skilled Staff

- The virtualization technology has evolved over several years and with cloud computing around, it is further changing. You would require training your staff to acquire newer skills to adopt new features and to better manage the datacentres. The virtual infrastructure administrator might need to have at least a basic understanding and hands-on-experience of the following :
  - Managing storage area network (SAN),
  - Managing networking for virtualized environments,
  - Installation of Guest Oss,
  - Provisioning (creating and configuring) of hosts and virtual machines,
  - Patching and upgrades,
  - Managing appropriate security controls,
- Traditional datacentre administrators may not be aware of all the areas and may require training.

3.4.4 Virtual Machine Sprawl

- The dictionary meaning of sprawl is "to spread out carelessly". Creating new virtual machines in the virtualized environment is very easy as compared to acquiring a physical machine and configuring it. Hence, the users might be careless about creating new virtual machines and keeping the older ones running unnecessarily. You might end up having a significantly high number of 'unwanted' virtual machines that might consume the hardware resources and become a blocker for legitimately using the virtualized environment.
- As a precaution,
  - You may allow only selected users to create new virtual machines.
  - You may setup an approval process for creating new virtual machines.
  - You may establish a policy detailing the configuration of virtual machines that are allowed in the environment. For example, you can have a policy that no virtual machine should be configured with RAM size of greater than 8 GB.
  - You may establish a policy for controlling the lifetime of a virtual machine. For example, if a virtual machine is not used in last seven days, it is likely that it is no more required and may be queued for deleting.

3.4.5 Capacity Planning is Hard

- As you consolidate your physical environment, it might be difficult to rightly estimate the size of the virtual environment that you may need presently and in future. You might buy host machines that might not be sufficient for accommodating all your physical machines (as virtual machines) and future growth opportunities. Beyond a certain scale, you may end up in similar situation where any further expansion requires buying physical server and other physical datacentre resources such as storage.

- Several such high resiliency features are either absent from the traditional datacentre or involve complex and expensive solutions. Virtualization provides such features seamlessly and improves the overall resiliency of your datacentre.
- Relating this to cloud computing, the cloud service providers provide a robust environment for running your workloads. Several of such virtualization features are automated to ensure service uptime and maximum availability for your cloud environment.

3.4 Challenges / Limitations / Pitfalls of Virtualization

University Question

Q. Discuss undesirable effects of virtualization. SPPU - Oct. 16 (In Sem.), 8 Marks

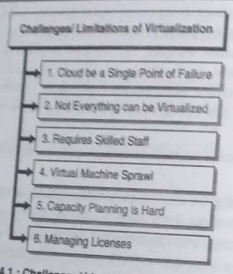


Fig. 3.4.1 : Challenges/ Limitation of Virtualization

Note : Despite any challenges or limitations of virtualization, it is widely used and is the industry-accepted way of running modern workloads. The points highlighted here are for your general reference only and they do not really impact the adoption of the virtualization technology and its usage. So, just know about what challenges could possibly be with virtualization, but do not get overly paranoid about any.

3.4.1 Could be a Single Point of Failure

Several physical machines could be consolidated on a host as virtual machines. But, if this host was to go down for any reason, you are likely to lose access to the virtual machines hosted on it. The host could be a single point of failure that could bring down your virtual machines along with it. As a precaution, it is recommended to have multiple hosts and some additional capacity reserved for any unforeseen issues. The virtual machines can then be migrated to other hosts as and when desired either manually or automatically.

3.4.2 Not Everything can be Virtualized

- Some applications are hardware dependent and require specific hardware specification to be present for running or using them. For example, firewall applications might use ASICs (Application-Specific Integrated Circuits) for controlling the malicious traffic. Similarly, there could be other applications, such as a USB flashing software or the internet or Bluetooth dongle based application that might require attaching a physical device to the machine for using them. Such dependency on the hardware may prevent the applications from working in a virtualized environment.

- There might also be extreme performance requirements that may not be met due to any overhead in the virtualized environments. These could be gaming applications, drawing and architecture applications or other applications requiring high performance from the hardware.
- Some application vendors also put installation restrictions in virtualized environments. The licenses are difficult to consume and account for in the virtualized environments.
- Hence, it might be better to run these applications on a physical machine.

3.4.3 Requires Skilled Staff

- The virtualization technology has evolved over several years and with cloud computing around, it is further changing. You would require training your staff to acquire newer skills to adopt new features and to better manage the datacentres. The virtual infrastructure administrator might need to have at least a basic understanding and hands-on-experience of the following :
  - Managing storage area network (SAN),
  - Managing networking for virtualized environments,
  - Installation of Guest Oss,
  - Provisioning (creating and configuring) of hosts and virtual machines,
  - Patching and upgrades,
  - Managing appropriate security controls,
- Traditional datacentre administrators may not be aware of all the areas and may require training.

3.4.4 Virtual Machine Sprawl

- The dictionary meaning of sprawl is "to spread out carelessly". Creating new virtual machines in the virtualized environment is very easy as compared to acquiring a physical machine and configuring it. Hence, the users might be careless about creating new virtual machines and keeping the older ones running unnecessarily. You might end up having a significantly high number of 'unwanted' virtual machines that might consume the hardware resources and become a blocker for legitimately using the virtualized environment.
- As a precaution,
  - You may allow only selected users to create new virtual machines.
  - You may setup an approval process for creating new virtual machines.
  - You may establish a policy detailing the configuration of virtual machines that are allowed in the environment. For example, you can have a policy that no virtual machine should be configured with RAM size of greater than 8 GB.
  - You may establish a policy for controlling the lifetime of a virtual machine. For example, if a virtual machine is not used in last seven days, it is likely that it is no more required and may be queued for deleting.

3.4.5 Capacity Planning is Hard

- As you consolidate your physical environment, it might be difficult to rightly estimate the size of the virtual environment that you may need presently and in future. You might buy host machines that might not be sufficient for accommodating all your physical machines (as virtual machines) and future growth opportunities. Beyond a certain scale, you may end up in similar situation where any further expansion requires buying physical server and other physical datacentre resources such as storage.

- Virtualization also lets you run over-capacity. So, for example, your host machine might just have 16 GB of RAM, but virtualization would allow you to create say 3 virtual machines with 6 GB RAM each thus totalling 18 GB which is over capacity of what the host actually has. The reason virtualization allows you to do this is because even though 6 GB of RAM is assigned to each virtual machine, each virtual machine may not be using full 6 GB RAM all the time. So, if two machines are just using say 2 GB each, the third machine can comfortably use 6 GB and stay within the total limit of 16 GB collectively on the host (2+2+6 = 10 GB < 16GB). But, such over-capacity usage might prove to be troublesome if those VMs actually require their assigned resources simultaneously during peak times. You might see performance degradation because the host would not be able to provide the optimum level of resources as required.
- As a precaution,
  - You should reserve the capacity for the virtual machines as required
  - Periodically audit your virtualized environment with respect to capacity usage and scope for growth
  - Plan for growth based on your foreseeable business requirements

3.4.6 Managing Licenses

OS and application vendors may have different license terms for virtualized environments from physical environments. If you have a mixed mode environment consisting of physical machines as well as virtual machines, then it might be hard for you to keep track of your license usage. Additionally, you might also require managing licenses for the virtualization software itself. The virtualization software license is usually granted per physical CPU socket or cores. You might also need to consider the license cost for the virtualization software when taking up or managing the virtualization environment and account it in your datacentre operations budget.

3.5 Implementation Levels of Virtualization (Types of Virtualization)

University Questions

Q. Explain different levels of virtualization. SPPU - Dec. 15, Oct. 16 (In Sem.), April 19, 4 Marks

Q. Explain different abstraction levels of virtualization. SPPU - April 18, 6 Marks

Virtualization can be implemented at the following levels :

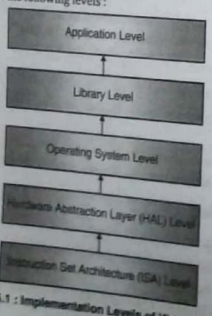


Fig. 3.5.1 : Implementation Levels of Virtualization

3.5.1 Instruction Set Architecture (ISA) Level Virtualization

- This is the bottom most level at which virtualization might be implemented. Instruction Set Architecture (ISA) refers to the type of instructions that are supported and understood by the processor family.
- Processor instructions are categorised into,
  - Arithmetic and Logical Instructions
  - Data Transfer Instructions
  - Branch and Jump Instructions
- The various processor vendors have implemented different instruction sets that are understood by only their brand of processors. For example, Intel developed the x86 and x64 architecture, ARM developed the ARM architecture, AMD developed the amd64 architecture and UC Berkeley developed an open source ISA in the RISC-V.

✍ ***Definition :*** *Instruction Set Architecture (ISA) Virtualization enables executing instructions meant for one processor architecture on another.*

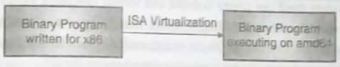


Fig. 3.5.2 : ISA Virtualization

- What this means is that virtualization at the ISA level, for example, could enable you to execute programs written for Intel x86 on amd64. Each program instruction is interpreted (read line by line) and translated into the instruction understood by the target architecture. There could be general translation overhead but at least you will be able to run the program across architectures using virtualization.
- QEMU (Quick Emulator) is one such solution that provides ISA level virtualization.

3.5.2 Hardware Abstraction Layer (HAL) Level Virtualization

- Virtualization of the hardware resources is the most common type of virtualization used in the industry today. In fact, in general, when you say virtualization, people tend to refer to hardware virtualization.

✍ ***Definition : Hardware Abstraction Layer*** *or simply hardware-level virtualization refers to the virtualization of hardware resources, such as CPU, Memory, Disk and Network, to create a virtualized platform that can run multiple virtual machines.*

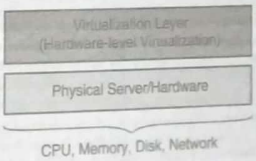


Fig. 3.5.3 : Hardware Abstraction Layer

- You have already learnt about the various components of hardware-level virtualization such as hypervisor, virtual machines and guest OS in the introduction section. Various commercial vendors, such as VMware, Nutanix, HP, provide virtualization solutions. There are also open source virtualization solutions such as Xen and KVM available.

3.5.3 Operating System Level Virtualization

Definition : Operating System Level Virtualization *provides isolated execution environments within the operating system itself.*

- OS Level virtualization has existed in the industry for quite a long time and has recently started receiving attraction again due to popularity of Docker Containers.

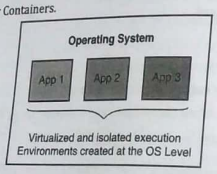


Fig. 3.5.4 : Operating System Level Virtualization

- Basically, the operating system itself can create isolated and virtual execution environments where applications can run unaware of anything else. To the application, it seems that it 'owns' the complete OS, but in reality, it is only limited to an isolated space within the OS. The OS can create multiple such isolated spaces where each application can run independently without being aware of what's running elsewhere on the OS.
- Apple's iOS, Docker Containers and Solaris Zones are examples of various OS Level virtualization.
- Just for your reference, following is an excerpt from Apple for iOS jail - https://support.apple.com/en-in/HT201954

> iOS is designed to be reliable and secure from the moment you turn on your device. Built-in security features protect against malware and viruses and help to secure access to personal information and corporate data. Unauthorized modifications to iOS (also known as "jailbreaking") bypass security features and can cause numerous issues to the hacked iPhone, iPad, or iPod touch, including:
>
> - **Security vulnerabilities:** Jailbreaking your device eliminates security layers designed to protect your personal information and your iOS device. With this security removed from your iOS device, hackers may steal your personal information, damage your device, attack your network, or introduce malware, spyware, or viruses.
> - **Instability:** Frequent and unexpected crashes of the device, crashes, and freezes of built-in apps and third-party apps, and loss of data.
> - **Shortened battery life:** The hacked software has caused an accelerated battery drain that shortens the operation of an iPhone, iPad, or iPod touch on a single battery charge.
> - **Unreliable voice and data:** Dropped calls, slow, or unreliable data connections, and delayed or inaccurate location data.
> - **Disruption of services:** Services such as iCloud, iMessage, FaceTime, Apple Pay, Visual Voicemail, Weather, and Stocks, have been disrupted or no longer work on the device. Additionally, third-party apps that use the Apple Push Notification Service have had difficulty receiving notifications or received notifications that were intended for a different hacked device. Other push-based services such as iCloud and Exchange have experienced problems synchronizing data with their respective servers.
> - **Inability to apply future software updates:** Some unauthorized modifications have caused damage to iOS that is not repairable. This can result in the hacked iPhone, iPad, or iPod touch becoming permanently inoperable when a future Apple-supplied iOS update is installed.
>
> Apple strongly cautions against installing any software that hacks iOS. It is also important to note that unauthorized modification of iOS is a violation of the iOS end-user software license agreement and because of this, Apple may deny service for an iPhone, iPad, or iPod touch that has installed any unauthorized software.

3.5.4 Library Level Virtualization

Definition : Library Level Virtualization *enables running applications across multiple platforms.*

- Any application depends upon the OS to fulfill its execution. For example, if it requires user input via keyboard or mouse, it submits such a request to the OS via system calls and the OS takes the required input and passes it to the application. The application developers program their applications considering the target OS where the application would likely run. So, for example, some applications could only run on Windows® whereas others could be Linux environment specific.

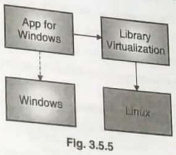
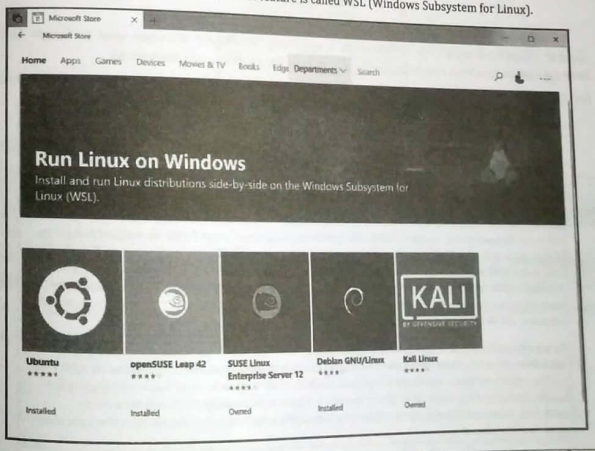


Fig. 3.5.5

- The library level virtualization allows application's system calls to be translated into the target platform. As a result, you can then run. For example, a Windows® application on a Linux system. The application requests are appropriately intercepted (captured) and the corresponding system calls are invoked depending on the target platform.
- An example of library level virtualization is Wine. It is capable of running Windows applications on several POSIX-compliant operating systems, such as Linux, mac OS, & BSD. Instead of simulating internal Windows® logic like a virtual machine or emulator, it translates the Windows® API calls into POSIX calls on-the-fly.
- Just for your reference, recently Microsoft has taken a similar approach where it allows to run Linux on a Windows® machine. It sounds unbelievable but is true! This feature is called WSL (Windows Subsystem for Linux).



3.5.5 Application Level Virtualization

Definition : Application Level Virtualization runs the application [illegible] *instead of directly running on the OS.*

- Usually, the applications are designed to run directly on the OS. In the application level virtualization approach, the application creates a virtual environment and runs via it. Let's take an example to understand it better.
- Heard of java programming language? I am sure you would have. A Java program can run on Windows® as well as Linux without requiring changes to the program or any other virtualization technology. What makes it platform independent? It is JVM (Java Virtual Machine). When you compile a java program, it generates java bytecode. The java bytecode is an intermediate format between the java program and the OS. The JVM understands the bytecode and the java program runs via the JVM without worrying about the underlying OS.
- Fig. 3.5.6 shows a high-level block diagram of JVM.

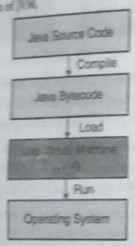


Fig. 3.5.6 : Block diagram of JVM

- Another example of application virtualization is application sandboxing technique used in Google Android OS. Each application runs in its own isolated environment without impacting other applications.
- Just for your reference, following is an excerpt from Android OS - https://source.android.com/security/app-sandbox

Application Sandbox

The Android platform takes advantage of the Linux user-based protection to identify and isolate app resources. This isolates apps from each other and protects apps and the system from malicious apps. To do this, Android assigns a unique user ID (UID) to each Android application and runs it in its own process.

Android uses this UID to set up a kernel-level Application Sandbox. The kernel enforces security between apps and the system at the process level through standard Linux facilities, such as user and group IDs that are assigned to apps. By default, apps can't interact with each other and have limited access to the operating system. For example, if application A tries to do something malicious, such as read application B's data or dial the phone without permission (which is a separate application), then the operating system protects against this behavior because application A does not have the appropriate user privileges. The sandbox is simple, auditable, and based on decades-old UNIX-style user separation of processes and file permissions.

Because the Application Sandbox is in the kernel, this security model extends to native code and to operating system applications. All of the software above the kernel, such as operating system libraries, application framework, application runtime, and all applications, run within the Application Sandbox. On some platforms, developers are constrained to a specific development framework, set of APIs, or language in order to enforce security. On Android, there are no restrictions on how an application can be written that are required to enforce security; in this respect, native code is as sandboxed as interpreted code.

3.5.6 Comparison between various Implementation Levels of Virtualization

Table 3.5.1 gives a quick comparison between the various implementation levels of virtualization.

Table 3.5.1 : Comparison between Various Implementation Levels of Virtualization

Comparison Attribute	ISA Level	Hardware Level	OS Level	Library Level	Application Level
Performance	Low	High	High	Medium	Medium
Flexibility	High	Medium	Low	Low	Low
Complexity	Medium	High	Medium	Low	High
Isolation	Medium	High	Low	Low	High
Industry Use	Low	High	High	Low	Medium

3.6 Virtualization Structures / Tools and Mechanisms (Virtualization Architecture and Software)

You have already learnt about how hardware level virtualization works. In this section, let's see types of hardware based virtualization and some other virtualization mechanisms.

3.6.1 Types of Hypervisors

University Question

Q. Explain different types of hypervisors with example. **SPPU - Aug. 15 (In sem.), April 18, May 19, 6 Marks**

There are two types of hypervisors.

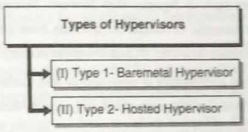


Fig. 3.6.1 : Types of Hypervisors

3.6.1(A) Type 1 : Baremetal Hypervisor

University Question

Q. Draw the diagram of Xen Architecture. **SPPU - April 19, 3 Marks**

The first type of hypervisor is called Type 1 hypervisor or baremetal hypervisor.

*Definition : In Type 1 or **Baremetal hypervisor**, the hypervisor runs directly on the underlying hardware, without a host OS.*

- This type of hypervisor does not require a separate operating system and can directly be installed on a physical server.
- Examples of Type 1 hypervisor are as following :
  - VMware ESXi
  - Microsoft Hyper-V
  - KVM
  - Xen

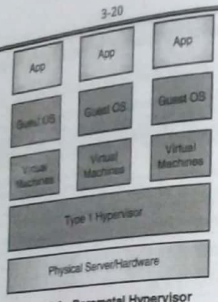


Fig. 3.6.2 : Baremetal Hypervisor

Xen

Definition : Xen (or more formally Xen Project) is an open source type 1 hypervisor that allows to run multiple virtual machines on a single host machine.

It was developed by the University of Cambridge in 2003 and is now being developed and maintained by The Linux Foundation. Various cloud providers such as Amazon and Alibaba use Xen for providing cloud based services.

Characteristics and Features of Xen

1. **Wide adoption and distribution :** The Xen Project Hypervisor is used by more than 10 million users, and powers some of the largest cloud providers such as Amazon Web Services, Tencent, Alibaba Cloud, Oracle Cloud and IBM SoftLayer. It is also the base for commercial virtualization products from Citrix, Huawei, Inspur and Oracle. It is also used for providing security solutions such as Qubes OS and Bromium vSentry.
2. **Open source and flexible :** The Xen architecture is open source and flexible allowing vendors to create Xen-based products and services for servers, cloud, desktop, embedded, and security-first environments.
3. **Support for multiple Guest OS :** The Xen hypervisor supports a wide variety of Guest OS such as Linux, Windows, FreeBSD and NetBSD.
4. **High scalability and performance :** The Xen hypervisor can scale up to 4,095 host CPUs with 16 Tb of RAM. Using Para Virtualization (PV), the hypervisor supports a maximum of 512 virtual CPUs with 512 Gb RAM per guest. Using Hardware Virtualization (HVM), it supports a maximum of 128 virtual CPUs with 1 Tb RAM per guest.
5. **Support for both Para Virtualization and Hardware Virtualization :** Xen supports running two different types of virtual machines – Para virtualization (PV) and Full or Hardware assisted Virtualization (HVM). Both virtual machine types can be used at the same time on a single Xen hypervisor system.
6. **Small size :** Xen uses a microkernel design. It has a small memory footprint and limited interface to the virtual machines. It is just around 1 MB in size.
7. **Security :** Xen is one of the safest hypervisors. Some of its major security features include
   (a) **Virtual Machine Inspection :** Using virtual machine inspection, you can build systems to protect the virtual machines against malicious intrusion and malware attacks.

(b) **KCONFIG :** KCONFIG gives developers, Xen distributors and system administrators the ability to remove core Xen Hypervisor features at compile time. This ability creates a more lightweight hypervisor and eliminates extra attack surfaces that are beneficial for high security environments and environments that have heavy compliance and certification needs such as automobiles.

(c) **Live Patching :** Live Patching enables reboot free deployment of security patches on the Xen hypervisor. This minimises any disruption and downtime during security upgrades for your virtual environment. Thus, you can install security patches immediately without having to plan for a scheduled maintenance window or downtime.

Architecture of Xen

The Fig. 3.6.3 depicts the high-level architecture of Xen.

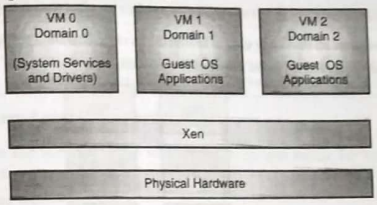


Fig. 3.6.3 : Architecture of Xen

1. **Physical Hardware :** The physical hardware is the bottom most layer that consists of the actual hardware devices such as CPU, RAM and Storage enclosed as in a baremetal server. Xen supports a wide variety of hardware.
2. **Xen hypervisor :** Xen hypervisor runs directly on the hardware and is responsible for managing CPU, memory, and other hardware components and processes. Note here that the hypervisor itself has no knowledge of the I/O functions such as networking and storage. It is very lightweight and has a footprint of less than 1 MB.
3. **Domain 0 :** In Xen's terminology a virtual machine is called a domain. Domain 0 (or the Control Domain) is a specialised virtual machine (VM 0) that has special privileges such as,
   (a) Accessing the physical hardware directly.
   (b) Handling all access to the system's I/O functions such as networking and storage.
   (c) Interacting with all other user created virtual machines running Guest OSs.

   The Xen hypervisor is not usable without Domain 0. Domain 0 which is the first VM started by the hypervisor system.

   Domain 0 (VM 0) has the following functions.

   (a) **Running system services :** It runs all the system services such as XenStore/XenBus (XS) for managing settings, the Toolstack (TS) exposing a user interface to a Xen based system, Device Emulation (DE) which is based on QEMU in Xen based systems.

   (b) **Native Device Drivers :** It contains all the required device drivers for the supported hardware running the Xen hypervisor system.

   (c) **Virtual Device Drivers :** It also contains virtual device drivers required for the user created virtual machines.

(d) **Toolstack :** It allows a user to manage virtual machine creation, destruction, and configuration. The toolstack exposes an interface that is either driven by a command line console, by a graphical interface or by a cloud orchestration stack such as OpenStack or CloudStack. Note that several different toolstacks can be used with Xen.

4. **Guest Domains / Virtual Machines :** These are user created virtual machines each running its own operating system and applications. The hypervisor supports several different virtualization modes such as para virtualization and hardware virtualization. Guest VMs are totally isolated from the hardware. They do not have any privilege to access hardware or I/O functionality. Thus, they are also called unprivileged domains. They communicate with the hardware via Domain 0 (or VM 0).

3.6.1 (B) Type 2 : Hosted Hypervisor

Definition : *In Type 2 or Hosted Hypervisor, the hypervisor runs on top of an OS.*

- This type of hypervisor requires an operating system to support its virtualization activities. The hypervisor is installed as a regular software application on the OS and provides virtualization.

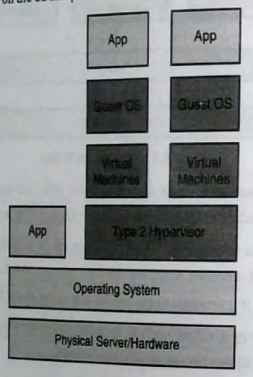


Fig. 3.6.4 : Hosted Hypervisor

- Examples of Type 2 hypervisors are as following :
  - VMware Workstation
  - VMware Fusion
  - VMware Player
  - Microsoft Virtual PC
  - Oracle VM VirtualBox

3.6.1(C) Comparison between Type 1 and Type 2 Hypervisor

University Question

Q. Compare KVM, Xen and Vmware Workstation. SPPU - May 18, May 19, 6 Marks

Table 3.6.1 provides a comparison between Type 1 and Type 2 hypervisor.

Table 3.6.1 : Comparison between Type 1 and Type 2 Hypervisor

Comparison Attribute	*Type 1 Hypervisor*	*Type 2 Hypervisor*
Separate OS required	No	Yes
Performance	Very High	High
Efficiency	Very High	Medium
Usage	Commercial Use	Personal or Limited Use
OS Security	Not applicable	Dependent on OS
Virtualization Features	Many	Few
Cost	High	Low
Examples	Xen. KVM, VMware ESXi and others	VMware Workstation, VirtualBox and others

3.6.2 Types of Hardware-Level Virtualization

- Let's dive slightly deeper and understand how virtualization technique works. But before that, take a moment to understand the general OS architecture.
- The OS typically uses a ring architecture where,
  - The most trusted instructions, such as working with the hardware directly, are executed with the highest privileges at Ring 0.
  - The least trusted instructions, such as user instructions, are executed with the least privileges at Ring 3. Users or applications are not allowed to execute privileged instructions directly. All privileged instructions must be submitted via the OS and the OS executes those instructions on the user's behalf and returns the results. Non-sensitive instructions can directly be executed without going through the OS.

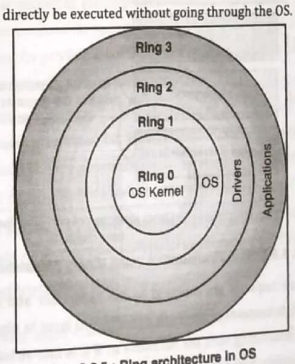


Fig. 3.6.5 : Ring architecture in OS

- The OS generally assumes that it completely owns the entire hardware and has access to Ring 0 to execute the privileged instructions that work with hardware such as CPU, memory and I/O devices such as hard disk.

- With virtualization, the guest OS does not completely own the hardware and thus cannot execute directly in Ring 0. Now, if the OS does not have access to Ring 0 then how should it access the hardware and execute sensitive and privileged instructions? Interesting problem, isn't it?
- Let's understand the three basic approaches to solve this problem.

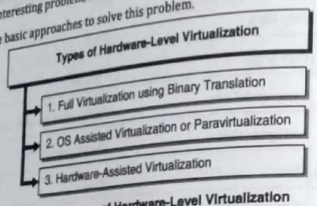


Fig. 3.6.6 : Types of Hardware-Level Virtualization

3.6.2(A) Full Virtualization using Binary Translation

University Question

Q. Explain Binary Translation with Full virtualization. **SPPU - April 19, 4 Marks**

Definition : *In the **Full Virtualization using Binary Translation** technique, the hypervisor captures and translates the privileged instructions of the guest OS to execute on the physical hardware and serves the results back to the guest OS.*

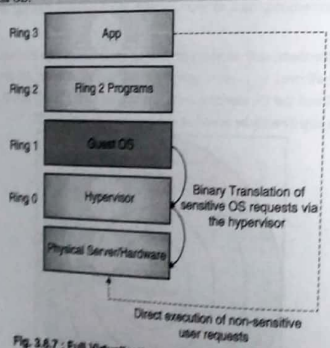


Fig. 3.6.7 : Full Virtualization using Binary Translation

- The sensitive guest OS hardware requests are captured by the hypervisor and are translated and executed at the physical hardware. The guest OS is unaware of any such virtualization layer in place between the hardware and itself and assumes that it is directly interacting with the hardware as if it is non-virtualized. The guest OS keeps sending hardware commands and hypervisor takes those commands and translate it to be executed at the hardware level seamlessly.
- Examples of virtualization software using this technique are VMware ESXi and Microsoft Virtual Server.

3.6.2(B) OS Assisted Virtualization or Paravirtualization

University Question

Q. Explain in brief about paravirtualization. **SPPU - April 18, 4 Marks**

Para is a Greek word meaning "alongside" or "with".

Definition : *In the **OS Assisted Virtualization** or Paravirtualization technique, the guest OS kernel is modified to interact with the hypervisor directly to execute the privileged commands.*

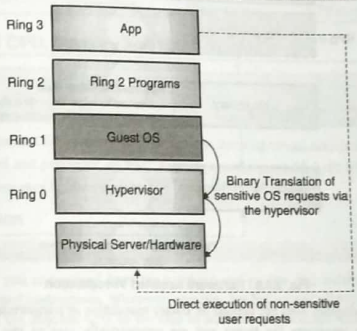


Fig. 3.6.8 : OS Assisted Virtualization

- Unlike full virtualization, in paravirtualization, the guest OS is aware that it is running as a virtual machine. So, instead of submitting system calls for executing privileged instructions, it sends the privileged instructions directly to the hypervisor and the hypervisor executes those instructions. These calls can be called as "hypercalls". There is no need for the hypervisor to watch out for privileged hardware instructions, capture them and then translate them. The guest OS directly submits any requests to the hypervisor for execution.
- This approach requires modification of the OS kernel to change its behaviour. Paravirtualization claims to be more efficient than the full virtualization approach but it varies significantly depending on the type of workload. There is no general benefit guarantee and it is not a general practice to use paravirtualized virtual machines until there is a valid reason to do so. Additionally, since paravirtualization requires OS modification, not all OSs can be paravirtualized.
- Various OS vendors such as Red Hat, Microsoft and Oracle provide paravirtualized format of their OS distributions. You can run these OSs virtually using Xen, VMware ESXi or any other paravirtualization supporting hypervisor.

3.6.2(C) Hardware-Assisted Virtualization

Hardware vendors continuously make enhancements in hardware technology to support various demands in the industry.

Definition : *In **Hardware-Assisted Virtualization** technique, hardware capabilities are used to carry out virtualization operations.*

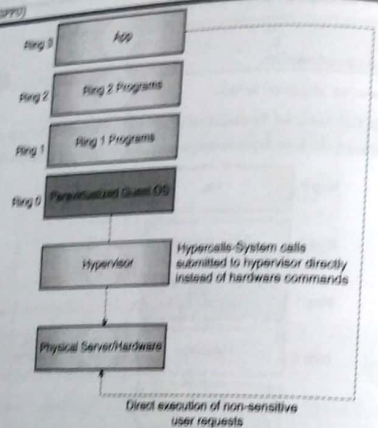


Fig. 3.6.9 : Hardware Assisted Virtualization

- Hardware-assisted virtualization does not depend on binary translation or paravirtualization. The guest OS calls are considered as hardware interrupts (or traps) and are automatically sent to the hardware for execution. The hypervisor passes through these traps to the hardware and enables execution of guest OS system calls.
- When using this assistance, the guest OS can use a separate mode of execution called guest mode. The guest code, whether application code or privileged code, runs in the guest mode. On certain events, the processor exits out of the guest mode and enters the root mode. The hypervisor executes in the root mode. It determines the reason for the exit, takes any required actions, and puts the guest in guest mode again.
- Intel Virtualization Technology (VT-x) and AMD's AMD-V are examples of hardware-assisted virtualization capability. They place the CPU in a new mode called "root mode" below Ring 0. Processors with Intel VT and AMD-V became available in 2006, so only newer OSs contains these hardware assist features.
- This technology is new and there is a significant overhead for the hypervisor to serve traps from various guest OS. It is used limitedly. Modern hypervisors allow you to expose hardware assisted virtualization to the guest OS so that the applications that require hardware virtualization can run on virtual machines without requiring binary translation or paravirtualization.

3.6.2(D) Comparison between Types of Hardware-Level Virtualization

The Table 3.6.2 provides a comparison between the types of hardware-level virtualization.

Table 3.6.2 : Comparison between Types of Hardware-Level Virtualization

Comparison Attribute	Binary Translation	Paravirtualization	Hardware Assisted
Performance	High	Sometimes better	Medium
OS modification	Not required	Required	Not Required

Comparison Attribute	Binary Translation	Paravirtualization	Hardware Assisted
Hardware Support	Not required	Not required	Required
Support for older OSs	Yes	No	No
Troubleshooting	Easy	Complex	Medium
Industry Use	Frequently	Rarely	Sometimes
Technique	Translation	Hypercalls	CPU in root mode

3.7 Virtualization of CPU, Memory and I/O Devices

University Question

Q. Describe different types of virtualization. SPPU - Aug. 15 (In Sem.), Oct. 16 (In Sem.), 4 Marks

Any computer system requires CPU, memory and I/O devices. Running virtual machines is no different. CPU, memory and I/O devices are virtualized and presented as "real" hardware to the guest OS. Let's understand some basic concepts behind the virtualization of these core hardware entities.

3.7.1 CPU Virtualization

Definition : CPU virtualization *allows multiple virtual machines to share one or more physical CPUs.*

In CPU virtualization, as you understand from the previous section, the ring architecture separates the privileged instructions from non-privileged instructions. Wherever possible, the hypervisor allows to directly run the guest OS's non-sensitive instructions on the physical CPU. In the case of sensitive instructions, it either uses binary translation to mediate the system calls or use paravirtualization to execute privileged commands through hypercalls. This is the crucial behaviour to ensure stability and integrity of the virtual environment.

Types of CPU Virtualization

There are two types of CPU virtualization.

1. **Software-Based CPU Virtualization :** Binary translation and paravirtualization based hypercalls come in this category. The hypervisor software takes care of virtualizing the CPU and separating out privileged and non-privileged instructions and executing them on the physical CPU either directly or indirectly.
2. **Hardware-Assisted CPU Virtualization :** As you have already learnt, In Hardware-Assisted Virtualization technique, hardware capabilities are used to carry out virtualization operations. Modern processors allow the privileged instructions to be executed directly without requiring binary translation or hypercalls.

3.7.2 Memory Virtualization

University Question

Q. Draw the diagram of 'Two level memory mapping' with reference to memory virtualization. SPPU - April 19, 3 Marks

Definition : Memory virtualization *allows multiple virtual machines to share the physical memory.*

The hypervisor manages the memory virtualization. The physical memory on the server is virtualized and shared amongst various virtual machines by the hypervisor. The virtual machines are provided with a contiguous (adjoining or without break) address space which may not be contiguous on the real physical memory. This address space is mapped to the actual physical memory on the server. Each virtual machine maintains its page tables that provide the mapping between the virtual page numbers to physical page numbers as assigned by the hypervisor.

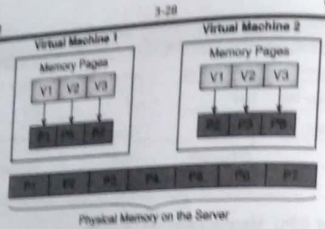


Fig. 3.7.1 : Memory Virtualization

3.7.3 I/O Device Virtualization

Definition : I/O Device virtualization allows multiple virtual machines to share the physical I/O Devices.

- Input / Output (I/O) devices are attached physically to the hypervisor or could be present on the virtualization management client software. These devices could be disks, keyboard, mouse, monitor, sound card, network card or anything else that requires the general virtual machine operations.
- If the virtual machines cannot access these devices then how will they operate? For example, how will you install the operating system on the virtual machine without having some mechanism to "insert" the OS DVD? How will you type inside the virtual machine? How will you provide disk to the virtual machine? That is the precise problem that I/O virtualization solves. It allows the shared use of physical I/O devices by virtual machines.

Similar to CPU virtualization, there are three ways to virtualize I/O.

1. **Full device emulation :** This is like binary translation. The hypervisor presents the physical I/O devices as "real" devices to the virtual machines and captures all the hardware instructions for these virtual devices to execute on the actual physical I/O devices.
2. **Paravirtualized I/O :** In this technique, a part of the I/O device driver runs within the guest OS and the other part runs on the hypervisor. The guest OS directly submits the I/O request by its I/O driver and the hypervisor executes the request.
3. **Direct I/O virtualization :** In this technique, virtual machines are allowed access to the physical I/O devices directly. It is generally used for networking in virtual machines. It does not require full device emulation or paravirtualized I/O. It depends on the hardware technology such as Intel Virtualization Technology for Directed I/O (VT-d).

3.8 Virtual Clusters and Resource Management

- The dictionary meaning of cluster is an aggregation or a group of similar things. Clusters allow you to effectively manage the resources and carry out various operations on them. Traditionally, various software vendors provided clustering functionality that allowed physical servers to operate together. Such clusters were used to host websites (as in webserver farms) or for running clustered databases for ensuring high availability.
- A similar concept has been brought forward in the virtualization environment as well where you can club virtual resources in their respective clusters and effectively manage them.

3.8.1 Virtual Clusters

Definition : Virtual clusters allow aggregating virtual resources for effective operations and management.

You can club various virtual resources and put them in respective groups. Fig. 3.8.1 shows two sample virtual clusters – one for development and one for production.

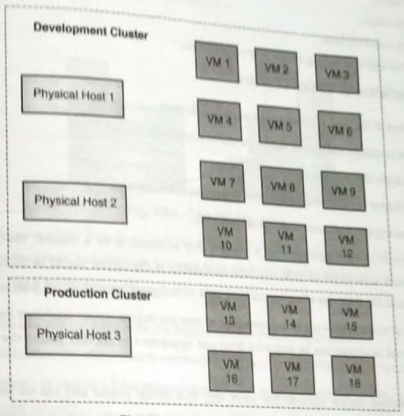


Fig. 3.8.1 : Sample Virtual Cluster

The Fig. 3.8.2, snapshot gives a cluster view on VMware vSphere.

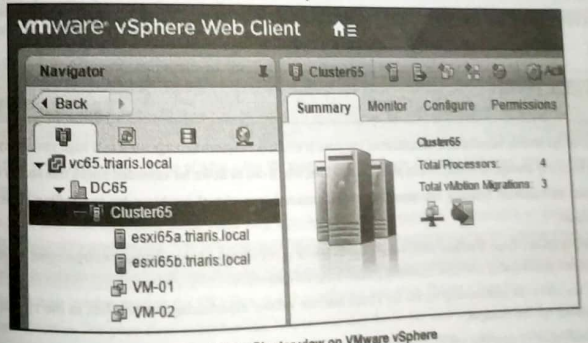


Fig. 3.8.2 : Cluster view on VMware vSphere

- You can create clusters based on your own requirements such as,
  - By location
  - By purpose
  - By department,
  - By administration,
  - By security requirements,
  - By any other logical grouping as per your requirement.

3.8.2 Characteristics of Virtual Clusters

Following are some of the basic characteristics of virtual clusters.

1. Multiple physical machines and virtual machines can be part of one cluster.
2. There can be multiple clusters in a virtual environment.
3. The size of each cluster can vary depending upon the resources it contains.
4. Different virtual machines in a cluster can run different OS.
5. The physical resources are aggregated in a cluster. For example, if in a cluster, there are three hypervisors (physical hosts) each with 16 GB of RAM, the RAM availability in the cluster would be 48 GB. You can create virtual machines that exceed the RAM size of a physical host say a virtual machine with 20 GB RAM.
6. The failure of one or more hypervisor in a cluster may or may not impact virtual machines depending upon the cluster configuration. Virtual machines can be migrated from one hypervisor to another within the same or across different clusters.
7. Various vendors provide several features that work only in a virtual cluster and not on independent hypervisor or virtual machine. For example, cluster high availability is a feature that works only on virtual clusters. It automatically moves virtual machines of one host to another host when the host goes down for any reason.
8. A virtual cluster provides effective resource management and is the preferred way of managing a virtual environment.

3.8.3 Live VM Migration

University Question

Q. What is Live VM migration? **SPPU - May 18, 2 Marks**

One of the several benefits that virtualization provides is effective resource management and high resiliency (ability to recover from any disruptive event). Unlike physical machines, which can be down for extended hours due to any disruption, the virtual environment typically has capabilities to continuously run virtual machines by controlling their execution environment.

***Definition : Live Virtual Machine Migration** is a capability of the virtual environment to change the execution context of a virtual machine without any downtime.*

- Yes, the users can continuously access the virtual machine without experiencing any problem as the virtual machine's execution context changes.
- The execution context that can be changed live is,

  - The physical host or the hypervisor where the virtual machine is currently running,
  - The storage device that is assigned to the virtual machine,
  - The network on which the virtual machine is placed,
  - The cluster where the virtual machine is placed.

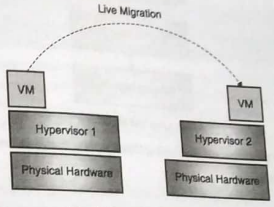


Fig. 3.8.3 : Live VM Migration

So, for example, a virtual machine from host 1 can move to host 2 while it is still running, and users are connected to it without experiencing any downtime. Unbelievable, but true! In the industry, the live VM migration is also called vMotion.

3.8.3(A) Advantages of Live VM Migration

1. **High resiliency :** You can move the VM from one host to another without downtime.
2. **Better resource utilisation :** In case a host is overloaded, you can move its VMs to other hosts.
3. **Planned maintenance :** In case of a host maintenance, you can temporarily move its VMs and bring them back once the maintenance activity is over.
4. **Automate VM placement :** Based on certain conditions, you can automate the placement of VMs. For example, if a host is underutilized, you can have a policy to automatically migrate some of the VMs to it from more utilized hosts and use the physical resources more effectively.

3.8.3(B) Live VM Migration Steps

University Question

Q. Write down the steps required for Live VM migration. **SPPU - May 18, 4 Marks**

- Now that you have a general idea of what a live VM migration is, let's learn about the high-level steps involved in VM migration.
- Note here that the migration steps could vary from hypervisor to hypervisor, environment to environment and configuration to configuration. But, at a high-level, the migration would include moving the virtual machine's host, memory, connections, etc. from one execution context to another execution context. Note here that the live VM migration may involve moving the VM's files on disk to different disk drive. That is a separate process and is called storage vMotion. For your clear understanding, let's assume that the source host and the target host have a shared storage where the VM's files are located and can be accessed after vMotion. Hence, you don't need to move the VM's files during VM migration process.

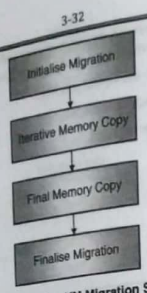


Fig. 3.8.4 : Live VM Migration Steps

- Let's understand the live VM migration steps assuming that the VM is currently running on Host A (source) and needs to be migrated to Host B (destination).

1. **Initialise Migration**

In the initialise migration step, the following tasks are carried out.

(a) First of all, it is checked that the VM running on the source host (Host A) can be operated on the destination host (Host B) and the destination host (Host B) fulfils the execution criteria such as availability of memory, devices, shared storage, resource policies, placement policies, etc.

(b) On the destination host (Host B), the resources to accommodate the VM from the source host (Host A) are reserved.

(c) The source host (Host A) creates a memory checkpoint to capture the memory changes that occur when the live migration process is in progress. This way only the minimal memory changes would require to be replicated on the destination host (Host B) once the rest of the VM memory is successfully copied.

2. **Iterative Memory Copy**

During this step, the VM memory is copied to the destination host (Host B). The memory transfer takes place via the network connection between the source host (Host A) and the destination host (Host B). This steps continues iteratively (several times) until the VM memory is copied till the reference checkpoint created in the previous step is reached.

3. **Final Memory Copy**

In this step, the virtual machine is temporarily quiesced (frozen temporarily) for fraction of milliseconds. During this time, the final memory changes, that happened after the checkpoint was created, are copied to the destination host (Host B). After the memory changes are copied, the networking devices are notified of the change in the VM's MAC address.

4. **Finalise Migration**

In the final step, the VM on source host (Host A) is stopped and its memory pages are set free. It is deleted from the source host (Host A). The VM is initialised on the destination host (Host B) and it resumes operation.

3.9 Grid, Cloud and Virtualization

Let's learn how virtualization helps in grid and cloud computing.

3.9.1 Virtualization in Grid

Note : Grid computing used to be a preferred choice for distributed computing in early days of computing. With the evolution and growth of cloud computing, even very high resource intensive workloads are deployed in the cloud instead of using grid computing. The topic of grid computing is just covered here for your understanding.

Grid computing can mean different things to different individuals. The grand vision is often presented as an analogy to power grids where users (or electrical appliances) get access to electricity through wall sockets with no care or consideration for where or how the electricity is actually generated. In this view of grid computing, computing becomes pervasive and individual users (or client applications) gain access to computing resources (processors, storage, data, applications, and so on) as needed with little or no knowledge of where those resources are located or what the underlying technologies, hardware, operating system, and so on are.

Definition : *Grid computing uses distributed interconnected computers and resources collectively to achieve higher performance computing and resource sharing.*

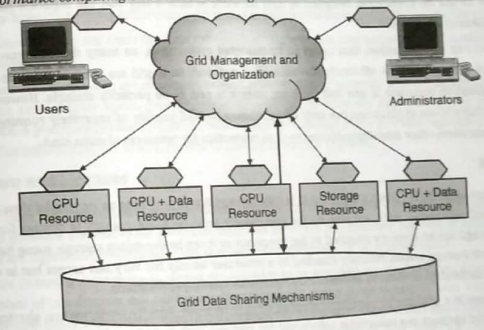


Fig. 3.9.1

- It was developed in the mid-1990s with the growth of high-speed networks and the Internet that allowed distributed computer systems to be readily interconnected. Grid computing has become one of the most important techniques in high performance computing by providing resource sharing in science, technology, engineering, and business. By taking advantage of the Internet and high-speed networks, geographically distributed computers can be used collectively for collaborative problem solving. In Grid computing, different organisations can supply the resources and personnel, and the Grid infrastructure can cross organisational and institutional boundaries.
- This concept has many benefits, including:
  - Problems that could not be solved previously for humanity because of limited computing resources can now be tackled. Examples include understanding the human genome and searching for new drugs.
  - Interdisciplinary teams can be formed across different institutions and organisations to tackle problems that require the expertise of multiple disciplines.

- Specialised experimental equipment can be accessed remotely and collectively within a Grid infrastructure.
- Large collective databases can be created to hold vast amounts of data.
- Unused compute cycles can be harnessed at remote sites, achieving more efficient use of computers.
- Business processes can be re-implemented using Grid technology for dramatic cost saving.

3.9.2 Types of Resources in Grid

A grid is a collection of machines, sometimes referred to as nodes, resources, members, donors, clients, hosts, engines, and many other such terms. They all contribute any combination of resources to the grid as a whole. Some resources may be used by all users of the grid, while others may have specific restrictions.

3.9.2(A) Computation

The most common resource is computing cycles provided by the processors of the machines on the grid. The processors can vary in speed, architecture, software platform, and other associated factors, such as memory, storage, and connectivity. There are three primary ways to exploit the computation resources of a grid.

1. The first and simplest is to use it to run an existing application on an available machine on the grid rather than locally.
2. The second is to use an application designed to split its work in such a way that the separate parts can execute in parallel on different processors.
3. The third is to run an application, that needs to be executed many times, on many different machines in the grid. Scalability is a measure of how efficiently the multiple processors on a grid are used. If twice as many processors makes an application complete in one half the time, then it is said to be perfectly scalable. However, there may be limits to scalability when applications can only be split into a limited number of separately running parts or if those parts experience some other interdependencies such as contention for resources of some kind.

3.9.2(B) Storage

- The second most common resource used in a grid is data storage. A grid providing an integrated view of data storage is sometimes called a data grid. Each machine on the grid usually provides some quantity of storage for grid use, even if temporary. Storage can be memory attached to the processor or it can be secondary storage, using hard disk drives or other permanent storage media. Memory attached to a processor usually has very fast access but is volatile. It would best be used to cache data or to serve as temporary storage for running applications.
- Secondary storage in a grid can be used in interesting ways to increase capacity, performance, sharing, and reliability of data. Many grid systems use mountable networked file systems, such as Andrew File System (AFS®), Network File System (NFS), Distributed File System (DFS™), or General Parallel File System (GPFS). These offer varying degrees of performance, security features, and reliability features.
- Capacity can be increased by using the storage on multiple machines with a unifying file system. Any individual file or database can span several storage devices and machines, eliminating maximum size restrictions often imposed by file systems shipped with operating systems. A unifying file system can also provide a single uniform name space for grid storage. This makes it easier for users to reference data residing in the grid, without regard for its exact location. In a similar way, special database software can federate an assortment of individual databases and files to form a larger, more comprehensive database, accessible using database query functions.
- More advanced file systems on a grid can automatically duplicate sets of data, to provide redundancy for increased reliability and increased performance. An intelligent grid scheduler can help select the appropriate storage devices to hold data, based on usage patterns. Then jobs can be scheduled closer to the data, preferably on the machines directly connected to the storage devices holding the required data.

- Data striping can also be implemented by grid file systems. When there are sequential or predictable access patterns to data, this technique can create the virtual effect of having storage devices that can transfer data at a faster rate than any individual disk drive. This can be important for multimedia data streams or when collecting large quantities of data at extremely high rates from CAT scans or particle physics experiments, for example.
- A grid file system can also implement journaling so that data can be recovered more reliably after certain kinds of failures. In addition, some file systems implement advanced synchronization mechanisms to reduce contention when data is shared and updated by many users.

3.9.2(C) Communications

- The rapid growth in communication capacity among machines today makes grid computing practical, compared to the limited bandwidth available when distributed computing was first emerging. Therefore, it should not be a surprise that another important resource of a grid is data communication capacity. This includes communications within the grid and external to the grid. Communications within the grid are important for sending jobs and their required data to points within the grid. Some jobs require a large amount of data to be processed, and it may not always reside on the machine running the job. The bandwidth available for such communications can often be a critical resource that can limit utilization of the grid.
- External communication access to the Internet, for example, can be valuable when building search engines. Machines on the grid may have connections to the external Internet in addition to the connectivity among the grid machines.
- When these connections do not share the same communication path, then they add to the total available bandwidth for accessing the Internet.
- Redundant communication paths are sometimes needed to better handle potential network failures and excessive data traffic. In some cases, higher speed networks must be provided to meet the demands of jobs transferring larger amounts of data. A grid management system can better show the topology of the grid and highlight the communication bottlenecks. This information can in turn be used to plan for hardware upgrades.

3.9.2(D) Software and Licenses

- The grid may have software installed that may be too expensive to install on every grid machine. Using a grid, the jobs requiring this software are sent to the particular machines on which this software happens to be installed. When the licensing fees are significant, this approach can save significant expenses for an organisation.
- Some software licensing arrangements permit the software to be installed on all of the machines of a grid but may limit the number of installations that can be simultaneously used at any given instant. License management software keeps track of how many concurrent copies of the software are being used and prevents more than that number from executing at any given time. The grid job schedulers can be configured to take software licenses into account, optionally balancing them against other priorities or policies.

3.9.2(E) Special Equipment, Capacities, Architectures, and Policies

- Platforms on the grid will often have different architectures, operating systems, devices, capacities, and equipment. Each of these items represents a different kind of resource that the grid can use as criteria for assigning jobs to machines. While some software may be available on several architectures, for example, PowerPC and x86, such software is often designed to run only on a particular type of hardware and operating system. Such attributes must be considered when assigning jobs to resources in the grid.
- In some cases, the administrator of a grid may create a new artificial resource type that is used by schedulers to assign work according to policy rules or other constraints. For example, some machines may be designated to only be used for medical research.

- These would be identified as having a medical research attribute and the scheduler could be configured to only assign jobs that require machines of the medical research resource. Others may participate in the grid only if they are not used for military purposes. In this situation, jobs requiring a military resource would not be assigned to such machines. Of course, the administrators would need to impose a classification on each kind of job through some certification procedure to use this kind of approach.

3.9.2(F) Jobs and Applications

- Although various kinds of resources on the grid may be shared and used, they are usually accessed via an executing application or job. Usually we use the term application as the highest level of a piece of work on the grid. However, sometimes the term job is used equivalently. Applications may be broken down into any number of individual jobs, as illustrated in the following figure.

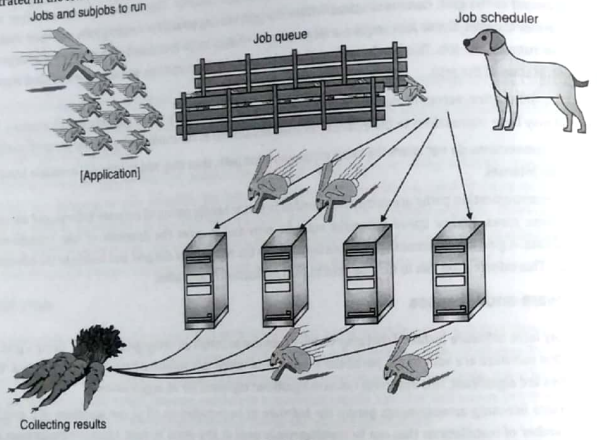


Fig. 3.9.2

- Those, in turn, can be further broken down into subjobs. The grid industry uses other terms, such as transaction, work unit, or submission, to mean the same thing as a job.
- Jobs are programs that are executed at an appropriate point on the grid. They may compute something, execute one or more system commands, move or collect data, or operate machinery. A grid application that is organised as a collection of jobs is usually designed to have these jobs execute in parallel on different machines in the grid.
- The jobs may have specific dependencies that may prevent them from executing in parallel in all cases. For example, they may require some specific input data that must be copied to the machine on which the job is to run. Some jobs may require the output produced by certain other jobs and cannot be executed until those prerequisite jobs have completed executing. Jobs may spawn additional subjobs, depending on the data they process. This workflow can create a hierarchy of jobs and subjobs. Finally, the results of all of the jobs must be collected and appropriately assembled to produce the ultimate output/result for the application.

3.9.3 Virtualization and Cloud Computing (Virtualization in Cloud)

- Virtualization is the core technology that cloud computing is based off. The resources provided by the cloud service providers are virtualized at various layers. Compute, Storage, Network – each part of cloud computing has some or the other form of virtualization technology behind the scenes.
- Table 3.9.1 summarises the hypervisors used by major cloud service providers.

Table 3.9.1

Sr. No.	Cloud Service Provider	Hypervisor used
1.	Amazon Web Services (AWS)	Xen and KVM
2.	Google Cloud Platform	KVM
3.	Microsoft Azure	Customised Hyper-V
4.	Alibaba Cloud	Xen and KVM

- But, note here that virtualization and cloud computing are not the same. Virtualization is just a technology that cloud computing uses. Cloud computing embraces several characteristics and design principles that you have learned in Unit 1. A set of resources without cloud characteristics and design principles could just be a virtualized environment and cannot be rightly called a cloud computing environment.
- Table 3.9.2 summarises the key differences between virtualization and cloud computing.

Table 3.9.2

Sr. No.	Comparison Attribute	Virtualization	Cloud Computing
1.	What it is?	Technology	Methodology and Principles
2.	Purpose	Get the most from hardware	Deliver compute resources on demand
3.	Lifespan	Years	Short-lived and on-demand
4.	Expenditure	High	Low
5.	Investment	Capital as well as Operational	Operational investment only
6.	Scalability	Up to hardware limit only	Nearly infinite
7.	Ownership	Owned by one	Shared tenancy
8.	Innovation and changes	Slow	Rapid
9.	Adopted by	Large enterprises only	Individuals, small to large enterprises
10.	Skills required to operate	High	Low and specific to the consumed service
11.	Shifting to another vendor	Complex and Costly	Comparatively easier and cheap
12.	Governance	Self-owned	Shared between cloud provider and tenant
13.	Primary consumption method	Direct interaction	Programmatic via APIs

Q. 49 What are the advantages of Live VM Migration? [4 Marks]

Q. 50 Detail out the steps involved in a Live VM Migration. [6 Marks]

[F] Grid, Cloud and Virtualization

Q. 51 Explain grid computing. [4 Marks]

Q. 52 Describe advantages of grid computing. [4 Marks]

Q. 53 Explain types of resources in grid computing. [6 Marks]

Q. 54 Explain how jobs and applications are run in grid computing. [4 Marks]

Q. 55 Write a short note on Virtualization in Cloud Computing. [4 Marks]

Q. 56 Compare virtualization and cloud computing. [6 Marks]

□□□

4 Cloud Platforms and Cloud Applications

Syllabus

At the end of this unit, you should be able to understand and comprehend the following syllabus topics :

- Amazon Web Services (AWS)
  - Amazon Web Services and Components
  - Amazon Simple DB
  - Elastic Cloud Computing (EC2)
  - Amazon Storage System
  - Amazon Database services (Dynamo DB)
- Microsoft Cloud Services
  - Azure core concepts
  - SQL Azure
  - Windows Azure Platform Appliance
- Cloud Computing Applications
  - Healthcare: ECG Analysis in the Cloud
  - Biology: Protein Structure Prediction
  - Geosciences: Satellite Image Processing
  - Business and Consumer Applications: CRM and ERP, Social Networking
  - Google Cloud Application: Google App Engine
- Overview of OpenStack architecture

4.1 Amazon Web Services (AWS)

University Question

Q. Write a note on Amazon Web Services. SPPU : May 18, 5 Marks

- Amazon Web Services (AWS) is one of the major public cloud service providers. AWS has significantly more services, and more features within those services, than any other cloud provider. The AWS cloud platform offers over 165 fully featured services.
- AWS provides its services throughout the world. Fig. 4.1.1 is a snapshot of its regions currently (Nov 2019). Each of the regions is further divided into availability zones to isolate the datacentres within the region from impacting each other in case of disruptions.

✍ ***Definition :*** *Region is the geographical boundary of a cloud service.*

✍ ***Definition :*** *Availability Zone is the isolation boundary of a cloud service within a particular region.*

4 Cloud Platforms and Cloud Applications

Syllabus

At the end of this unit, you should be able to understand and comprehend the following syllabus topics :

- Amazon Web Services (AWS)
  - Amazon Web Services and Components
  - Amazon Simple DB
  - Elastic Cloud Computing (EC2)
  - Amazon Storage System
  - Amazon Database services (Dynamo DB)
- Microsoft Cloud Services
  - Azure core concepts
  - SQL Azure
  - Windows Azure Platform Appliance
- Cloud Computing Applications
  - Healthcare: ECG Analysis in the Cloud
  - Biology: Protein Structure Prediction
  - Geosciences: Satellite Image Processing
  - Business and Consumer Applications: CRM and ERP, Social Networking
  - Google Cloud Application: Google App Engine
- Overview of OpenStack architecture

4.1 Amazon Web Services (AWS)

University Question

Q. Write a note on Amazon Web Services. SPPU - May 18 - 5 Marks

- Amazon Web Services (AWS) is one of the major public cloud service providers. AWS has significantly more services, and more features within those services, than any other cloud provider. The AWS cloud platform offers over 165 fully featured services.
- AWS provides its services throughout the world. Fig. 4.1.1 is a snapshot of its regions currently (Nov 2019). Each of the regions is further divided into availability zones to isolate the datacentres within the region from impacting each other in case of disruptions.

✓ ***Definition : Region is the geographical boundary of a cloud service.***

✓ ***Definition : Availability Zone is the isolation boundary of a cloud service within a particular region.***

- A region may have two or more availability zones to provide redundancy. An availability zone may have one or more datacentres equipped with independent power, cooling and networking.

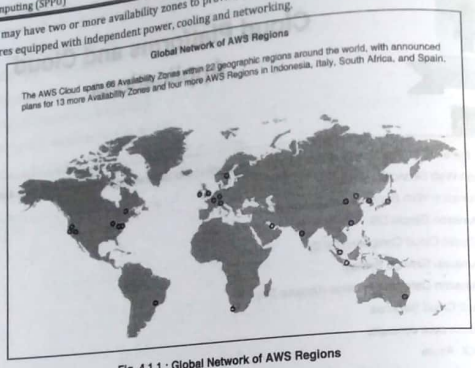


Fig. 4.1.1 : Global Network of AWS Regions

Conceptually, the regions and availability zones within them can be visualized as shown in Fig. 4.1.2.

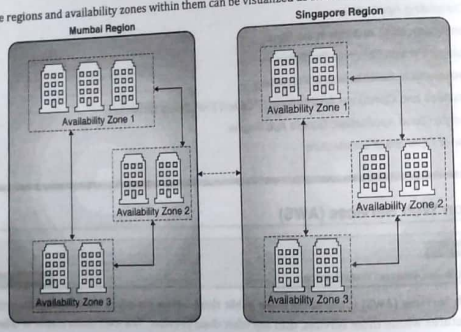


Fig. 4.1.2 : Availability Zones within Various Regions

Note : Any cloud service provider adds several services in a year. Given the number of services and features within those services, it is out of the scope for this book to detail out or even list all the services and their features. So, I would be listing only the services required for as per the syllabus. If you are interested to dive deeper or learn more, please visit https://aws.amazon.com. It would additionally help if you create a free account on AWS to play around with the services for a while.

4.2 Amazon EC2

Definition : *Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity (virtual machines) in the cloud.*

Amazon EC2 is perhaps the most used service in the AWS environment. EC2 provides virtual machines that you can use for any of your computing requirements. These virtual machines are commonly called as EC2 instances.

4.2.1 Characteristics and Features of EC2

University Question

Q. What are the types of instances of Amazon EC2 ? SPPU - May 19, 5 Marks

1. IaaS

EC2 is an example of an IaaS. You own the OS, networking, what is run on the machine and the responsibility to keep it protected. You are provided complete access to it as in a virtual machine running in your own datacentre. However, you do not have access to the underlying hypervisor on which your EC2 instance is running.

2. Several instance types

- You can choose from a variety of instance types to match your computing requirements. Instance types define the hardware configuration of your EC2 instances. You are charged differently depending upon the instance type you choose. The higher is your hardware configuration, the costlier it is to run the corresponding EC2 instance. Table 4.2.1 gives a few examples of various instance types.

Table 4.2.1

Sr. No.	Instance Family	Instance Type Example	Hardware Configuration	Price per hour
1.	General Purpose	a1.medium	1 CPU, 2 GB RAM	$0.0255
2.	General Purpose	a1.4xlarge	16 CPU, 32 GB RAM	$0.408
3.	General Purpose	m5.24xlarge	96 CPU, 384 GB	$4.608
4.	Compute Optimized	c5d.18xlarge	72 CPU, 144 GB	$3.456
5.	Accelerated Computing	g3s.xlarge	4 CPU, 30.5 GB	$0.75
6.	Memory Optimized	x1e.32xlarge	128 CPU, 3,904 GB	$26.688
7.	Storage Optimized	i3.4xlarge	16 CPU, 122 GB	$1.248

- Imagine if you had to purchase a hardware having 3,900 GB RAM. Can you? The cloud really makes it cost effective and feasible to choose from various instance types depending on your computing requirements.
- Caution : The pricing is per region and is subject to change. Consider the instance family and their details for your reference only.

3. Start and terminate instances as per your requirements

You can start and terminate EC2 instances as per your requirement. Like typically happens in cloud environment, you only pay for what you use. EC2 instances are typically billed per hour.

4. **Elastic IP Addresses**

You can either assign a static or dynamic IP address for your EC2 instances. IP addresses are assigned based on the region.

5. **Auto Scaling**

Based on various conditions, such as CPU consumption, increased load, etc., you can automatically add more EC2 instances to support your computing requirements. Once the increased demand is taken care and you no more require as many EC2 instances, you can automatically terminate the extra instances.

6. **Multiple OS to choose from**

You can choose from a variety of Linux and Windows OS for your EC2 instances. You can also choose customised Amazon Machine Images (called AMI in short) from various vendors to support a particular use case or a specific computing requirement.

4.2.2 Creating an EC2 Instance

- You can either create an EC2 instance manually using the AWS management console (User Interface) or programmatically via APIs. You can choose a region and an availability zone within that region to launch the EC2 instance.

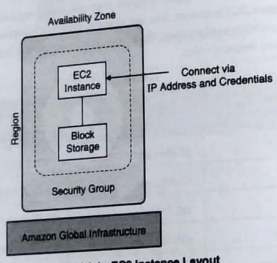


Fig. 4.2.1 : EC2 Instance Layout

- Typically, following steps are taken to create and launch an EC2 instance manually. Similar steps are taken to create instances via APIs.

1. Create credentials that you want to assign to the EC2 instance.
2. Choose an Amazon Machine Image (AMI).
3. Choose an Instance Type.
4. Configure Instance Details such as network and storage.
5. Add labels or tags for identifying your EC2 instances.
6. Configure firewall (called security group) as appropriate.
7. Review and launch the EC2 instance.
8. Once the EC2 instance is created, you can connect to it using the chosen credentials.
9. Let's see some snapshots and details around it.

Note : I would again recommend creating a free AWS account and explore the steps. The steps are subject to change.

1. **Create Credentials**

- Amazon EC2 uses public key cryptography to encrypt and decrypt login information. The public and private keys are known as a key pair. Key pair enables you to securely access your EC2 instances using a private key instead of a password. When you launch an instance, you specify the key pair. You can specify an existing key pair or a new key pair that you create at launch.
  - For Windows EC2 instances, you use the private key to obtain the administrator password and then log in using Remote Desktop.
  - For Linux instances, you can login directly using the key pair.
- To create a keypair, typically, following steps are taken.

Step 1 : Go to AWS Console and choose EC2 service and the region in which you want to create the key pair for launching instances. Click on Key Pairs in the navigation menu (either on left hand side or central pane).

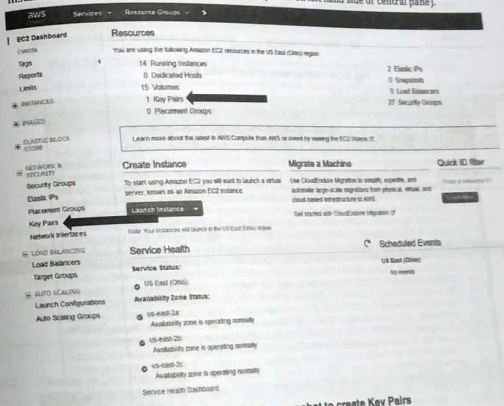


Fig. 4.2.2 : AWS screenshot to create Key Pairs

Step 2 : Click on Create Key Pair. Provide Key Pair Name and click on Create.

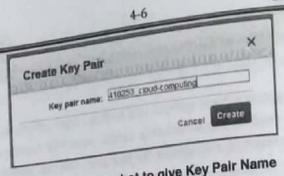


Fig. 4.2.3 : Screenshot to give Key Pair Name

- This creates a new key pair and also automatically downloads the private key. You should save this private key safely. You would need this for login into your EC2 instances.

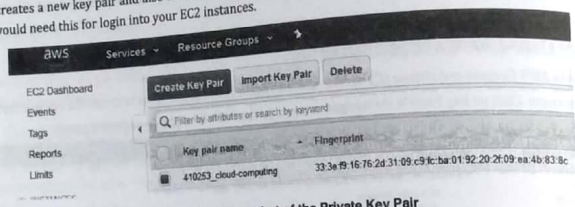


Fig. 4.2.4 : Screenshot of the Private Key Pair

- The private key file looks like the Fig. 4.2.5.

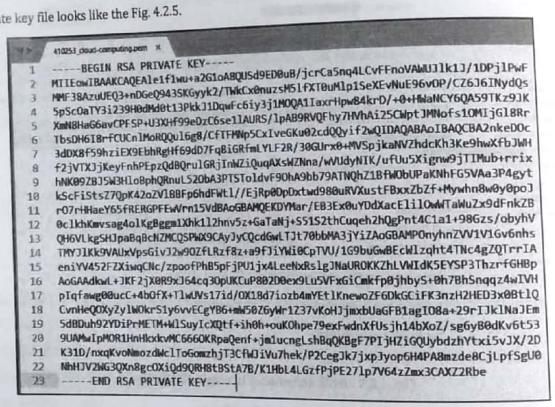


Fig. 4.2.5 : Screenshot showing Private Key File

2. Creating an instance

- Once you have the key pair created, you can create an EC2 instance. The EC2 instance creation wizard walks you through several configuration options. You can choose the options as per your requirements.
- On the EC2 service console, click on Launch Instance. This would launch an EC2 instance creation wizard.

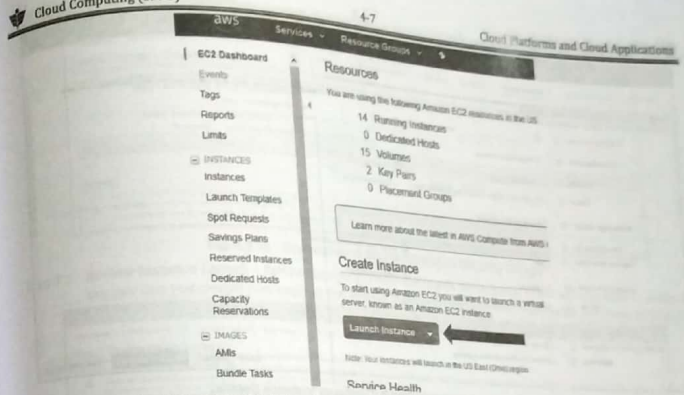


Fig. 4.2.6 : Screenshot showing EC2 Service Console to Launch EC2 Instance Wizard

- The wizard shows you the top ribbon consisting of various steps that you need to take for configuring the instance.

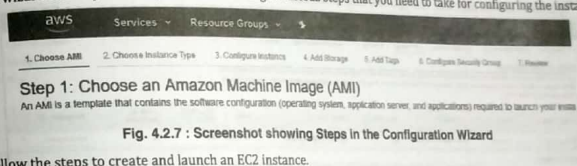


Fig. 4.2.7 : Screenshot showing Steps in the Configuration Wizard

- Follow the steps to create and launch an EC2 instance.

Step 1 : Choose an Amazon Machine Image (AMI) - I am choosing a Windows OS image.

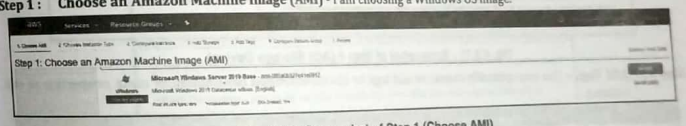


Fig. 4.2.8 : Screenshot of Step 1 (Choose AMI)

Step 2 : Choose an Instance Type - I chose the instance type "t2.micro".

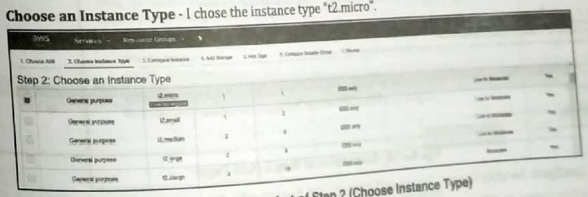


Fig. 4.2.9 : Screenshot of Step 2 (Choose Instance Type)

Step 3 : Configure Instance Details - I am good with defaults. You can configure these settings as per your requirements. For example, you can choose either a private IP or Elastic IP

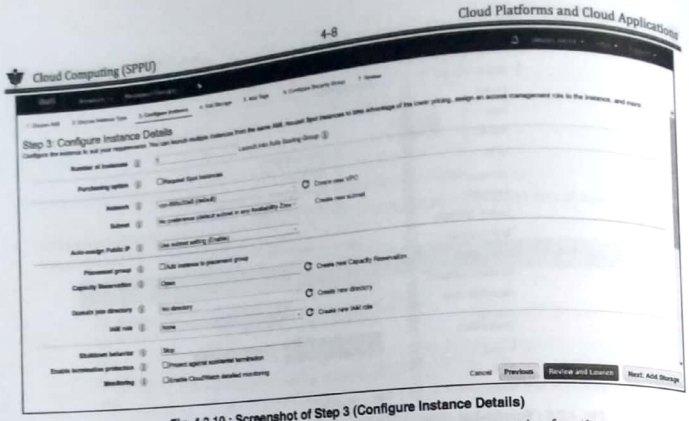


Fig. 4.2.10 : Screenshot of Step 3 (Configure Instance Details)

Step 4 : **Add Storage –** I am good with free-tier defaults. Note here that if you are not using free-tier you would have options to choose disk size, encryptions, disk speed, etc.

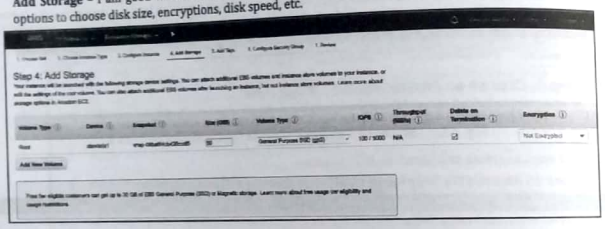


Fig. 4.2.11 : Screenshot of Step 4 (Add Storage Device)

Step 5 : **Add Tags –** You can optionally choose to add tags to add identification information or meta properties of this instance.

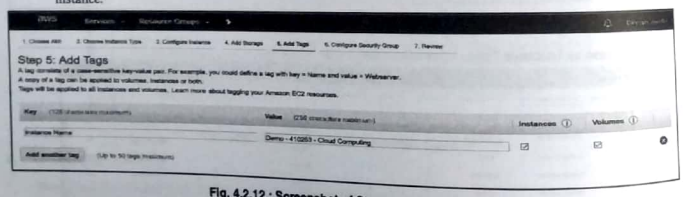


Fig. 4.2.12 : Screenshot of Step 5 (Add Tags)

Step 6 : **Configure Security Group –** You should carefully add firewall rules for your instance. For example, you should allow access to this instance only from your IP address and not from any other IP in the world. Firewall requirements would vary based on your scenario for which you are launching the instance.

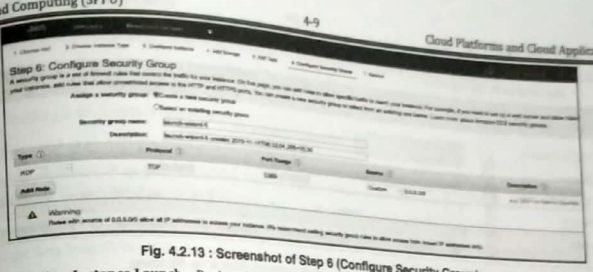


Fig. 4.2.13 : Screenshot of Step 6 (Configure Security Group)

Step 7 : **Review Instance Launch –** Review the configuration and launch the instance. You would also need to assign the key pair that you created previously so that you can login to the instance once it is created and ready for use.

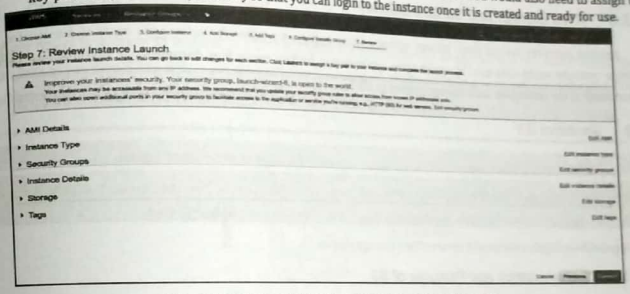


Fig. 4.2.14 : Screenshot of Step 7 (Review Instance Launch)

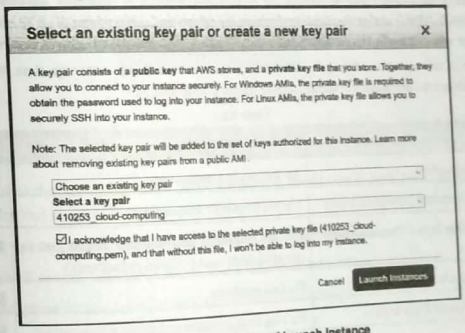


Fig. 4.2.15 : Screenshot of Launch Instance

This would launch an instance.

Step 8 : **Connect** - Once the instance is ready, it would show as running. You can click on Connect to specify the key pair and connect to the instance.

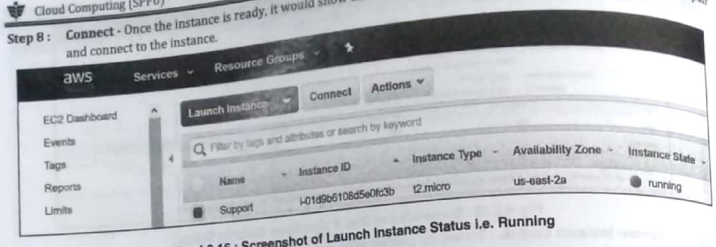


Fig. 4.2.16 : Screenshot of Launch Instance Status i.e. Running

4.2.3 AWS Storage and Content Delivery

Amazon provides several cloud storage options. These options are suited for various use cases and scenarios. Cloud storage typically stores data used by applications and websites, backup, media files, documents or anything else. Let's read about some of the commonly used storage options in AWS.

4.3 Amazon S3

Definition : *Amazon Simple Storage Service (Amazon S3) is an object storage service.*

Note : Object storage is meant to store files containing music, video, text, images, documents, etc. You cannot use object storage for running OS. Other examples of object storage are Dropbox and Google Drive.

It provides a highly reliable and secure object storage service.

4.3.1 Characteristics and Features of S3

1. **11 9's of data durability :** This is perhaps the most useful measure of reliability. Amazon S3 is designed for 99.999999999% (11 9's) of data durability. It automatically creates and stores copies of all S3 objects across multiple systems. This means that your data is available when needed and protected against failures, errors, and threats.
2. **Wide-range of storage class :** Based on your requirements, you can choose a storage class for storing your data. Table 4.3.1 gives a quick summary of various storage classes available for S3.

Table 4.3.1

Sr. No.	*Storage Class*	*Purpose*	*Price per GB*
1.	S3 Standard	For frequently accessed data	$0.023
2.	S3 Standard-Infrequent Access	For less frequently accessed data	$0.0125
3.	S3 One Zone-Infrequent Access	For less frequently accessed data stored just in one availability zone (reduced redundancy)	$0.01
4.	S3 Glacier	For data archiving	$0.004
5.	S3 Glacier Deep Archive	For long-term retention and digital preservation	$0.00099

Caution : The pricing is per region and is subject to change. Consider the storage class and their details for your reference only.

i. **Data lifecycle management :** S3 provides various data lifecycle management features such as
   a. Automatically moving the data to a different storage class when infrequently used.
   b. Maintaining data versions as it is updated.
   c. Locking down update or erasure of data.

ii. **Security and Access Control :** S3 provides various types of access control policies that you can apply to ensure that the data is accessible to only authorized entities. It also warns you if you expose your S3 data publicly. You can additionally encrypt the data stored in S3.

iii. **Query S3 data :** S3 also provides features and services to query the data stored in it for data analytics. You don't need to copy data from S3 to a different machine for performing analytics. You can directly perform analytics on the data stored in S3.

4.3.2 Creating an Amazon S3 Bucket

- Every object in Amazon S3 is stored in a bucket. Bucket is a placeholder for all objects. You can create one or more buckets as you would like. Before you can store data in Amazon S3, you must create a S3 bucket. You can either create a bucket manually using the management console or via the APIs.

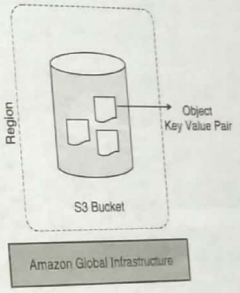


Fig. 4.3.1 : Layout of a S3 Bucket

- Once you have created a S3 bucket, you can upload objects to it. An object is a key value pair. It just references the filename as the key and its data as the binary value associated with the key.
- You can work with the object stored in the bucket. For example, you can view it, download it, process it or delete it if is no longer required.
- Let's see a step by step process of creating a S3 bucket and storing objects in it.

Step 1 : Login to AWS console and go to S3 service. There click on Create Bucket.

- Enter a DNS compliant domain name (meaning fully qualified domain name).
- Choose the Region where you want to create the bucket.

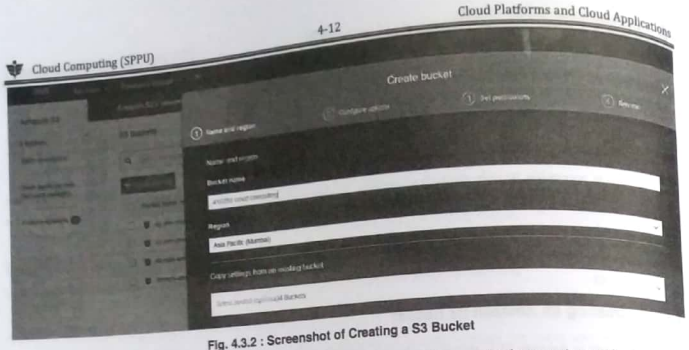


Fig. 4.3.2 : Screenshot of Creating a S3 Bucket

Step 2 : Configure bucket. You can choose options as per your requirement. I have enabled encryption and left everything else at default.

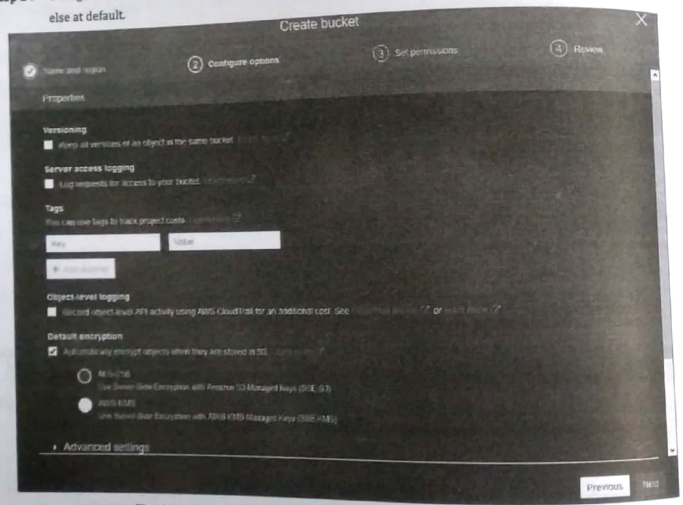


Fig. 4.3.3 : Screenshot showing Configurations Options for a S3 Bucket

Step 3 : Set Permissions. You can set permissions on the bucket as appropriate. Choose to not allow public access if you don't want your data to be accessed by anyone else from the internet.

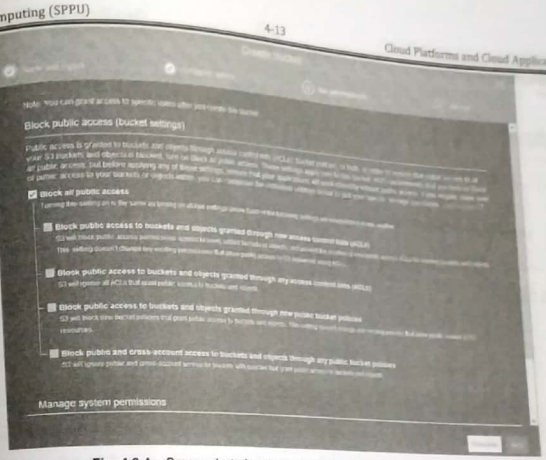


Fig. 4.3.4 : Screenshot showing Permissions Options for a S3 Bucket

Step 4 : Review and Create.

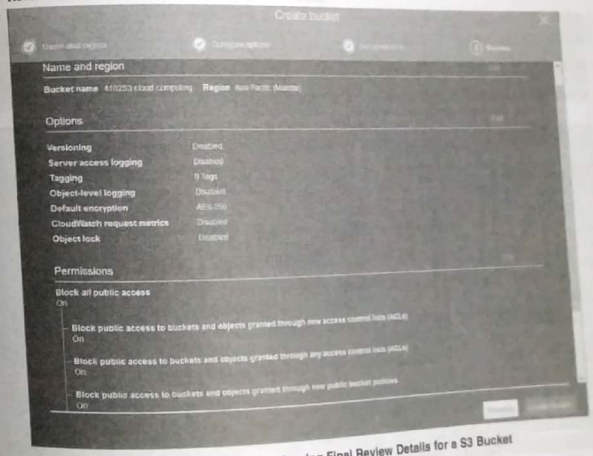


Fig. 4.3.5 : Screenshot showing Final Review Details for a S3 Bucket

This would create a bucket.

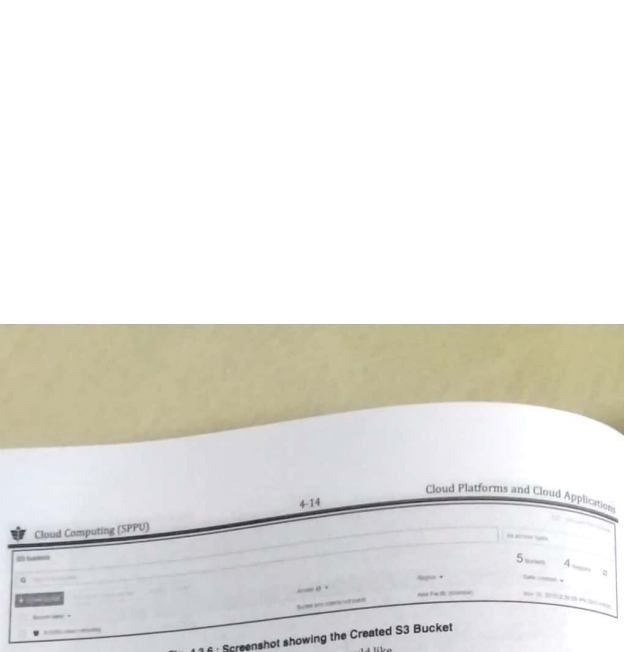


Fig. 4.3.6 : Screenshot showing the Created S3 Bucket

Step 5 : Upload files. You can now upload the objects (files) as you would like.

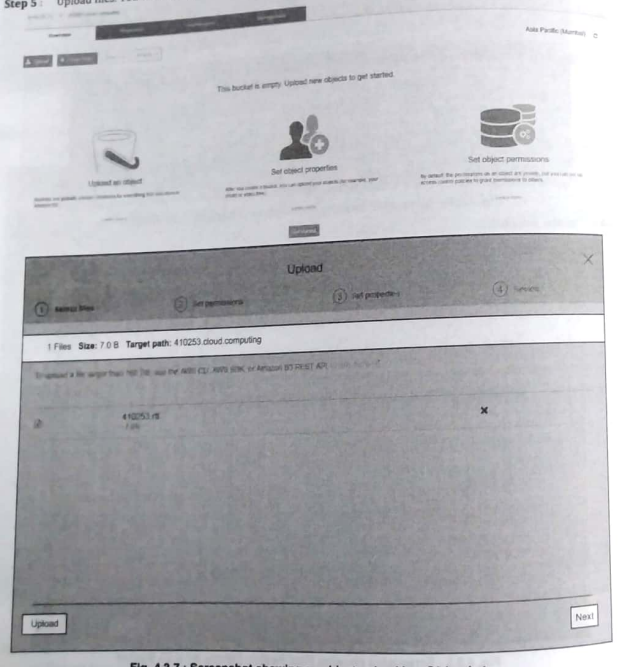


Fig. 4.3.7 : Screenshot showing an object upload in a S3 bucket

- You can upload objects and set the permissions as you would like to. You can then manage permissions as appropriate on this uploaded object.

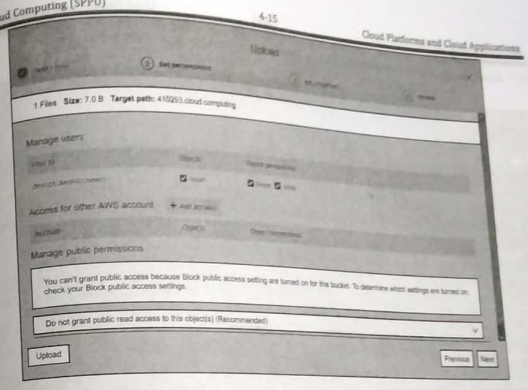
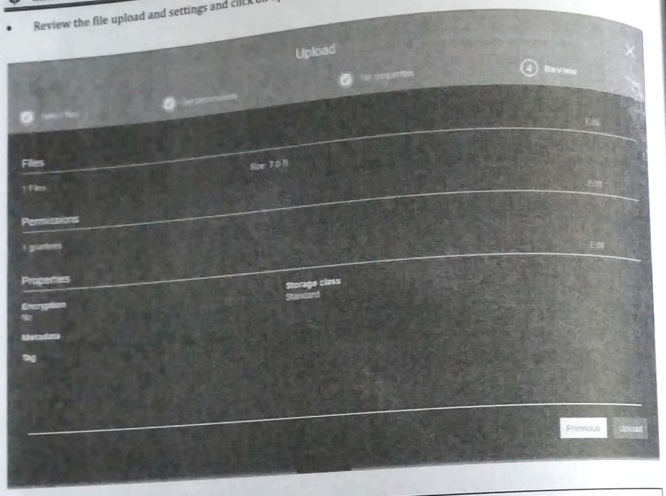


Fig. 4.3.8 : Screenshot showing Permission Options for an Object in S3 Bucket

- Then choose the desired storage class.

Fig. 4.3.9 : Screenshot showing Properties Options for an Object in S3 Bucket

- Review the file upload and settings and click on upload if everything is ok.



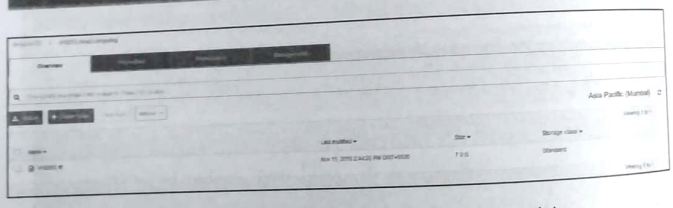


Fig. 4.3.10 : Screenshot showing Review Details for an Object in S3 Bucket

- You can then interact with the object as required or see and edit its properties.

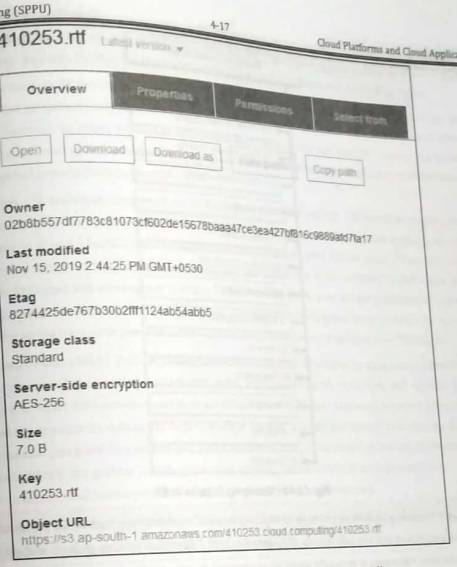


Fig. 4.3.11 : Screenshot showing Uploaded Object Details

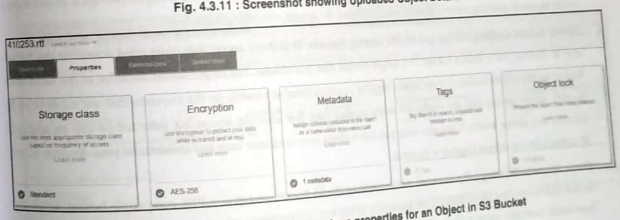


Fig. 4.3.12 : Screenshot showing various properties for an Object in S3 Bucket

4.3.3 Managing Objects in S3

- There are several configurations that you can do on objects in S3. They are as shown in Fig. 4.3.13.

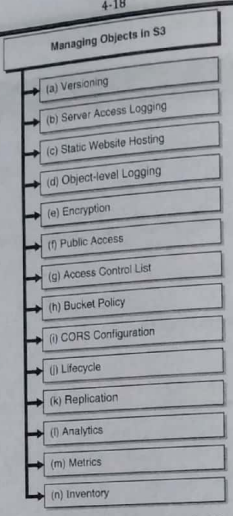


Fig. 4.3.13 : Managing Objects in S3

1. **Versioning :** Versioning enables you to keep multiple versions of an object in a bucket. So, for example, if you are working on a document and you take backup of it every day until 7 days then you can go back and get the file version on any of the days. Versioning is especially useful when you need to compare the file changes over time, or you have a requirement to possibly restore or read older versions. By default, versioning is disabled.

2. **Server Access Logging :** When you enable logging, Amazon S3 delivers access logs (who tried accessing the bucket and its objects) for a source bucket to a target bucket that you choose. The target bucket must be in the same AWS Region as the source bucket. Server access logging provides detailed records for the requests that are made to an S3 bucket. Server access logs are useful for many applications. For example, access log information can be useful in security and access audits. It can also help you learn about your customer base and understand your Amazon S3 bill. By default, Server Access Logging is disabled.

3. **Static Website Hosting :** If you recall, while creating the S3 bucket, you gave a DNS compatible bucket name. The bucket name can then be accessed from anywhere in the world if the permission is granted. This makes it possible to host static websites using a S3 bucket. You can host a static website on Amazon S3. On a static website, individual WebPages include static content, and they might also contain client-side scripts. By contrast, a dynamic website relies on server-side processing, including server-side scripts such as PHP, JSP, or ASP.NET. Amazon S3 does not support server-side scripting. To configure static website, you need to upload the website pages into the bucket, point to index.html as the homepage and allow public access to the bucket so that users can connect to it.

4. **Object-level Logging :** You can also enable object-level logging. Object-level logging enables Amazon S3 object-level API activity. For example, GetObject, DeleteObject and PutObject API operations. It is disabled by default.

5. **Encryption :** Amazon S3 default encryption provides a way to set the default encryption behaviour for an Amazon S3 bucket. You can set default encryption on a bucket so that all objects are encrypted when they are stored in the bucket. Amazon S3 encrypts an object before saving it to disk in its datacentres and decrypts it when you download the objects. By default, encryption is disabled. Amazon S3 can manage your encryption keys but it could also be configured such that you can manage the keys yourself.

6. **Public Access :** S3 bucket and its content can be made to be accessible publicly. You should be careful about it when configuring the S3 bucket permissions. Until and unless it is desired, for example when hosting static website, all public access to your S3 buckets must be disabled to avoid exposing your data to public. There have been several incidence of security and data breaches for even big companies where the sensitive business data was publicly available in S3 bucket and was exposed. Amazon S3 block public access prevents the application of any settings that allow public access to data within S3 buckets. You can configure block public access settings for an individual S3 bucket or for all the buckets in your account. By default, all public access is disabled on your S3 buckets.

7. **Access Control List (ACL) :** You can manage permissions on your S3 buckets by using Access Control Lists (ACLs). ACLs are resource-based access policies that grant access permissions to buckets and objects. You can grant permissions to other AWS account users or to predefined groups. The user or group that you are granting permissions to is called the grantee. By default, the owner, which is the AWS account that created the bucket has full permissions. Each permission you grant for a user or group adds an entry in the ACL that is associated with the bucket. The ACL lists grants that identify the grantee and the permission granted. You can typically provide access to list bucket content, write new content, read bucket permissions and write (change) bucket permissions.

Note : AWS recommends using S3 bucket policies (explained next) for access control. S3 ACLs is a legacy access control mechanism. Do not use it for new S3 buckets that you create.

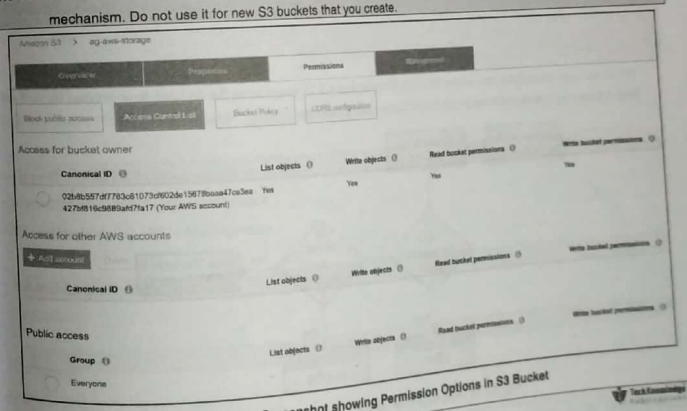


Fig. 4.3.14 : Screenshot showing Permission Options in S3 Bucket

8. **Bucket Policy :** A bucket policy is a resource-based AWS Identity and Access Management (IAM) policy. You add a bucket policy to a bucket to grant other AWS accounts or IAM users access permissions for the bucket and the objects in it. Object permissions apply only to the objects that the bucket owner creates. S3 bucket policies specify what actions are allowed or denied for which principals (users) on the bucket that the bucket policy is attached to. S3 bucket policies are a type of access control list (ACL). You attach S3 bucket policies at the bucket level. You cannot attach a bucket policy to an S3 object, but the permissions specified in the bucket policy apply to all the objects in the bucket. Following is a sample S3 bucket policy (taken from AWS documentation) that restricts access to a S3 bucket (410253.bucket) to specific IP Address ranges.

```
{
 "Version": "2012-10-17",
 "Id": "S3PolicyId1",
 "Statement": [
  {
   "Sid": "IPAllow",
   "Effect": "Allow",
   "Principal": "*",
   "Action": "s3:*",
   "Resource": "arn:aws:s3:::410253.bucket/*",
   "Condition": {
    "IpAddress": {"aws:SourceIp": "54.240.143.0/24"},
    "NotIpAddress": {"aws:SourceIp": "54.240.143.188/32"}
   }
  }
 ]
}
```

You should also understand how access decision is made. Fig. 4.3.15 summarises the decision flow process carried out before a user is allowed or denied access to the bucket and its objects.

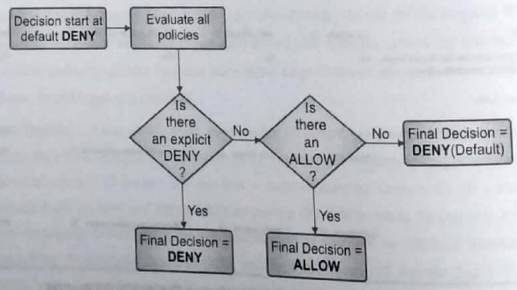


Fig. 4.3.15 : Flowchart for taking Action Decision in AWS

9. **CORS configuration :** CORS is an acronym for Cross-Domain Resource Sharing. You can configure Cross-Origin Resource Sharing (CORS) on an S3 bucket.

- Suppose that you are hosting a website in an Amazon S3 bucket named website. Your users load the website ashttp://website.s3-website-us-east-1.amazonaws.com. Now you want to use JavaScript on the webpages that are stored in this bucket to be able to make authenticated GET and PUT requests against the same bucket by using the Amazon S3 API endpoint for the bucket, website.s3.amazonaws.com. A browser would normally block JavaScript from allowing those requests, but with CORS you can configure your bucket to explicitly enable cross-origin requests from website.s3-website-us-east-1.amazonaws.com.
- CORS allows client web applications that are loaded in one domain to interact with resources in another domain. To configure your bucket to allow cross-origin requests, you add CORS configuration to the bucket. A CORS configuration is an XML document that defines rules that identify the origins that you will allow to access your bucket, the operations (HTTP methods) supported for each origin, and other operation-specific information.

Following is a sample CORS configuration file (taken from AWS documentation).

```
<CORSConfiguration>

<CORSRule>
<AllowedOrigin>http://www.example1.com</AllowedOrigin>

<AllowedMethod>PUT</AllowedMethod>
<AllowedMethod>POST</AllowedMethod>
<AllowedMethod>DELETE</AllowedMethod>

<AllowedHeader>*</AllowedHeader>
</CORSRule>

<CORSRule>
<AllowedOrigin>http://www.example2.com</AllowedOrigin>

<AllowedMethod>PUT</AllowedMethod>
<AllowedMethod>POST</AllowedMethod>
<AllowedMethod>DELETE</AllowedMethod>

<AllowedHeader>*</AllowedHeader>
</CORSRule>

<CORSRule>
<AllowedOrigin>*</AllowedOrigin>
<AllowedMethod>GET</AllowedMethod>
```

</CORSRule>

</CORSConfiguration>

10. **Lifecycle :** You can use lifecycle policies to define actions you want Amazon S3 to take during an object's lifetime. For example, you can
    a. Transition objects to another storage class,
    b. Archive them, or
    c. Delete them after a specified period of time.

- You can define a lifecycle policy for all objects or a subset of objects in the bucket. A versioning-enabled bucket can have many versions of the same object, one current version and zero or more previous versions. Using a lifecycle policy, you can define actions specific to current and previous object versions.

11. **Replication :** Replication allows automatic and asynchronous copying of objects across buckets in the same or different AWS Regions. Replication copies newly created objects and object updates from a source bucket to a destination bucket. Replication requires versioning to be enabled on both the source and destination buckets.

- The object replicas in the destination bucket are exact replicas of the objects in the source bucket. They have the same key names and the same metadata for example, creation time, owner, user-defined metadata, version ID, Access Control List (ACL), and storage class.
- Optionally, you can explicitly specify a different storage class for object replicas. Also, regardless of who owns the source bucket or the source object, you can choose to change replica ownership to the AWS account that owns the destination bucket.
- The time it takes for Amazon S3 to replicate an object depends on the object size. It can take up to several hours to replicate a large-sized object. Replication is useful for backup, archival or just for distribution of data.

12. **Analytics :** AWS S3 usage is charged based on the amount of storage you use and the storage class that you use. To make it easy for you to choose a particular storage class (based on your object access patterns) and to save on your AWS S3 bills, you can turn on AWS S3 storage analytics. By using the Amazon S3 analytics storage class analysis tool, you can analyse storage access patterns to help you decide when to transition the right data to the right storage class. Storage class analysis observes data access patterns to help you determine when to transition less frequently accessed STANDARD storage to the STANDARD_IA (IA - infrequent access) storage class. This way you can save your money and also transition the data to right storage class.

13. **Metrics :** There are two types of metrics for Amazon S3
    a. Storage metrics
    b. Request metrics

- Storage metrics are reported once per day and are provided to all customers at no additional cost. Request metrics are available at 1-minute intervals to quickly identify and act on operational issues and it is charged. Request metrics are reported for all object operations. Using the metrics information, you can deeper insights into how your objects are requested.
- Some of the commonly reported metrics are
  - The total number of HTTP requests made to an Amazon S3 bucket.
  - The number of HTTP GET requests made for objects in an Amazon S3 bucket.
  - The number of HTTP PUT requests made for objects in an Amazon S3 bucket.

  - The number of HTTP DELETE requests made for objects in an Amazon S3 bucket.
  - The number of HTTP HEAD requests made to an Amazon S3 bucket.
  - The number of HTTP POST requests made to an Amazon S3 bucket.
  - The number of bytes of data scanned with Amazon S3.
  - The number of HTTP requests that list the contents of a bucket.
  - The number of bytes downloaded for requests made to an Amazon S3 bucket.
  - The number of HTTP 4xx client error status code request.

14. **Inventory :** Amazon S3 inventory helps you to manage your storage. You can use it to list, audit and report on various configuration options such as replication and encryption status of your S3 objects. Amazon S3 inventory provides
    a. Comma-Separated Values (CSV)
    b. Apache Optimized Row Columnar (ORC) or
    c. Apache Parquet (Parquet) output files

- These files list your objects and their corresponding metadata on a daily or weekly basis for an S3 bucket. You can configure multiple inventory lists for a bucket.
- You can configure
  - what object metadata to include in the inventory,
  - whether to list all object versions or only current versions,
  - where to store the inventory list file output, and
  - whether to generate the inventory on a daily or weekly basis.
  - whether to encrypt the inventory list file itself
- Fig. 4.3.16 is a sample snapshot of various settings for objects that you can get in the inventory file.

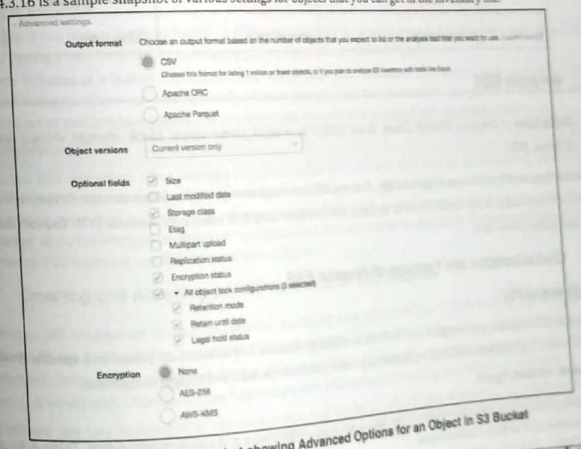


Fig. 4.3.16 : Screenshot showing Advanced Options for an Object in S3 Bucket

4.3.4 Amazon S3 Glacier

Definition : *Amazon S3 Glacier is a secure, durable, and extremely low-cost Amazon S3 cloud storage class for data archiving and long-term backup.*

- It is used for infrequently used archival and backup data. it provides high durability. To keep costs low yet suitable for varying needs, S3 Glacier provides three retrieval options that range from a few minutes to hours.
  1. **Expedited :** Expedited retrieval requests typically complete within 1 – 5 minutes.
  2. **Standard :** Standard retrieval requests typically complete within 3 – 5 hours.
  3. **Bulk retrievals :** Bulk retrieval requests typically complete within 5 – 12 hours.
- You can upload objects directly to S3 Glacier or use S3 Lifecycle policies to transfer data between any of the S3 Storage Classes for active data and S3 Glacier. A vault is a way to group archives together in Amazon S3 Glacier. You organise your data in Amazon S3 Glacier using vaults. Each archive is stored in a vault of your choice.

4.3.4(A) Comparison between S3 and Glacier

Table 4.3.2

Sr. No.	Comparison Attribute	S3	Glacier
1.	Purpose	Store frequently used data	Store archival data
2.	Cost	Standard per GB	Very Low cost per GB
3.	Static website	Can be hosted	Cannot be hosted
4.	Data retrieval	Immediately	Takes time

4.4 Amazon EBS

Definition : *Amazon Elastic Block Store (EBS) is a high performance block storage device used with Amazon EC2.*

Unlike Amazon S3 which was object storage, Amazon EBS is a block storage device. Block storage devices are typically used as hard disk and you can install OS on it. Amazon EBS provides hard disk volume for Amazon EC2. You can install and run OS and applications on it.

4.4.1 Characteristics and Features of Amazon EBS

1. High availability

Amazon EBS volumes are designed for 99.999% availability. Amazon EBS volumes are placed in a specific Availability Zone where they are automatically replicated to protect you from the failure of a single component.

2. Several volume types

You can choose from various volume types based on your performance and sizing requirements. Table 4.4.1 gives a few examples of various volume types.

Table 4.4.1

Sr. No.	Volume Type	Performance	Workload Type	Volume Size	Max IOPS / Volume	Cost
1.	EBS Provisioned IOPS SSD	Highest	I/O intensive	4 GB – 16 TB	64,000	$0.125/GB-month
2.	EBS General Purpose SSD	High	Low latency	1 GB – 16 TB	16,000	$0.10/GB-month
3.	Throughput Optimized HDD	Medium	Big data, log processing	500 GB – 16 TB	500	$0.045/GB-month
4.	Cold HDD	Low	Infrequently used data	500 GB – 6 TB	250	$0.025/GB-month
5.	EBS Magnetic	Least	Infrequently used data	1 GB – 1 TB	40 – 200	$0.05/GB-month

Caution : The pricing is per region and is subject to change. Consider the volume types and their details for your reference only.

3. Lifecycle Manager for EBS Snapshots

- Amazon EBS snapshots provide a mechanism to preserve the state of the hard disk volume at any given time. You can go back to that state at any point in time.
- Lifecycle Manager for EBS snapshots provides a simple, automated way to back up data stored on EBS volumes by ensuring that EBS snapshots are created and deleted on your chosen schedule. You no longer need to use scripts or other tools to comply with data backup and retention policies specific to your organization or industry. With lifecycle management, you can be sure that snapshots are cleaned up regularly and keep costs under control.

4. Elastic Volumes

- Imagine if you have a fixed size hard disk. How do you increase its capacity if it gets full? You need to do several things from buying a new hard disk to reconfiguring it, copying data, etc. Difficult right?
- Elastic Volumes is a feature that allows you to easily adapt your volumes as the needs of your applications change. Elastic Volumes allows you to dynamically increase capacity, tune performance, and change the volume type with no downtime or performance impact. You can choose any size now and increase it later on as you require.

5. Encryption

Amazon EBS encryption offers seamless encryption of EBS data volumes, boot volumes and snapshots, eliminating the need to build and manage a secure key management infrastructure. EBS encryption enables data at rest security by encrypting your data volumes, boot volumes and snapshots using Amazon-managed keys or keys you create and manage. Also, the encryption occurs on the servers that host EC2 instances, providing encryption of data as it moves between EC2 instances and EBS data and boot volumes.

4.4.2 Creating and Attaching an EBS Volume to an EC2 Instance

- Amazon EBS volumes are created in a particular Availability Zone and can be from 1 GB to 16 TB in size. Once a volume is created, it can be attached to any Amazon EC2 instance in the same Availability Zone. Once attached, it will appear as a mounted device similar to any hard drive or other block device. At that point, the instance can interact with the volume just as it would with a local hard drive, formatting it with a file system or installing applications on it directly.
- A volume can only be attached to one instance at a time, but many volumes can be attached to a single instance. If an instance fails or is detached from an Amazon EBS volume, the volume can be attached to any other instance in that Availability Zone.

- Let's learn about the steps in detail.

Step 1 : Navigate to Amazon EC2 service console and click on Volumes under Elastic Block Store.

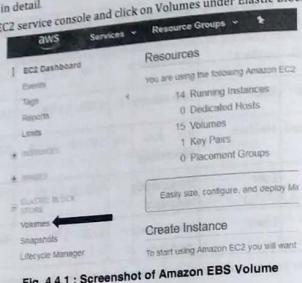


Fig. 4.4.1 : Screenshot of Amazon EBS Volume

Step 2 : Click on Create Volume. Choose the options, such as volume type, size, encryption, as per your requirements. Once chosen, click on Create Volume.

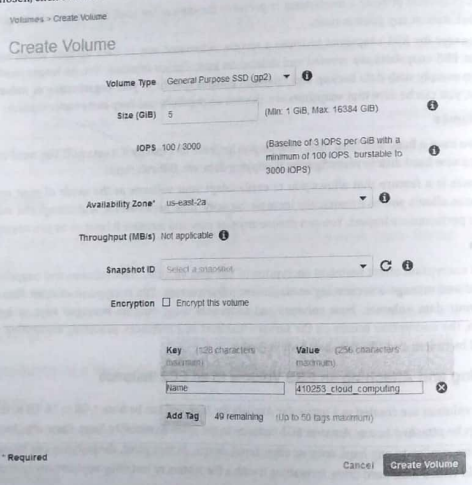


Fig. 4.4.2 : Screenshot showing Options to create an EBS Volume

This would create a new volume.

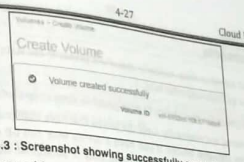


Fig. 4.4.3 : Screenshot showing successfully created EBS Volume

Step 3 : Choose the created volume and from Actions, choose Attach Volume to attach it to an Amazon EC2 instance.

Fig. 4.4.4 : Screenshot Showing navigation options for EBS Volume

Step 4 : Choose the instance to which you want to attach the volume and click on Attach.

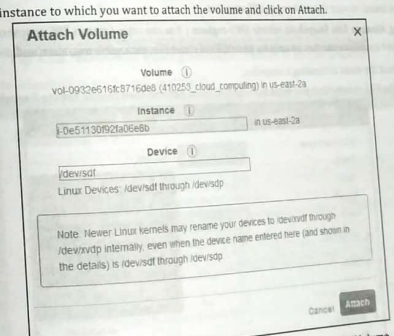


Fig. 4.4.5 : Screenshot to choose an Instance to attach an EBS Volume

Once attached, the created volume would be visible to your instance. You can format the volume and start using it as you would like it.

4.4.3 Amazon EBS Snapshots

Like photograph is a moment saved-in-time, similarly snapshots preserve the state of the disk at a given point in time.

Like a photograph, you can always go back and refer to it.

Definition : *Amazon EBS provides the ability to save point-in-time snapshots of your volumes to Amazon S3.*

- Amazon EBS Snapshots are stored incrementally. Only the blocks that have changed after your last snapshot are saved, and you are billed only for the changed blocks. If you have a device with 100 GB of data but only 5 GB has changed after your last snapshot, a subsequent snapshot consumes only 5 additional GB and you are billed only for the additional 5 GB of snapshot storage. When you delete a snapshot, you remove only the data not needed by any other snapshot. All active snapshots contain all the information needed to restore the volume to the instant at which that snapshot was taken.
- Snapshots can be used to
  - Attach to the newly created EC2 instances
  - Expand the size of a volume or
  - Move volumes across Availability Zones.
- When a new volume is created, you may choose to create it based on an existing Amazon EBS snapshot. In that scenario, the new volume begins as an exact replica of the snapshot.
- The following are key features of Amazon EBS Snapshots.
  1. **Immediate access to Amazon EBS volume data :** After a volume is created from a snapshot, there is no need to wait for all of the data to transfer from Amazon S3 to your Amazon EBS volume before your attached instance can start accessing the volume. Amazon EBS Snapshots implement lazy loading, so that you can begin using them right away.
  2. **Resizing Amazon EBS volumes :** You can resize an Amazon EBS volume. If you create a new volume based on a snapshot, you can specify a larger size for the new volume.
  3. **Sharing Amazon EBS Snapshots :** You can share Amazon EBS Snapshots. Users can create their own Amazon EBS volumes based on your shared Amazon EBS snapshot.
  4. **Copying Amazon EBS Snapshots across AWS regions :** You can copy Amazon EBS snapshots across AWS regions. The copied snapshots can then be used for geographical expansion, datacentre migration and disaster recovery.
- Let's learn about the steps of creating a snapshot.

Step 1 : Choose the EBS volume that you want to snapshot. From Actions, then choose Create Snapshot.

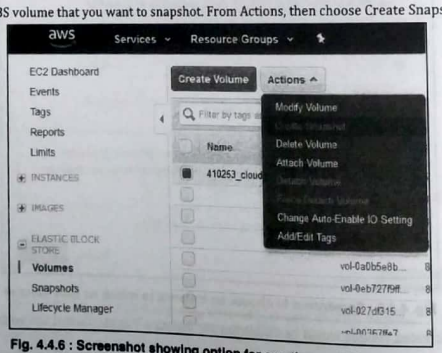


Fig. 4.4.6 : Screenshot showing option for creating an EBS Snapshot

Step 2 : Provide the snapshot name and tags. You can optionally encrypt the snapshot. Click on Create Snapshot.

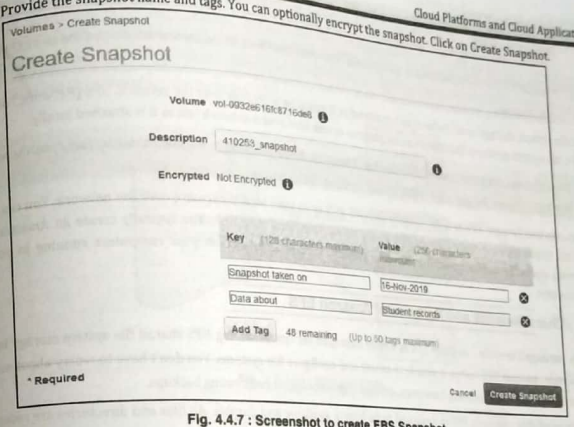


Fig. 4.4.7 : Screenshot to create EBS Snapshot

This would create a snapshot of the given volume.

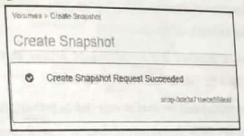


Fig. 4.4.8 : Screenshot showing successful creation of an EBS Snapshot

You can navigate to Snapshots to see the snapshot details.

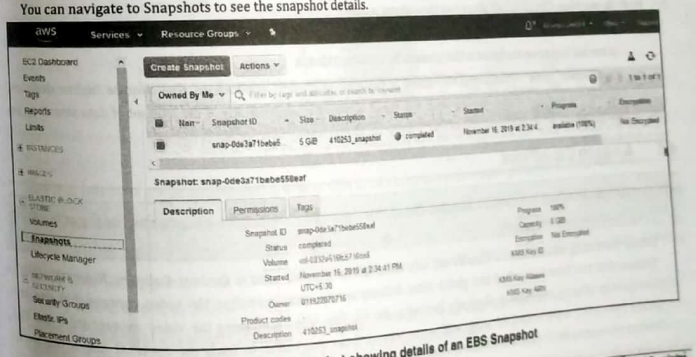


Fig. 4.4.9 : Screenshot showing details of an EBS Snapshot

4.5 Amazon EFS

Note : Amazon EFS is not prescribed in the syllabus. It is mentioned here for your understanding as the key storage service from Amazon.

- A file system storage could be local to a machine or it could be accessible over the network. NFS (Network File System) is a file system protocol that allows a computer to access files over a network just as it is attached locally.

✍ ***Definition :*** *Amazon Elastic File System (Amazon EFS) provides a simple, scalable, fully managed elastic NFS file system for use with AWS Cloud services (EC2) and on-premises resources.*

- You can use Amazon EFS as a file system on AWS EC2 or in your local datacentre over the network. You can use it to install applications or store files as you would do on your local hard disk. You typically create an Amazon EFS file system and then mount (attach) it on your Amazon EC2 instances or on your computers running in your local datacentre.

4.5.1 Characteristics and Features of Amazon EFS

1. **Fully managed service :** Amazon EFS is a fully managed service providing NFS shared file system storage for Linux workloads. Amazon EFS makes it simple to create and configure file systems. You don't have to worry about managing file servers or storage, updating hardware, configuring software, or performing backups.
2. **Highly available :** Amazon EFS is designed to be highly available and durable. All files and directories are redundantly stored within and across multiple Availability Zones in a region to prevent the loss of data from the failure of any single component. The distributed architecture of Amazon EFS provides data protection from an Availability Zone outages, system and component failures, and network connection errors.
3. **Storage class :** Amazon EFS offers a Standard and an Infrequent Access storage class. The EFS Infrequent Access (EFS IA) storage class is cost-optimized for files accessed less frequently.
4. **Encryption :** Amazon EFS provides encryption for data at rest and in transit. Data at rest is transparently encrypted using encryption keys managed by the AWS. Encryption of data in transit uses industry-standard Transport Layer Security (TLS) to secure network traffic.
5. **High throughput :** Amazon EFS offers two throughput modes -
   a. **Bursting mode :** In this throughput mode, the throughput scales with the size of the file system, dynamically bursting as needed to support the varied nature of many file-based workloads.
   b. **Provisioned mode :** Provisioned throughput is designed to support applications that require higher dedicated throughput than the default Bursting mode and can be configured independently of the amount of data stored on the file system.

4.6 Amazon CloudFront

Note : Amazon CloudFront is not prescribed in the syllabus. It is mentioned here for your understanding as the only content delivery service from Amazon.

- Before discussing about Amazon CloudFront, let's understand why you need a Content Delivery Network (CDN). Assume that you have subscribed to a global video delivery service such as Netflix. The video content that you are interested to stream and view is physically located in the US. You, when streaming the video content from the US, would have poor experience (the video will pause several times and buffer to fetch the content) due to network latency (slowness) for the amount of time it would take to stream the content from the US to India.

- To improve the user experience, the content providers use a content delivery network that locally stores a copy of the content in cache closer to the user's location. The user when requests the content, the content is served from the local cache instead of pulling it again from the original source location. That way, the user need not have to experience high network latency when interacting with the content and can have a better user experience.
- Figs. 4.6.1(a) and (b) summarises the purpose of CDN.

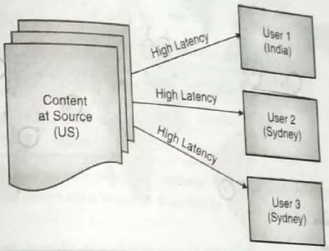


Fig. 4.6.1(a) : Without CDN

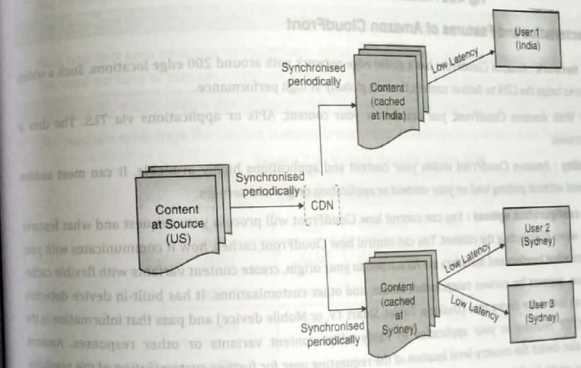


Fig. 4.6.1(b) : With CDN

- Now that you have got an understanding of what CDN is, let's jump into understanding Amazon CloudFront.

✍ ***Definition :*** *Amazon CloudFront is a fast Content Delivery Network (CDN) service that securely delivers data, videos, applications, and APIs to customers globally with low latency and high transfer speeds.*

- CloudFront is integrated with AWS – both physical locations that are directly connected to the AWS global infrastructure, as well as other AWS services.
- Amazon CloudFront has around 200 edge locations (as of Nov 2019) from where the content is distributed to the users globally. These edge locations are periodically updated from the content source.

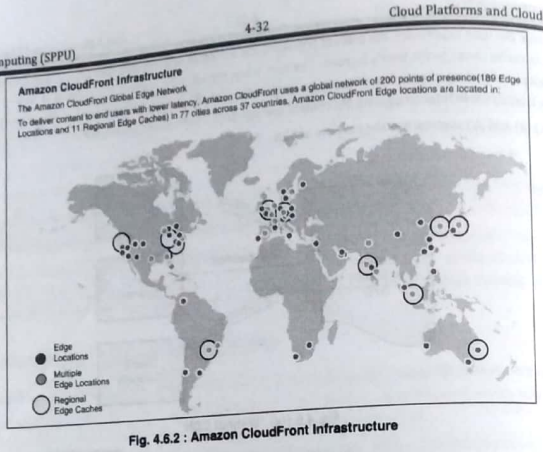


Fig. 4.6.2 : Amazon CloudFront Infrastructure

4.6.1 Characteristics and Features of Amazon CloudFront

1. **Global Edge Network :** Amazon CloudFront has a global edge network with around 200 edge locations. Such a widely spread networks helps the CDN to deliver content to users globally at high performance.
2. **Encryption :** With Amazon CloudFront, you can deliver your content, APIs or applications via TLS. The data is encrypted in transit.
3. **High Availability :** Amazon CloudFront makes your content and applications highly available. It can meet sudden spikes in demand without putting load on your content or application origin and servers.
4. **Various edge configuration options :** You can control how CloudFront will process your request and what features will be applied when distributing the content. You can control how CloudFront caches, how it communicates with your origin, customize what headers and metadata are forwarded to your origin, create content variants with flexible cache-key manipulation, support for various compression modes, and other customisations. It has built-in device detection using which it can detect the device type (Desktop, Tablet, Smart TV, or Mobile device) and pass that information in the form of new HTTP Headers to your application to easily adapt content variants or other responses. Amazon CloudFront can also detect the country-level location of the requesting user for further customisation of the response. For example, you could, by default, serve the same content in a regional language or in English based on where the user request is coming from.

4.7 Amazon SimpleDB

***Definition : Amazon SimpleDB** is a highly available NoSQL data store.*

Amazon SimpleDB provides a simple web services interface to create and store multiple data sets, query your data easily, and return the results. Your data is automatically indexed, making it easy to quickly find the information that you need. There is no need to pre-define a schema or change a schema if new data is added later. You can scale-out by creating new domains, rather than building out new servers.

4.7.1 How Amazon SimpleDB Works

- The data model used by Amazon SimpleDB makes it easy to store, manage and query your structured data. You can organise your dataset into domains and can run queries across all of the data stored in a particular domain. Domains are collections of items that are described by attribute-value (key-value) pairs.
- Let's take an example.
- Suppose you have the Table 4.7.1.

Table 4.7.1 : Sample Records in SimpleDB

Student-ID	Name	Department	Grade
A001	Andy	Computer	Pass
A002	Alex	Computer	Fail

- The whole table would be your domain named say "students". Individual students would be rows in the table or items in your domain. The Grade information would be described by column header (attributes). Values are in individual cells.
- Now imagine that you want to add a new student record. Unlike traditional database, you need to mention key value pair for each item. The API call would be something as following.

```
PUT (Student-ID, A003), (Name, Aaron), (Department, IT), (Grade, Pass)
```

- Note here that each item may be individually added, modified or deleted without changing other items in the domain. You need not have to manage a database schema. For example, you can add another student record as following.

```
PUT (Student-ID, A004), (Name, Ara), (Department, IT), (Grade, Pass), (Email, abc@new.com)
```

- The resultant table from the last two statements would look something like as shown in the Table 4.7.2.

Table 4.7.2 : Sample Records in SimpleDB

Student-ID	Name	Department	Grade	Email
A001	Andy	Computer	Pass	
A002	Alex	Computer	Fail	
A003	Aaron	IT	Pass	
A004	Ara	IT	Pass	abc@new.com

4.7.2 Amazon DynamoDB

***Definition :** Amazon DynamoDB is a key-value and document database that delivers extremely high performance at scale.*

- It is a fully managed, multi-region, multi-master, durable database with built-in security, backup and restore, and in-memory caching for internet-scale applications. DynamoDB can handle more than 10 trillion requests per day and can support peaks of more than 20 million requests per second.
- Many of the world's fastest growing businesses such as Lyft, Airbnb, and Redfin as well as enterprises such as Samsung, Toyota, and Capital One depend on the scale and performance of DynamoDB to support their mission-critical workloads. Hundreds of thousands of AWS customers have chosen DynamoDB as their key-value and document database for mobile, web, gaming, ad tech, IoT, and other applications that need low-latency data access at any scale.

Characteristics and Features of Amazon DynamoDB

1. **Data models :** DynamoDB supports both key-value and document data models. This enables DynamoDB to have a flexible schema, so each row can have any number of columns at any point in time. This allows you to easily adapt the tables as your business requirements change, without having to redefine the table schema as you would in relational databases.
2. **High-performance :** DynamoDB is a key-value and document database that can support tables of virtually any size with horizontal scaling. This enables DynamoDB to scale to more than 10 trillion requests per day with peaks greater than 20 million requests per second, over petabytes of storage.
3. **Automated global replication with global tables :** DynamoDB global tables replicate your data automatically across your choice of AWS Regions and automatically scale capacity to accommodate your workloads. With global tables, your globally distributed applications can access data locally in the selected regions to get single-digit millisecond read and write performance.
4. **Lower administrative overhead :** You can setup DynamoDB instances without requiring to setup the underlying infrastructure first. You just pay and use the database instances as per your requirements. There are no servers to provision, patch, or manage, and no software to install, maintains, or operate. DynamoDB automatically scales tables to adjust for capacity and maintains performance with zero administration. Availability and fault tolerance are built in, eliminating the need to architect your applications for these capabilities.
5. **On-demand backup and restore :** On-demand backup and restore allows you to create full backups of your DynamoDB tables' data for data archiving, which can help you meet your corporate and governmental regulatory requirements. You can back up tables from a few megabytes to hundreds of terabytes of data and not affect performance or availability to your production applications.

4.8 Microsoft Azure

Microsoft Azure is one of the major public cloud service providers. It offers over 600 cloud services that you can use for your various computing requirements. It has 60+ regions and multiple availability zones within each as depicted in the following snapshot (as of May 2021).

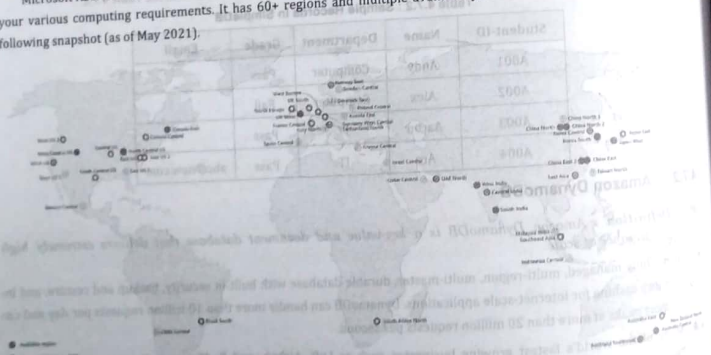


Fig. 4.8.1 : Snapshot of Microsoft Azure Cloud (May 2021)

Note : The concepts such as regions and availability zones remain consistent across any cloud service provider. So, you can very well reuse the definitions and diagrams from AWS when explaining about Microsoft Azure. Also, like any other cloud service provider, Azure adds multiples services every year. Detailing about each service is out of scope of this book. You can go to https://azure.microsoft.com to learn more about Azure services and also create a free account to play around various Azure services yourself.

4.8.1 Azure Virtual Machines (Azure VM)

✍ ***Definition :*** *Azure Virtual Machines (Azure VM) is a web service that provides secure, resizable compute capacity (virtual machines) in the cloud.*

- It is one of the most used services within Microsoft Azure. It gives you Linux or Windows based virtual machines that you can use to run your workloads. It generally provides the same features, such as auto scaling, multiple instance types, static or dynamic IP address, etc., as any other cloud service provider.
- The Table 4.8.1 gives a few examples of instance types available in Azure.

Table 4.8.1 : Examples of Instance Types available in Azure

Instance Family	Instance Type Example	Hardware Configuration	Price per Hour
General Purpose	B1MS	1 CPU, 2 GB RAM	$0.0208
General Purpose	B16MS	16 CPU, 64 GB RAM	$0.666
General Purpose	D64s v3	64 CPU, 256 GB	$3.072
Compute Optimized	F72s v2	72 CPU, 144 GB	$3.06
GPU Instances	NC24	24 CPU, 224 GB	$3.60
Memory Optimized	M128m	128 CPU, 3,892 GB	$26.688

Caution : The pricing is per region and is subject to change. Consider the instance family and their details for your reference only.

4.8.2 Blob Storage

Blob is a short form for Binary Large Objects.

✍ ***Definition :*** *Blob Storage is an object storage service.*

It provides scalable and cost-effective object storage. You can store files, music, video, images or anything else. It provides similar features, such as 11 9's of high durability, multiple storage classes, security and access control, as any other cloud service provider.

Table 4.8.2 : Summary of Various Storage Classes available for Blob Storage

Storage Class	Purpose	Price per GB
Premium	For frequently accessed data	$0.15
Hot	For less frequently accessed data	$0.0184
Cool	For rarely accessed data	$0.01
Archive	For data archiving	$0.0010

The Table 4.8.2 gives a quick summary of various storage classes available for Blob storage.

Caution : The pricing is per region and is subject to change. Consider the storage class and their details for your reference only.

4.8.3 Database Services(SQL Azure)

Definition : *Azure Database Services enable you to run database in the cloud without you having to manage the underlying database software.*

- Azure Database Services make it easy for you to quickly setup a database instance without you having to first setup a virtual machine and then install and configure a database software on it yourself. It provides a cost-efficient way to run database in the cloud.
- It provides similar features, such as lower administrative overhead, support for a variety of database software, scalability, security and monitoring, as any other cloud service provider.

4.8.4 Azure Monitor

Definition : *Azure Monitor is a monitoring service using which you can continuously monitor your Azure cloud resources.*

It helps you to,

- Monitor your applications
- Monitor your infrastructure
- Monitor your network
- Get a unified view of operational health

Azure Monitor collects monitoring and operational data in the form of logs, metrics, and events from various sources such as OS, Azure services, etc. It then analyses the collected data to produce actionable information. It can send notifications or can take appropriate actions based on the analysis.

4.9 Windows Azure Platform Appliance

Note : This was an appliance released by Microsoft Azure way back in 2010. It is now obsolete. It is just covered here for sake of reference as per the syllabus. The latest service from Microsoft is Azure Stack. Azure Stack is a portfolio of products that extend Azure services and capabilities to your environment of choice from the datacentre to edge locations and remote offices. You can build and deploy hybrid and edge computing applications and run them consistently across location boundaries.

- The Windows Azure Platform Appliance consists of Windows Azure, SQL Azure and a Microsoft-specified configuration of network, storage and server hardware. Service providers, governments and large enterprises who would, for example, invest in a 1000 servers at a time, will be able to deploy the Windows Azure platform on their own hardware in their datacentre. Microsoft Windows Azure Platform Appliance is optimised for scale out applications and datacentre efficiency across hundreds to thousands to tens-of-thousands servers.
- The main benefit of the appliance is that it provides the benefits of the Windows Azure platform with greater physical control, geographic proximity, regulatory compliance and data sovereignty. The appliance runs only on network, storage and server hardware that meets the Windows Azure platform reference specifications. Microsoft had invested significant engineering resources to ensure that the hardware required by the appliance is optimized to enable service availability, automated management and power, cooling and operational efficiency across tens of thousands of servers.

4.10 Overview of OpenStack Architecture

Definition : *OpenStack is a cloud operating system that controls large pools of compute, storage, and networking resources throughout a datacentre.*

It is used for managing various resources such as CPU, memory, networking and storage. Beyond standard infrastructure-as-a-service functionality, additional components provide orchestration, fault management and service management amongst other services to ensure high availability of user applications.

4.10.1 Features of OpenStack

Some of its major features provided by OpenStack are as following,

1. **Compatibility with Cloud Service Providers**

OpenStack APIs are compatible with cloud service providers such as AWS. Your applications can then seamlessly run either on OpenStack or AWS (or any other cloud service provider).

2. **Modularity**

- OpenStack is broken up into services to allow you to plug and play components depending on your needs. Its services are broken up into multiple categories as following.

Compute	Hardware Lifecycle
Storage	Networking
Shared Services	Orchestration
Workload Provisioning	Application Lifecycle
API proxies	Web Frontend
Monitoring tools	

3. **Open-source**

OpenStack is a fully functional open-source software that can be used to build cloud computing environment.

4. **Security**

OpenStack provides robust security frameworks for authentication and authorization. It provides API client authentication, service discovery, and distributed multi-tenant authorization by implementing OpenStack's Identity API. It supports LDAP, OAuth, OpenID Connect, SAML and SQL.

5. **Management and Automation**

It provides a web interface and APIs for managing and automating tasks with the virtual resources.

4.10.2 Components of OpenStack and its Architecture

- OpenStack is a broad set of tools and components. Some components are mandatory whereas others are optional and can be used as per your requirements.

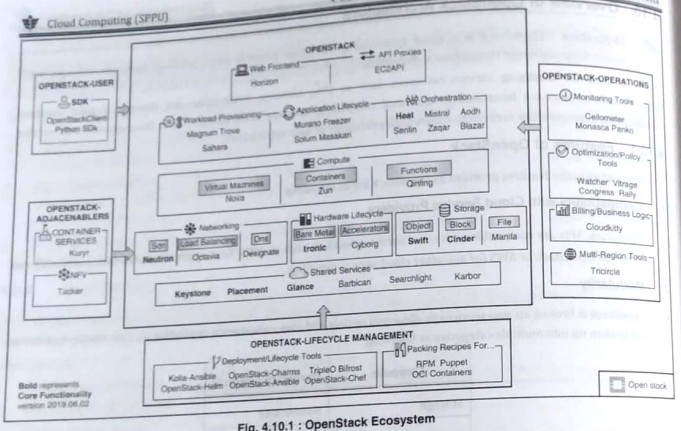


Fig. 4.10.1 : OpenStack Ecosystem

- Table 4.10.1 summarises the various components and their usage (you need not remember all of them! This is just for your reference to understand the scope of various services available as of Dec 2019).

Table 4.10.1

Service Category	Service Name	Service Usage
Compute	NOVA	Compute Service
	ZUN	Containers Service
	QINLING	Functions Service
Hardware Lifecycle	IRONIC	Bare Metal Provisioning Service
	CYBORG	Lifecycle management of accelerators
Storage	SWIFT	Object Store
	CINDER	Block Store
	MANILA	Shared File system
Networking	NEUTRON	Networking
	OCTAVIA	Load Balancer
	DESIGNATE	DNS Service

Service Category	Service Name	Service Usage
Shared Services	KEYSTONE	Identity Service
	PLACEMENET	Placement Service
	GLANCE	Image Service
	BARBICAN	Key Management
	KARBOR	Application Data Protection as a Service
	SEARCHLIGHT	Indexing and Search
Orchestration	HEAT	Orchestration
	SENLIN	Clustering Service
	MISTRAL	Workflow Service
	ZAQAR	Messaging Service
	BLAZAR	Resource reservation service
	AODH	Alarming Service
Workload Provisioning	MAGNUM	Container Orchestration Engine Provisioning
	SAHARA	Big Data Processing Framework Provisioning
	TROVE	Database as a Service
Application Lifecycle	MASAKARI	Instances High Availability Service
	MURANO	Application Catalog
	SOLUM	Software Development Lifecycle Automation
	FREEZER	Backup, Restore, and Disaster Recovery
API Proxies	EC2API	EC2 API proxy
Web Frontend	HORIZON	Dashboard
Monitoring Tools	CEILOMETER	Metering & Data Collection Service
	PANKO	Event, Metadata Indexing Service
	MONASCA	Monitoring
Optimization/policy tools	WATCHER	Optimization Service
	VITRAGE	Root Cause Analysis service
	CONGRESS	Governance
	RALLY	Benchmark service

Service Category	Service Name	Service Usage
Billing / Business Logic	CLOUDKITTY	Billing and chargebacks
Multi-Region Tools	TRICIRCLE	Networking Automation for Multi-Region Deployments
Containers	KURYR	OpenStack Networking integration for containers
NFV	TACKER	NFV Orchestration

- Based on the above ecosystem, a simple architecture for general cloud computing could be as shown in Fig. 4.10.2.

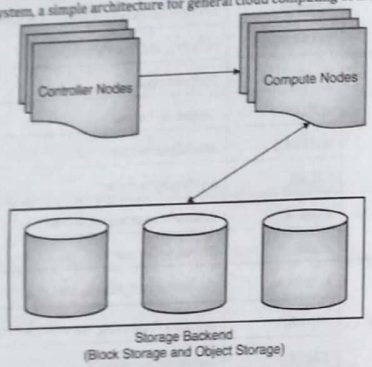


Fig. 4.10.2 : Architecture for General Cloud Computing

1. **Controller Nodes :** The controller node runs the
   - Identity service.
   - Virtual image service.
   - Workload placement service.
   - Management portions of compute.
   - Management portion of networking.
   - Various networking agents.
   - The overview dashboard.

   It also includes supporting services such as an SQL database, message queue, and NTP.

2. **Compute Nodes :** The compute node runs the hypervisor and the virtual machines on it. By default, compute uses the KVM hypervisor. The compute node also runs a networking service agent that connects virtual machine instances to virtual networks and provides various networking services such as firewalls. Compute Nodes are managed by OpenStack Compute Service called Nova.

> ***Definition :*** *Nova is the OpenStack project that provides a way to provision compute instances (virtual servers). Nova supports creating virtual machines, baremetal servers and has limited support for system containers.*

Nova runs as a set of daemons on top of existing Linux servers to provide its service.

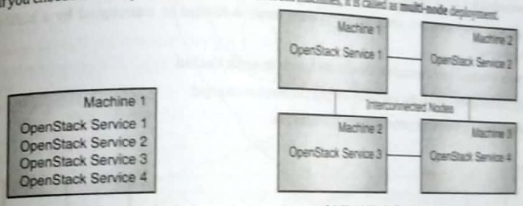
3. **Storage Backend :** The storage backend provides storage services.
   - **Block Storage :** It provides the disk that can work as a block device and can be used for running guest OS and the system on a virtual machine. The OpenStack implementation of block storage is named Cinder.
   - **Object Storage :** It provides the disk that can work as object store. You can store various objects such as files, videos, music, images, etc. Note here that the object storage can only be used for storing objects. You cannot run OS on it. The OpenStack implementation of object storage is named Swift.

Like any other cloud service provider, OpenStack provides various services that you can use in your own datacentre and run a private cloud.

4.10.3 Mode of Operations

- As you have learnt so far, there are various OpenStack services. Based on your requirements, you can appropriately pick and choose which services you want to run. The services that you have chosen can then be made to run either all on a single machine or a set of different machines. Based on how you want to run those services, there are two modes of operations.

1. If you choose to run all OpenStack services on a single machine, it is called as **single-node** deployment.
2. If you choose to run OpenStack services on different machines, it is called as **multi-node** deployment.



(a) Single Node Deployment **(b) Multi-Node Deployment**

Fig. 4.10.3

- On a multi-node setup, the networking service, nova connects all the nodes (machines) together and then the OpenStack services can communicate with each other.

Comparison between Modes of Operation

Table 4.10.2

Sr. No.	Comparison Attribute	Single node deployment	Multi-node deployment
1.	Used for	Testing Environment only	Real (production) environment
2.	Single point of failure	Possible	No, since there are many nodes
3.	Scalable	No	Yes
4.	Number of IP addresses required	1	As many as nodes in the deployment
5.	Performance	Low	High
6.	Complexity	Low	High

4.11 Cloud Computing Applications

Let's learn about some of the cloud computing applications. I will be giving you case studies publicly posted on various cloud providers' websites to give you a real-life sense of applications.

4.11.1 Healthcare: ECG Analysis in the Cloud

- Quite a few healthcare companies are using cloud computing to transform their medical services. Healthcare and life science organisations are reinventing how they collaborate, make data driven clinical and operational decisions, enable precision medicine, and decrease the cost of care.
- An electrocardiogram (ECG) is a simple test that can be used to check your heart's rhythm and electrical activity. Sensors attached to the skin are used to detect the electrical signals produced by your heart each time it beats. These signals are recorded by a machine and are looked at by a doctor to see if they are unusual. An ECG may be requested by a heart specialist (cardiologist) or any doctor who thinks you might have a problem with your heart. An ECG is often used alongside other tests to help diagnose and monitor conditions affecting the heart. It can be used to investigate symptoms of a possible heart problem, such as chest pain, palpitations (suddenly noticeable heartbeats), dizziness and shortness of breath.
- An ECG can help detect:
  - **Arrhythmias :** where the heart beats too slowly, too quickly, or irregularly
  - **Coronary heart disease :** where the heart's blood supply is blocked or interrupted by a build-up of fatty substances
  - **Heart attacks :** where the supply of blood to the heart is suddenly blocked
  - **Cardiomyopathy :** where the heart walls become thickened or enlarged
- Following is how a typical heart pattern through ECG looks like.

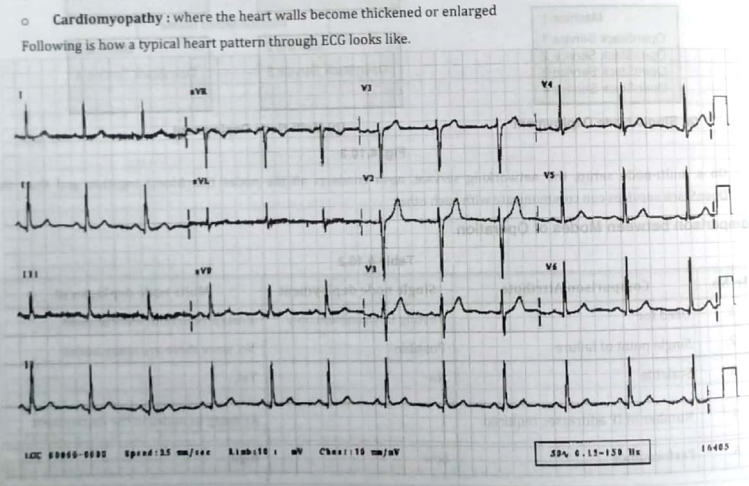


Fig. 4.11.1

- A series of ECGs can also be taken over time to monitor a person already diagnosed with a heart condition or taking medication known to potentially affect the heart.
- ECG technology has been around for more than 100 years and is still the most frequently used test for monitoring the heart. Globally, over 300 million resting ECG tests are done each year. However, access to the technology and monitoring has been limited to healthcare facilities. To make an ECG accessible and affordable for everyone, QT Medical developed compact, 12-lead ECG devices that can be used in the home and transmit data wirelessly to clinicians anywhere using cloud services. QT Medical is a MedTech startup focused on combating heart disease using state-of-the-art electrocardiogram (ECG) technologies. Through its innovative 12-lead ECG system (PCA 500) and exclusive home testing service (Xpress ECG), patients can easily get a hospital-grade ECG test without leaving their homes.

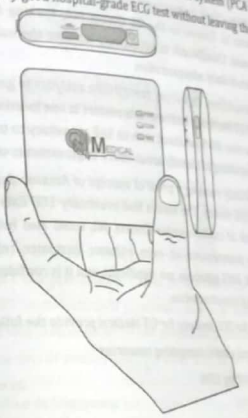


Fig. 4.11.2

- Demand for this kind of technology is growing rapidly and is expected to be worth $3.1 billion globally by 2029. Their goal is nothing short of bringing wireless ECG technology to everyone wherever one needs it.
- QT Medical looked to implement SAP Business One in a short turnaround time, though it had neither the capital to spare for building a platform nor the staff to manage the SAP solution. It was obvious that it needed to go to the cloud. QT Medical turned to Amazon Web Services (AWS), which was already running the backend infrastructure for its ECG solution. AWS had always delivered the reliable performance required for their ECG technology. AWS infrastructure is stable and connectivity between offices is reliable as needed for a medical service.
- QT medical uses Amazon Elastic Compute Cloud (Amazon EC2) instances for highly scalable compute performance, making it easy to add capacity as the business grows, and Amazon Simple Storage Service (Amazon S3) for low-cost data replication. Running SAP on AWS helped QT Medical overcome the additional costs associated with building its own SAP platform. The total cost of ownership (TCO) for running SAP on AWS compared to an on-premises infrastructure was about 15 percent less. Saving 15 percent is important for any business, and all the more so for a startup where each cent matters. Moreover, the TCO saving will increase over time as QT Medical scales the SAP solution on AWS, while avoiding the expense of managing more hardware.

- Although the savings are significant, they aren't the most important benefit for QT Medical. Startups aim to expand quickly, gain first-mover advantage wherever possible, and establish a firm foothold in a country. By relying on AWS Global Cloud Infrastructure, QT Medical can immediately replicate its SAP infrastructure to the nearest AWS Region, enabling rapid expansion to new locations. Using AWS gives it the agility to grow fast in what could be a winner-takes-all local market.
- A major advantage of AWS is that the infrastructure behind an office can be scaled down when the location isn't open, saving resources that can be reinvested in other areas of the business. This capability came to the fore when QT Medical moved its US office to Diamond Bar, California, in 2019. eCloudvalley manages its AWS infrastructure and through the eCloudvalley Atlas cloud management platform, it can temporarily shut down the SAP Business One system supporting operations in Taiwan or the US during non-working hours to reduce unnecessary expenses. eCloudvalley also uses Amazon CloudWatch to proactively monitor the use of QT Medical's Microsoft SQL Server database to reduce the burden of daily administration.
- With AWS, QT Medical has streamlined its journey toward data analytics to gain insight for improved decision making. With SAP Business One, all of QT Medical's business data resides in one location as the organisation continues to grow. The startup can then integrate new AWS solutions with its SAP repository to transform and model operational data on finance, inventory, and manufacturing to identify areas where improvements could be made.
- Furthermore, QT Medical is already running a proof of concept of Amazon SageMaker to add machine learning (ML) capabilities to its QT Medical ECG device. The idea is that eventually ECG data combined with ML will help clinicians better calculate individuals' risk of cardiovascular disease. ML could also help QT Medical improve manufacturing processes through preventive maintenance or even enhance customer relationships through analysis of their behaviour. The possibilities with AWS solutions are significant, and it is confident that it is setting the right foundation to expand and kick-start its digital transformation.
- So, to summarise, cloud services for ECG analysis for QT Medical provide the following benefits.
  - Provide high performance and elastic computing resources
  - Reduce the cost of running SAP by 15%
  - Enable global expansion
  - Optimise operational efficiency
  - Speed up data analytics development

4.11.2 Biology: Protein Structure Prediction

- If you have read about Covid-19 deeply, chances are that you are already aware of what protein analysis is. Proteins are molecular machines that perform many functions we associate with life. They sense the environment (e.g. in taste and smell), perform work (e.g. muscle contraction and breaking down food), and play structural roles (e.g. your hair). They are made of a linear chain of chemicals called amino acids that, in many cases, spontaneously "fold" into compact, functional structures. Much like any other machine, it's how a protein's components are arranged and move that determines the protein's function. In this case, the components are atoms.
- Viruses also have proteins that they use to suppress our immune systems and reproduce themselves. To help tackle coronavirus, it is important to understand how these viral proteins work and how we can design therapeutics to stop them. There are many experimental methods for determining protein structures. While extremely powerful, they only reveal a single snapshot of a protein's usual shape. But proteins have lots of moving parts. So, it is really important to see the protein in action. The structures we can't see experimentally may be the key to discovering a new therapeutic. Using football as an analogy for the experimental situation, it's as if you could only see the players lined up for the snap (the single arrangement the players spend the most time in) and were blind to the rest of the game. Seeing a single structure of a protein is like seeing players lined up for the snap in football. Important information, but a lot missing too.

- The protein structure, in the Fig. 4.11.3, shows a sphere for each atom and the arrows highlight the one drug binding site in this protein.

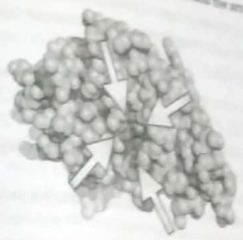


Fig. 4.11.3

- The study of protein localisation is important to comprehend the function of protein, which is essentially its structure, function, and regulate the body's tissues and organs. Protein localisation has great importance for drug design and other applications. For example, we can investigate methods to disrupt the binding of the spiky (S) protein of the SARS-Cov-2 virus. The binding of the S1 protein to the human receptor ACE2 is the mechanism which led to the (COVID-19 pandemic. It also plays an important role in characterizing the cellular function of hypothetical and newly discovered proteins.
- Protein structure analysis requires a lot of computing power to run machine learning based analytical models. One such model is ProtBERT. ProtBERT is a pretrained model on protein sequences using a masked language modelling objective. It is based on the BERT model, which is pretrained on a large quantity of protein sequences in a self-supervised fashion. This means it was pretrained on the raw protein sequences only, with no humans labelling them in any way (which is why it can use lots of publicly available data) with an automatic process to generate inputs and labels from those protein sequences.

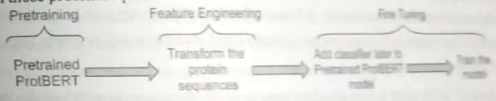


Fig. 4.11.4

- The protein sequence dataset has the following fields.
  - **id :** Unique identifier given each sequence in the dataset.
  - **sequence :** Protein sequence. Each character is separated by a space. This is useful for the BERT tokenizer.
  - **sequence_length :** Character length of each protein sequence.
  - **location :** Classification given each sequence. The dataset has 10 unique classes (subcellular localisation).
  - **is_train :** Indicates whether the record should be used for training or test. Is also used to separate the dataset for training and validation.

4.11.3 Geosciences : Satellite Image Processing

- Satellite images (also Earth observation imagery, spaceborne photography, or simply satellite photo) are images of Earth collected by imaging satellites operated by governments and businesses around the world. Satellite imaging companies sell images by licensing them to governments and businesses such as Apple Maps and Google Maps.

- Satellite images have many applications in meteorology, oceanography, fishing, agriculture, biodiversity conservation, forestry, landscape, geology, cartography, regional planning, education, intelligence and warfare. There are also elevation maps, usually made by radar images. Image interpretation and analysis of satellite imagery is conducted using specialised remote sensing software.
- There are five types of resolution when discussing satellite imagery in remote sensing as follows.
  1. **Spatial resolution :** It is defined as the pixel size of an image representing the size of the surface area being measured on the ground, determined by the sensors' instantaneous field of view.
  2. **Spectral resolution :** It is defined by the wavelength interval size (discrete segment of the Electromagnetic Spectrum) and number of intervals that the sensor is measuring.
  3. **Temporal resolution :** It is defined by the amount of time (e.g. days) that passes between imagery collection periods for a given surface location.
  4. **Radiometric resolution :** It is defined as the ability of an imaging system to record many levels of brightness (contrast for example) and to the effective bit-depth of the sensor (number of grayscale levels) and is typically expressed as 8-bit (0–255), 11-bit (0–2047), 12-bit (0–4095) or 16-bit (0–65,535).
  5. **Geometric resolution :** It refers to the satellite sensor's ability to effectively image a portion of the Earth's surface in a single pixel and is typically expressed in terms of Ground sample distance, or GSD. GSD is a term containing the overall optical and systemic noise sources and is useful for comparing how well one sensor can "see" an object on the ground within a single pixel. For example, the GSD of Landsat is ≈30m, which means the smallest unit that maps to a single pixel within an image is ≈30m x 30m. The latest commercial satellite (GeoEye 1) has a GSD of 0.41 m. This compares to a 0.3 m resolution obtained by some early military film based Reconnaissance satellite such as Corona.
- The resolution of satellite images varies depending on the instrument used and the altitude of the satellite's orbit. For example, the Landsat archive offers repeated imagery at 30 meter resolution for the planet, but most of it has not been processed from the raw data. Landsat 7 has an average return period of 16 days. For many smaller areas, images with resolution as high as 41 cm can be available. Satellite imagery is sometimes supplemented with aerial photography, which has higher resolution, but is more expensive per square meter. Satellite imagery can be combined with vector or raster data in a GIS provided that the imagery has been spatially rectified so that it will properly align with other data sets.
- As you understand, satellite image processing requires massive computing power as well storage for continuously working on the received satellite images. With cloud computing, satellite image processing is reaching new possibilities. Some of the case studies are as following.
  - Mantle Labs took advantage of artificial intelligence (AI) to build its flagship Geobotanics platform, which analyses data feeds from satellites in space. This data is used to develop agricultural risk and decision-making indices specific to industry sectors linked to agriculture: banks, insurers, food retailers, crop input providers, and commodity companies that either trade or manage farms directly. The insurance index, for example, monitors crop growth to compare against historical patterns, also factoring in external data on past and projected weather conditions to shape the index.

    By subscribing to Geobotanics, banks and insurers can make more informed decisions on rates for agri-loans and farming insurance premiums. Farmers benefit from new and improved sustainable finance solutions, and commodity companies, food retailers, and other members of the agricultural supply chain can better plan their businesses and prevent food shortages.

Mantle Labs uses Amazon Web Services (AWS) to automate processing of satellite data from 500 million hectares of agriculture spanning six continents. By building Geobotanics on AWS, Mantle Labs has drastically reduced the manual effort required to process high volumes of data using its AI algorithms. The company has access to on-demand infrastructure that can be scaled up and down seamlessly, and it can stay at the cutting edge of technology while controlling costs.

- Every week, Mantle Labs processes 32–50 TB of data from satellites around the globe. The availability of on-demand, large-scale infrastructure was critical for this platform's success. AWS is able to provide that kind of capacity without going through any provisioning bureaucracy. During peak data processing, Mantle Labs can easily run 1,000 or more virtual machines simultaneously. Accessing satellite data to initiate processing is likewise seamless. Imagery downloaded from the National Aeronautics and Space Administration (NASA) and other space agencies are available to AWS customers via Amazon Simple Storage Service (Amazon S3) buckets.

  When weather threatens drilling rigs, refineries, and other energy facilities, oil and gas companies want to move fast to protect personnel and equipment. Firms that trade commodity shares in oil, precious metals, crops, and livestock, the weather can significantly impact their buy-sell decisions. To limit damage, these companies need the earliest possible notice before a major storm strikes. That's the challenge Maxar Technologies set out to solve.

- Historically, many industries have relied on reports generated by the on-premises supercomputer operated by the National Oceanic and Atmospheric Administration (NOAA). However, the weather predictions take an average of 100 minutes to process global data. Over time, many companies began to realise they would require much faster weather warnings to protect their interests. Similar to how NASA has expanded its partnerships with private firms to acquire commercial space hardware and services, the processing and delivery of critical weather data products could also be effectively commercialized.

- To resolve this issue, Maxar sought to significantly reduce the time needed to generate numerical weather predictions. Its data scientists, engineers, and DevOps team decided to build a high performance computing (HPC) solution to deliver forecasts in half the time of the NOAA supercomputer. HPC on AWS could provide an environment that balances performance, cost, and manageability. Maxar designed a cloud HPC cluster with 234 Amazon EC2 instances capable of producing a numerical weather prediction forecast in roughly 53 minutes, just about half the 100 minutes that the NOAA supercomputer takes to complete the same forecast.

- This accomplished Maxar's initial performance goal, so the team set its eyes on enhancing the design to reduce cost. It further shortened the forecast time—from 53 to 42 minutes, a 22 percent decrease. The team's new configuration can now produce a forecast 58 percent faster than NOAA's supercomputer. Additional testing and optimisation with AWS revealed Maxar could complete a forecast in under 30 minutes. With further system tuning, Maxar projects it can cut its processing time by an additional 25 percent.

- Commodity-crop farmers across the United States often depend on satellite images of their fields to get an updated view of the health of their crops. The resolution of these images, though, is not high enough for farmers to get the most accurate picture of their fields. In fact, for many specialty-crop growers, satellite images are somewhere between useless and misleading. TerrAvion is changing that. The company uses airplanes and drones to obtain full-frame and thermal images from high-resolution cameras. TerrAvion gives farmers, retailers, agronomists, and ag distributors the best possible pictures through OverView, the company's core subscription service. Its pictures offer resolutions of 9 or 18 centimeters per pixel, which satellite can't do. Using such high-resolution images, farmers can more accurately view the health of a plant.

- To accommodate its increasing image-storage needs, the company began using Amazon Simple Storage Service (Amazon S3) to ingest the aerial images each day. Amazon S3 is highly scalable and reliable. It uses S3 to store aerial images as raw data. TerrAvion also uses Amazon S3 Glacier for long-term image-data storage and AWS Lambda to automatically identify when image data has been uploaded to the processing application. The company partners with several analytics technology companies to integrate image analysis and machine learning for farmers, who can turn that analysis into business insights about crop yields.

4.11.4 AWS Ground Station

- AWS Ground Station is a fully managed service that lets you control satellite communications, process data, and scale your operations without having to worry about building or managing your own ground station infrastructure. Satellites are used for a wide variety of use cases, including weather forecasting, surface imaging, communications, and video broadcasts. Ground stations form the core of global satellite networks. With AWS Ground Station, you have direct access to AWS services and the AWS Global Infrastructure including a low-latency global fibre network. For example, you can use Amazon S3 to store the downloaded data, Amazon Kinesis Data Streams for managing data ingestion from satellites, and Amazon SageMaker for building custom machine learning applications that apply to your data sets. You can save up to 80% on the cost of your ground station operations by paying only for the actual antenna time used, and relying on the global footprint of ground stations to download data when and where you need it. There are no long-term commitments, and you gain the ability to rapidly scale your satellite communications on-demand when your business needs it.

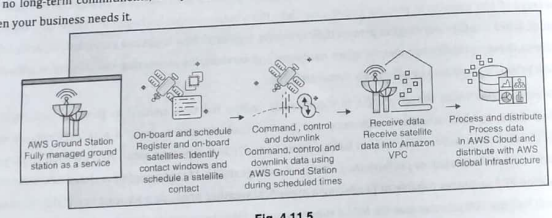


Fig. 4.11.5

- AWS Ground Station currently supports LEO and MEO (Medium Earth Orbit) satellites. Because of their orbital periods, these satellites are visible from the ground only for a few minutes during each pass, and communication is only possible when they are within line of sight of a ground station. AWS Ground Station establishes contact with the satellites, and then receives, demodulates, and decodes its radiofrequency signals. It then pushes the decoded data to the receiver Amazon Elastic Compute Cloud (Amazon EC2) instance as a VITA 49 stream.
- A data capture application running on the receiver EC2 instance ingests the incoming VITA49 stream. The payload from within each VITA49 packet is extracted and combined into a raw data file. The raw data file is held locally and also pushed to Amazon S3 to be reprocessed as needed at a later date.

4.12 Business and Consumer Applications: CRM and ERP, Social Networking

CRM (Customer Relationship Management), ERP (Enterprise resource planning), and Social Networking solutions require high-performance computing and storage as well. Cloud providers have several services in this area. For example, AWS has the following services.

Table 4.12.1

Category	Service description	AWS service
Line of Business Applications	Easy to use omnichannel cloud contact center	Amazon Connect
	Multichannel marketing communications	Amazon Pinpoint
Productivity Applications	Build apps for managing your team's work with no-code	Amazon Honeycode
	Frustration - free meetings , video call, and chat	Amazon Chime
	Secure enterprise document storage and sharing	Amazon WorkDocs
	Secure email and calendaring	Amazon WorkMail
	Empower your organization with Alexa	Alexa for Business
Communication Developer Services	Real-time messaging, audio, video, and screen sharing	Amazon Chime SDK
	High-scale inbound and outbound email	Amazon Simple Email Service (SES)
	Flexible mobile SMS and push notifications	Amazon Pinpoint APIs
	Cost-effective SIP trunking and advanced telephony features	Amazon Chime Voice Connector
	Secure file collaboration and management	Amazon WorkDocs SDK

- Some of the case studies in these areas are as following.
  - Symbee Connect is a Global Contact Center Solution that integrates directly with Amazon Connect providing advanced Enterprise OmniChannel features. When deployed with Amazon Connect, it supports Voice (In/Out/Direct Agent), Callback, Preview Dial, Email, WebChat, SMS, Social Messaging and OmniChannel Reporting. Symbee Connect also has core features such as a Headset Integration(Jabra and Poly), Supervisor Client, UC Client, Voicemail (Agent and Queue), Presence, Agent Stats, Work Codes, Task History and many more Enterprise features.

    Symbee Connect Fusions allow for CRM Solutions such as Salesforce, ServiceNow, Zendesk, Zoho, MS Dynamics, SAP as well as custom developed applications to integrate all customer engagement channels into the back-office solution.

    Symbee Connect Analytics unifies the Contact Center Channel data that comes from Amazon Connect and Symbee Connect into Historical and Real-time views in one Solution. It also allows WFO/WFM Providers to easily take data from Preview Dial, Email, WebChat, SMS and Social Messaging into their Solutions without heavy development and integration.
  - Netflix is the world's leading internet television network, with more than 150 million members in 190 countries. The Netflix Messaging team builds the systems and applications to communicate with members across various channels and relies on Amazon's SES for email delivery. Before it migrated to Amazon Simple Email Service (SES), Netflix had to maintain an in-house solution for sending emails. This in-house solution carried its own operational overheads, including running dedicated servers with email delivery software and optimizing email sending practices for each Internet Service Provider. It evaluated several email delivery solutions and decided on Amazon SES because it is flexible, affordable, highly scalable, has global reach, and promises excellent deliverability. Through a combination of the SES Dedicated IP offering and some strategies implemented on its end, after migrating to Amazon SES, it was able to streamline operations, improve Inbox Placement and increase our email sender reputation score.

- Founded in June of 2012, Coinbase has built the world's leading compliant cryptocurrency platform serving over 30 million accounts in over 100 countries. With the success of its flagship product, Coinbase Consumer, and its vocal advocacy for blockchain technology, it has played a major part in mainstream awareness and adoption of cryptocurrency. At Coinbase, security, scale, and flexibility are paramount to its customer engagement. When there are changes to currency trends, its customers need to know immediately and in the channel they prefer.

  Using Amazon Pinpoint and Amazon Simple Email Service (SES), it is able to react quickly and securely. By using Amazon Pinpoint, it is able to send push notifications whenever there is a movement in price, or a price hits a certain threshold. It sends millions of push notifications a week, up to 8M in 60 minutes. With Amazon SES, it can send millions of emails a day to keep its customers informed and happy.

- FOX SPORTS is Australia's leading producer of sports coverage and is home to Australia's favourite subscription television sports channels as well as Australia's number one multi-sports website and app. To deliver live sport of very high volume to its subscribers requires scale. It has implemented the AWS Cloud to leverage the end-to-end live event and content monetisation workflows only available in the cloud through AWS Elemental Media Services.

  Founded in 2007, Sonico.com is a social networking site with more than 48 million registered users, 85 per cent of them located in Latin America. Sonico employs 75 people and is headquartered in Buenos Aires, Argentina, with offices in Miami and Brazil. Sonico.com offers three separate "spaces" (Private, Public, and Professional) to help users organise their lives online. Uploading, sharing, and commenting on photos are among the most popular user activities.

  Before signing on with Amazon Web Services (AWS), Sonico.com stored user photos in more than 25 servers under a managed hosting agreement. Because it has small technical and operations teams, it wanted to explore the possibility of offloading this content from its back end. Reducing costs and improving data backup and scalability were a priority. The company looked into several options, including network-attached storage, storage area networks, and cloud computing alternatives.

  The combination of Amazon Simple Storage Service (Amazon S3) and Amazon Elastic Compute Cloud (Amazon EC2) was the best match for overall price and performance. Initially, Sonico.com faced the challenge of moving a large number of files (more than 1 billion images) over to Amazon S3. Sonico's engineers moved their data to AWS over a period of four months.

  Now, 100 percent of their image upload, processing, and storage is done with AWS. They use multiple Amazon EC2 instances running Linux and Apache to receive and process images, and Amazon S3 to store them.

  This migration resulted in a 70 percent savings compared to the cost of its previous architecture, while improving storage redundancy and scalability. It does not have to worry about provisioning and financing new storage servers anymore. AWS solved one of its key infrastructure pains with its scalable, cost-efficient solution.

4.13 Google App Engine

Definition : *Google App Engine is a cloud service that allows you to run your application code in the cloud without you having to manage the underlying infrastructure.*

- As a developer, you just bring your application code and upload it to the Google Cloud App Engine. The App Engine sets up the execution environment for your application code (virtual machines, run time environment, etc.) and then runs your provided application code.
- You just need to focus on your application and not managing the underlying execution environment.

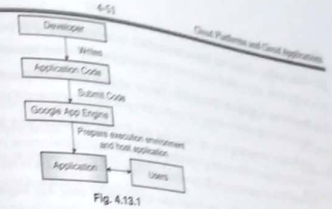


Fig. 4.13.1

- You can use various other Google Cloud services along with the Google App Engine for your application. For example, Fig. 4.13.2 is a sample application architecture from Google Cloud.

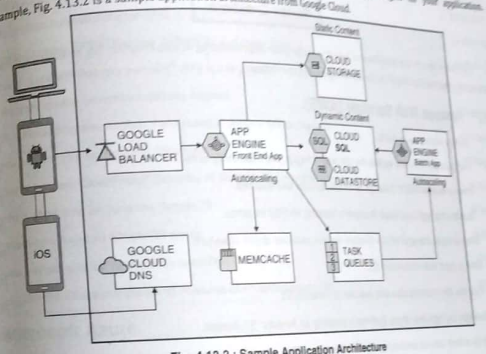


Fig. 4.13.2 : Sample Application Architecture

4.13.1 Characteristics and Features of Google App Engine

1. **Fully managed serverless application platform :** As a developer, you just need to bring your application code. Google App Engine manages the execution environment without you having to manage the underlying infrastructure. You don't need to,
   (a) Setup virtual machines
   (b) Install run time environments for your application
   (c) Configure infrastructure

   Google App Engine takes care of the infrastructure and application run time environment.
2. **Support for popular languages :** Google App Engine supports the popular application development languages such as Node.js, PHP, Python, Go, .NET, Java, Ruby, etc.

   It also provides options for bringing your own language runtimes and frameworks if you choose to do so.

3. **Auto Scaling :** Google App Engine automatically scales depending upon your application traffic. You don't need to worry about under provisioning or over provisioning your application resources.
4. **Monitoring, Logging and Diagnostics :** You can integrate with services such as Google Stackdriver to get powerful application diagnostics to debug and monitor the health and performance of your application.
5. **Application Versioning :** You can create and host different versions of your application and do user testing. You can also do intelligent traffic routing depending up on which application version you want users to use. For example, you can send 80% of application traffic to version 1, 15% to version 2 and 5% to version 3.
6. **Application Security :** You can protect your application by using App Engine Firewall. You can also use the TLS certificate service to encrypt the application traffic.

Review Questions

Here are a few review questions to help you gauge your understanding of this chapter. Try to attempt these questions and ensure that you can recall the points mentioned in the chapter

[A] Amazon Web Services (AWS)

Q. 1	Write a short note on AWS.	[4 Marks]
Q. 2	Using a block diagram, explain the concept of region and availability zone.	[6 Marks]
Q. 3	What is Amazon EC2? Describe its characteristics and features.	[6 Marks]
Q. 4	Describe the high-level steps involved in creating an EC2 Instance.	[8 Marks]
Q. 5	What is object storage? Which Amazon service provides object storage?	[4 Marks]
Q. 6	Write a short note on Amazon S3.	[4 Marks]
Q. 7	Describe the characteristics and features of Amazon S3.	[6 Marks]
Q. 8	Describe the high-level steps involved in creating an Amazon S3 Bucket.	[6 Marks]
Q. 9	Write a short note on versioning objects in Amazon S3.	[4 Marks]
Q. 10	Write a short note on server access logging in Amazon S3.	[4 Marks]
Q. 11	Write a short note on static website hosting in Amazon S3.	[4 Marks]
Q. 12	Write a short note on public bucket access in Amazon S3.	[4 Marks]
Q. 13	Write a short note on access control lists in Amazon S3.	[4 Marks]
Q. 14	Write a short note on bucket policy in Amazon S3.	[4 Marks]
Q. 15	With an example, explain bucket policy in Amazon S3.	[6 Marks]
Q. 16	With a flow chart, explain how access decision is made in Amazon S3.	[6 Marks]
Q. 17	What is CORS? With an example, explain CORS configuration in Amazon S3.	[8 Marks]
Q. 18	Write a short note on lifecycle management of objects in Amazon S3.	[4 Marks]
Q. 19	Write a short note on replication of objects in Amazon S3.	[4 Marks]

Q. 20	Write a short note on analytics of objects in Amazon S3.	[4 Marks]
Q. 21	Write a short note on metrics in Amazon S3.	[4 Marks]
Q. 22	Write a short note on inventory management in Amazon S3.	[4 Marks]
Q. 23	Write a short note on Amazon Glacier.	[4 Marks]
Q. 24	Compare Amazon Glacier and Amazon S3.	[4 Marks]
Q. 25	What is Amazon EBS? Describe its characteristics and features.	[4 Marks]
Q. 26	Compare the various volume types available in Amazon EBS.	[5 Marks]
Q. 27	Describe the high-level steps involved in creating and attaching an EBS Volume to an EC2 instance.	[8 Marks]
Q. 28	What is Amazon EBS Snapshots? Describe its features.	[8 Marks]
Q. 29	Write a short note on Amazon EFS.	[5 Marks]
Q. 30	What is Amazon EFS? Describe its features and characteristics.	[4 Marks]
Q. 31	What is a content delivery network? Why is it required?	[5 Marks]
Q. 32	Write a short note on content delivery network.	[4 Marks]
Q. 33	With a block diagram, explain the concept of content delivery network.	[4 Marks]
Q. 34	Write a short note on Amazon CloudFront.	[5 Marks]
Q. 35	What is Amazon CloudFront? Describe its features and characteristics.	[4 Marks]
Q. 36	Write a short note on Amazon SimpleDB.	[5 Marks]
Q. 37	Explain how Amazon SimpleDB works.	[4 Marks]
Q. 38	Write a short note on Amazon DynamoDB.	[5 Marks]
Q. 39	Describe the Characteristics and Features of Amazon DynamoDB.	[4 Marks]

[B] Microsoft Azure

Q. 40	Briefly explain any three Azure Services.	[5 Marks]
Q. 41	Write a short note on Azure Monitor.	[4 Marks]

[C] Overview of OpenStack Architecture

Q. 42	Write a short note on OpenStack.	[4 Marks]
Q. 43	Describe the various features of OpenStack.	[5 Marks]
Q. 44	List a few commonly used OpenStack services.	[4 Marks]
Q. 45	Describe the components of OpenStack and its architecture.	[5 Marks]
Q. 46	Describe OpenStack modes of operation.	[5 Marks]
Q. 47	Compare OpenStack modes of Operation.	[5 Marks]
Q. 48	You are planning to deploy OpenStack for your production environment consisting of 100 virtual machines. would be approximately 1,000 active users every hour. Which OpenStack mode of operation would you choo why?	[8 Marks]

5 Security in Cloud Computing

Syllabus

At the end of this unit, you should be able to understand and comprehend the following syllabus topics :

- Risks in Cloud Computing
  - Risk Management
  - Enterprise-Wide Risk Management
  - Types of Risks in Cloud Computing
- Data Security in Cloud
  - Security Issues
  - Challenges
  - Advantages
  - Disadvantages
  - Cloud Digital persona and Data security
  - Content Level Security
- Cloud Security Services
  - Confidentiality, Integrity and Availability
  - Security Authorization Challenges in the Cloud
  - Secure Cloud Software Requirements
  - Secure Cloud Software Testing

5.1 Risks in Cloud Computing

You have already covered this topic and its subtopics in unit 1 under Section 1.4, "Risks and Challenges in Cloud Computing". You are good!

5.2 Basic Terms and Concepts

Before you understand cloud security, let's build some general concepts around security.

General Security Concepts

Let me ask you a question	:	*"How do you manage your Debit Card and its PIN? Do you leave your Debit Card unattended and with PIN information available to everyone?"*
Your Response	:	*(Laughingly)"Of course, not. I keep my Debit Card with me all the time and never share my PIN with anyone".*
My Response	:	*"Oh, that's nice. But, why do you need to do that?"*
Your Response	:	*"Because, I need to ensure that my money is safe, and no one takes it out except me. I don't trust everyone with my money these days, you know".*
My Response	:	*"Got it. You are a security champion".*

If you followed our conversation, you already know what security is. Our job is easy now. Let us define some terms around our conversation above.

5.2.1 Assets

You were trying to protect your money, isn't it? It is called Assets. Money is your Asset in our conversation that you were trying to protect.

Definition : *Assets are something that has value and is worth protecting.*

Security is all about ensuring that the assets are kept protected all the time as much as possible within your capabilities or means.

5.2.2 Security Controls (or Countermeasures or Security Mechanisms)

So, how did you actually safeguard your money? You didn't leave the Debit Card around and you memorised your PIN, isn't it?

Definition : *Any countermeasures, mechanisms or actions that you take to safeguard an asset are called security controls.*

So, in our conversation, you have put two controls in place to safeguard your money – first is to keep your Debit Card with you and second is to memorise your PIN.

5.2.3 Threat

Hey, you told me that you don't trust everyone with your money, isn't it? That unknown everyone who can do evil to you or can harm you is called a Threat Agent. A threat agent executes a threat.

Definition : *A threat is a potential danger that can exploit an asset.*

You knew there are threats around your money, and you protected it so well.

5.2.4 Vulnerability

What if you left your Debit Card and PIN on the table for anyone to get hold of them and use? I hear you scream, "Come on, why would I do that to myself?". Exactly! You would not want to create a situation in which your assets can be harmed. This is precisely called addressing (or avoiding) a Vulnerability.

Definition : *Vulnerability is the weakness or lack of controls around assets.*

I am happy that you have put two good controls (keeping your Debit Card safe and memorizing your PIN) and you avoided the vulnerability around your money (asset).

5.2.5 Risk

So far, you would agree that leaving Debit Card and PIN unattended poses likelihood that someone might just grab them and use them.

Definition : *That likelihood of a harm occurring to an asset is called Risk.*

It is this Risk that you want to reduce by applying controls around your assets. Remember one thing here, Risk can NEVER be 0 (zero). Someone can steal your Debit Card from your wallet and force you at gunpoint to tell your PIN. The core thing that you need to ensure when dealing with Risk is "to reduce it to an acceptable level". Never aim to make anything (or any asset more precisely speaking) risk free because that's not possible, really.

5.2.6 Exposure

Someday suppose you do accidently leave your Debit Card behind and your PIN was known to someone, you could actually lose some or all your money. That particular day or rather that particular situation of you forgetting your Debit Card behind could lead to an exposure.

Definition : *Exposure is an instance of being harmed.*

So, if you got exposed anytime, immediately change your PIN and take a lesson in security to apply controls always around assets so that you do not have future exposures. I am sure you won't have exposures because you are a security champion already, aren't you?

Let me summarise the above terms in a simple block diagram shown in Fig. 5.2.1.

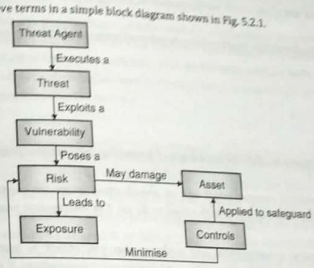


Fig. 5.2.1

Now that you have a general understanding of security, let's set some context about Information Security. When we say information security – what exactly are we protecting? What is the asset? The asset here is "Information" or more precisely "Digital Information". The information could be about your Facebook user account, Online bank account, OS password, email or pretty much anything that touches a computer system.

There are 3 tenets (or pillars) of security :

1. Confidentiality
2. Integrity
3. Availability

These tenets in short are also called as the **CIA triad** or any other combination of the first letters in their words. These are also sometimes called goals of security.

Let's dive deeper into each one of them.

5.2.7 Confidentiality

Definition : *Confidentiality can be defined as, an act of protecting information from unauthorized disclosure to an entity.*

- It ensures that the protected information is kept secret throughout its lifetime and is made available only to the authorized entities as and when needed.

- The information should be,
  - **Protected at Rest :** When stored on the disk.
  - **Protected in Motion :** When transmitted over the network.
  - **Protected during Use :** When processing.
- Remember our conversation from Debit Card and PIN? How did you protect your PIN and provide confidentiality to it?
  - **Protected at Rest :** You didn't write it down. You kept it in your mind. No one could know or use it except you.
  - **Protected in Motion :** You physically moved to an ATM (carrying your mind and the protected PIN there) instead of revealing it to anyone.
  - **Protected during use :** You watch out if someone is looking at your fingers as you punch the PIN on the ATM keyboard.
- In terms of digital information, confidentiality is enforced using several mechanisms :

1. Encryption
2. Access control
3. Data classification

5.2.8 Integrity

✍ ***Definition*** *:Integrity can be defined as, an act of protecting information from unauthorized modification by an entity.*

- It ensures that the information remains intact and no unauthorized entity can modify it. Any modification to the information is allowed only if the entity is authorized to do so. The information requires to maintain its integrity throughout its lifetime.
- For example, during criminal investigations, any evidence that you collect is protected from touching or any modifications to ensure that those evidences can be used during court proceedings. If evidence is tampered, it is not admissible in the court and cannot be used. Another example is email. If I send you an email and someone changes it before you read it, you might get wrong information, or it could be severely damaging to our relations.
- In terms of digital information, integrity is enforced using several mechanisms :

1. Hashing
2. Access control
3. Data classification
4. Input and output sanitization

5.2.9 Availability

✍ ***Definition*** *: Availability can be defined as, an act of protecting information from unauthorized destruction by an entity.*

- It ensures that the information is adequately protected to remain available when it is needed. Any unauthorized entity should not be able to destroy it. Also, the availability principle extends to any equipment such as computers, network devices and printers. These should be available and be able to perform as expected. If someone can get access to them and then prevent you from using these then that impacts availability of the system for your use.
- For example, your Windows or Linux systems track all activities done on the system via log files. If I do some mischiefs around your computer and then delete the log files, you would have no way to prove that I did something to your computer. The availability of log files is crucial to ensure that the system is adequately monitored and protected from any security mishaps.

- Availability is generally enforced using several mechanisms :

1. Access control
2. Isolation
3. Back up
4. Disaster recovery
5. Business continuity processes

5.2.10 Identification

✍ ***Definition*** *: Identification (in short ID) is defined as a way to claim an entity's presence with respect to the process being carried out.*

This means that during a process, your presence (or your consent) is ascertained (or established). For example, when you try to login to your Facebook account, you provide your Email or Phone number to establish your presence during the login process. There are several other forms of identification that we use today such as Aadhar Card, PAN Card, Voter ID, Debit Card, Admit card, etc. All of these identification methods bring a sense of credibility that you are present, or you give your consent (approval) to complete a particular process.

5.2.11 Authentication

✍ ***Definition*** *: Authentication is defined as; a way to ensure that the entity is indeed what it claims to be.*

This means that providing just the ID is not enough. You must additionally prove that the ID belongs to you. For example, even if I know your Facebook email address or phone, I cannot login as you until I also know the password. Thus, knowing just the ID is not enough. You need to prove that the ID belongs to you and that is what is precisely called authentication. It is for this reason that you need to additionally sign when you submit Aadhar card or PAN card as an ID proof to ensure that someone didn't just use the photocopy of those IDs without your permission (or consent). Some of the ways to authenticate an ID are passwords, biometric (like your Aadhar fingerprints or phone sensor), PIN (like for Debit Card), or OTP (SMS that you get to confirm a transaction).

5.2.12 Authorisation

✍ ***Definition*** *: Authorisation is defined as, a way to determine what resource an entity can access.*

Once you have provided your ID and have been successfully authenticated, the next step is authorisation where the system determines if you have the permission to access the desired object. For example, even if you have a valid voter ID card but if your name is not on the electoral list at a particular area booth, you won't be allowed to vote. Having authenticated ID is one thing and getting access to the resource is another. Just because you have an authenticated ID, does not mean that you have automatically access to the resources. So, authenticated ID is a must for authorisation but that does not always guarantee that you would be allowed access.

5.2.13 Accountability

✍ ***Definition*** *: Accountability is defined as, a way to record your actions.*

Suppose, you used a system to take print outs. That system logs this action (pretty much like you record attendance in lab or classroom) to build a trace (evidence or proof) that you used the printer. If you were not supposed to use the printer, the evidence can be used to find you accountable for using it without permissions and could result in particular consequences. Accountability is a key determinant of how securely a system is operating. The logs generated are continuously monitored and necessary alarms are raised if any entry is found to be suspicious.

Let's summarise the 4 access control steps with the help of a Fig. 5.2.2.

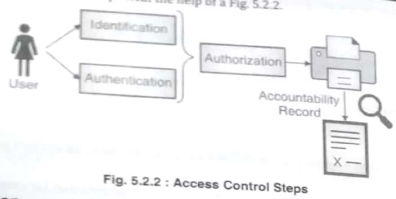


Fig. 5.2.2 : Access Control Steps

5.2.14 Non-repudiation

Definition *: Non-repudiation is defined as, a way to prove your actions.*

It is used in conjunction with authentication and accountability. Non-repudiation provides an assurance that someone cannot deny their actions later on. For example, if I sent you an email, I cannot later deny that I did not. To send an email, I must have used my email ID and password and then sent it over to you over a secure network where no one could change the email body. If you can establish all of these facts truthfully, you have proven that I sent that email and thus established non-repudiation.

5.2.15 Security Policies

University Question

Q. How are security policies and security mechanisms used to counter threats by threat agents?

SPPU - May 19, 8 Marks

Definition *: Information Security Policies or just Security Policies are aggregate of directives, regulations, rules, and practices that prescribe how an organisation manages, protects, and distributes information.*

- Policies, in general, define abroad set of rules that governs all aspects of how an organisation would operate. Like any political government, the organisation defines several policies covering the broad continuum of organisation management and operations.
- Note here that the term policy is used widely meaning various things depending on the context. For example, there could be authentication policy, authorisation policy, loan approval policy, etc. The policies that you are learning about in this section majorly refer to the Information Security Policy that is defined organisation wide to ensure protection for relevant business data and secure operations.
- Computer systems, data, applications and staff are crucial to the organisation's mission and objectives. Senior management needs to ensure that these all are adequately protected. The management needs to unify how these are handled, operated and protected and cannot just leave them at the personal understanding, experience and skills of various people or technology. The policies are defined at the top executive level and are pushed down to the various departments and individuals for adhering to those policies.

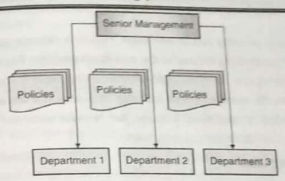


Fig. 5.2.3 : Security Policies

5.3 Principles for Cloud Security (Secure Cloud Software Requirements and Testing)

There are several security principles for protecting your cloud resources. Fig. 5.3.1 shows some of the major security principles for cloud.

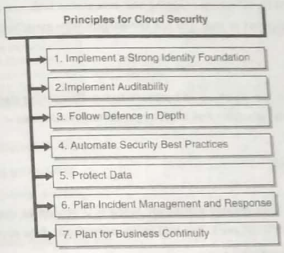


Fig. 5.3.1

1. **Implement a Strong Identity Foundation**

- Identity and Access Management (IAM) is perhaps the most crucial part of securing your cloud account and your cloud resources. It is like securing access to your email or social media account. If someone is to get access to your email user name and password, she can then pretty much control anything – read emails, send emails, delete emails or use your email for fraudulent activities.
- Similarly, if someone can get access to your cloud account or cloud resources, then she can pretty much control the resources and use them as she wishes. You may or may not be able to detect such a compromise. Hence, it is extremely important to build a strong identity and access management foundation. Use strong passwords, preferably multi-factor authentication, authenticated actions, strong access control for cloud resources and secure communication everywhere.

2. **Implement Auditability**

You should have a solution for auditing and monitoring your cloud account and cloud resources. Your cloud environment could be large, and it may not be humanly possible to trace everything yourself. Set effective monitoring and alerting solutions so that you are notified if something requires your immediate attention. For example, if an administrator account has more than three failed login attempts, you could send an alert to the cloud account owner. Similarly, if any of your cloud account resources are being compromised, intrusion detection tools can appropriately alert you.

3. Follow Defence in Depth

As you understand, do not build security just at one layer. Follow a defence in depth approach where each type of cloud resource has its own security capabilities utilised to the maximum possible. For example, storage service may provide its own access control apart from the regular cloud account controls. Using explicit access control for storage service would ensure that the objects stored within the storage service can only be accessed if the access control policy explicitly allows so. You may have several users created in your cloud account. Just because someone has a cloud account, does not mean that she automatically has access to everything your account owns.

4. Automate Security Best Practices

Wherever possible, automate security best practices. You then do not require to constantly have a human around for monitoring security. For example, create secure OS baselines that everyone can use in your cloud account. Automate the job of reporting login information on daily basis. Send notification if the account usage is going higher than what you have set a threshold for. Alert if your resources are open to the public. Automated controls such as these can handle the cloud scale and keep you aware of things that matter the most. Many a times, automation can take the appropriate actions as well. For example, if someone switches off the cloud account logging, the automated task can automatically switch on the logging as well as notify you of such a security event so that you can investigate who switched off the logging and why and take the required corrective actions.

5. Protect Data

- It is needless to say that data is the lifeblood of business today. Protecting data is crucial to running your business smoothly. Data could be private or sensitive and must be protected at all times –
  - Protection for Data at Rest
  - Protection for Data in Transit
  - Protection for Data in Use
- Your data in the cloud requires even greater protection since it is stored in the shared environment and is usually accessed and processed over a public network. You will learn more about data protection in the later section.

6. Plan Incident Management and Response

You could have data breaches or security events in the cloud as well. You should plan for incident management and response. What if your account is compromised? How will you handle a DDoS attack? How will you handle a critical security patch? These are some of the questions that require definitive answers and you must plan for such events.

7. Plan for Business Continuity

Even though cloud resources are highly resilient and build to guarantee a certain level of service uptime, your application and data access might face disruption due to unforeseen issues. You should plan for business continuity in case of any disaster or any other service downtime. You may sometimes lose network connectivity such that to have no access to your resources.

5.4 Security and Governance Services

✍ ***Definition :*** *Security and Governance Services provide various types of identity management, access control, key management, encryption and other security and compliance related services.*

- Cloud service providers offer several services for securing your cloud resources. Some of the security services are as following :
  - Identity and access management

  - Encryption and key management
  - Digital certificate management
  - Web application firewall
  - Intrusion detection and prevention systems
  - Configuration management
  - Regulatory Compliance and Audit management
- You get enough flexibility to choose the security services that match your protection requirements. Cloud service providers ensure that their services are periodically audited and match a certain level of security assurance.
- You are billed as per your consumption or billing plan.

5.5 Cloud Identity and Access Management (IAM)

- Identity and Access Management (IAM) is perhaps the most crucial element of cloud security. IAM ensures that only authorised entities have access to the cloud resources and everyone else is denied access.
- IAM typically follows the PARC model.

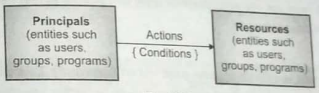


Fig. 5.5.1

- Access control policies typically follow a PARC model.
  - P = Principals (users, groups, programs)
  - A = Actions (Create, Read, Update, Delete)
  - R = Resources (OS, Network, Files, etc.)
  - C = Conditions (time of the day, type of OS, etc.)
- In cloud environment as well, you need to ensure that the principals can only take the desired actions on authorised resources under specific conditions. You need to define IAM policies for all your cloud resources and the access is granted only if the principal satisfies the IAM policy for the given resource.

5.5.1 IAM Challenges in the Cloud (Security Authorization Challenges in the Cloud)

Fig. 5.5.2 some of the challenges of establishing a robust IAM in the cloud.

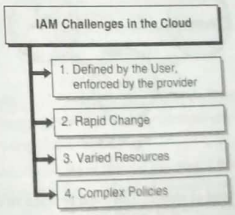


Fig. 5.5.2

1. Defined by the User, Enforced by the Provider

Unlike the traditional datacentre, the IAM policies are defined by the user but enforced by the cloud service provider. It is a joint responsibility that the defined IAM policies are honoured. Users who are new to the cloud may not be fully aware of how the policies need to be defined and how are they evaluated. Understanding the IAM policies, as applicable to the particular cloud service provider's environment, is necessary to define the correct policies.

2. Rapid Change

The cloud resources can be brought up within seconds and brought down quickly as well to manage the temporary peaks and downs in the demand. With the rapid change of resources, it could be hard to set an access policy quickly. You may require automation to set the policies as soon as the resources are brought up.

3. Varied Resources

Not all cloud resources have the common mechanism to define policies. Some resources use identity based policies, whereas some use resource based policies. You must know which policy to apply to which resource and how. Additionally, you must also manage IAM for the resources you own internally in your organisation. The users for both internal and external resources could be the same and you must properly apply IAM policies so that the users get the minimum level of access required to complete their respective jobs.

4. Complex Policies

Cloud based IAM policies could be difficult to understand and write correctly. Each cloud provider has its own way of defining IAM policies. If you are using multiple cloud providers, it could be further complex to understand the policies and how they interplay.

5.5.2 Identity Management Lifecycle

Typically, identities have the following lifecycle.

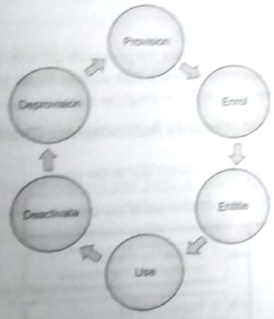


Fig. 5.5.3

1. Provision

Provisioning is the mechanism of creating an identity. This could be when an employee joins the organisation or joins a new project or role within the same organisation. Typically, the administrators have the privilege of creating an identity as desired, following the process established within the organisation.

2. Enrol

Once the identity is created, it is enrolled in the system where it is designated for use. For example, you can enrol an employee's identity for using with the cloud service provider. The enrolment process could be manual or automated.

3. Entitle

After the identity is enrolled on the system, it is assigned roles and permissions within the system. The entitlement is based on the job to be performed by that identity. You should be careful in this step to ensure that the identity is not over-permissive and is only allowed the access that is actually required for the job to be done.

4. Use

During the useful lifetime of the identity, it is used as per the entitlements assigned to it. Sometimes, the entitlement might require update if the nature of job changes during the lifetime of the identity.

5. Deactivate

Once an identity has reached the end of its useful lifetime, it is first deactivated on the system. It is not immediately deleted because the identity may hold the cryptographic keys or other important information associated with it that might be required in the near future. In the inactive state, the identity cannot perform any new functions but can retain the previous information owned by it.

6. Deprovision

Finally, the identity is deleted once it is confirmed that it is no more required. This step might also require deleting the information that was associated with the identity during its lifetime or the information be transferred to some other identity for future retention.

5.5.3 Types of Identity Providers Used in the Cloud (Identity and Presence)

Definition : *Identity Provider is the party that manages the identities and services around it.*

You can decide to choose the identity provider that manages the identity and authentication information in the cloud. Some identity providers can also provide authorisation and other identity related services. Following are some of the types of identity providers used in the cloud.

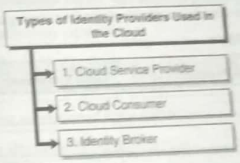


Fig. 5.5.4

1. Cloud Service Provider

- This is the simplest model of identity management. In this model, the identities are managed directly with the cloud service provider. You follow the entire lifecycle of identity management with the cloud service provider. It is like creating an account on Google for yourself and then using it.
- The advantage of using cloud service provider to manage identities is that all the identities that must require cloud access are directly created and managed in the cloud portal (hosted by the cloud service provider). You don't need to configure and manage anything outside it. An organisation might have several employees. The identities for only the ones that require cloud access can be created explicitly with the cloud service provider.

1. Defined by the User, Enforced by the Provider

Unlike the traditional datacentre, the IAM policies are defined by the user but enforced by the cloud service provider. It is a joint responsibility that the defined IAM policies are honoured. Users who are new to the cloud may not be fully aware of how the policies need to be defined and how are they evaluated. Understanding the IAM policies, as applicable to the particular cloud service provider's environment, is necessary to define the correct policies.

2. Rapid Change

The cloud resources can be brought up within seconds and brought down quickly as well to manage the temporary peaks and downs in the demand. With the rapid change of resources, it could be hard to set an access policy quickly. You may require automation to set the policies as soon as the resources are brought up.

3. Varied Resources

Not all cloud resources have the common mechanism to define policies. Some resources use identity based policies whereas some use resource based policies. You must know which policy to apply to which resource and how. Additionally, you must also manage IAM for the resources you own internally in your organisation. The users for both internal and external resources could be the same and you must properly apply IAM policies so that the users get the minimum level of access required to complete their respective jobs.

4. Complex Policies

Cloud based IAM policies could be difficult to understand and write correctly. Each cloud provider has its own way of defining IAM policies. If you are using multiple cloud providers, it could be further complex to understand the policies and how they interplay.

5.5.2 Identity Management Lifecycle

Typically, identities have the following lifecycle.

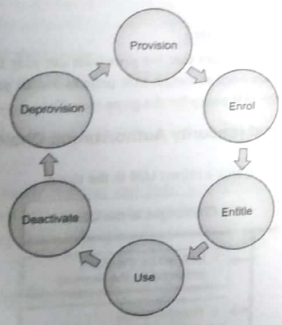


Fig. 5.5.3

1. Provision

Provisioning is the mechanism of creating an identity. This could be when an employee joins the organisation or joins a new project or role within the same organisation. Typically, the administrators have the privilege of creating an identity as desired, following the process established within the organisation.

2. Enrol

Once the identity is created, it is enrolled in the system where it is designated for use. For example, you can enrol an employee's identity for using with the cloud service provider. The enrolment process could be manual or automated.

3. Entitle

After the identity is enrolled on the system, it is assigned roles and permissions within the system. The entitlement is based on the job to be performed by that identity. You should be careful in this step to ensure that the identity is not over-permissive and is only allowed the access that is actually required for the job to be done.

4. Use

During the useful lifetime of the identity, it is used as per the entitlements assigned to it. Sometimes, the entitlement might require update if the nature of job changes during the lifetime of the identity.

5. Deactivate

Once an identity has reached the end of its useful lifetime, it is first deactivated on the system. It is not immediately deleted because the identity may hold the cryptographic keys or other important information associated with it that might be required in the near future. In the inactive state, the identity cannot perform any new functions but can retain the previous information owned by it.

6. Deprovision

Finally, the identity is deleted once it is confirmed that it is no more required. This step might also require deleting the information that was associated with the identity during its lifetime or the information be transferred to some other identity for future retention.

5.5.3 Types of Identity Providers Used in the Cloud (Identity and Presence)

Definition : *Identity Provider is the party that manages the identities and services around it.*

You can decide to choose the identity provider that manages the identity and authentication information in the cloud. Some identity providers can also provide authorisation and other identity related services. Following are some of the types of identity providers used in the cloud.

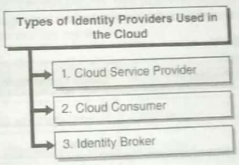


Fig. 5.5.4

1. Cloud Service Provider

- This is the simplest model of identity management. In this model, the identities are managed directly with the cloud service provider. You follow the entire lifecycle of identity management with the cloud service provider. It is like creating an account on Google for yourself and then using it.
- The advantage of using cloud service provider to manage identities is that all the identities that must require cloud access are directly created and managed in the cloud portal (hosted by the cloud service provider). You don't need to configure and manage anything outside it. An organisation might have several employees. The identities for only the ones that require cloud access can be created explicitly with the cloud service provider.

- The disadvantage of using cloud service provider to manage the identities is that the organisation might lose control of identities associated with users as they join and leave the organisation or change the role within the organisation. The organisation needs to duplicate the identity management – both at its own premises and also with the cloud service provider. This could be inefficient and error prone. Additionally, the users might have to remember two different account details – one for the organisation and one for the cloud service provider. If the organisation uses several cloud service providers, this could be further challenging and problematic.

2. Cloud Consumer

- As a cloud consumer, you can provide identity information from within the organisation. The organisation already has the information about its users in solutions such as Active Directory or LDAP (Lightweight Directory Access Protocol). This is usually done using identity federation. You will learn about federation later. For now, just understand that there exists a way using which the identities defined within the organisation system can be used outside of it with the cloud service provider.
- The advantage of linking the organisation identity systems with the cloud service provider is that you don't have to duplicate the effort in maintaining the identities. Also, as users join or leave the organisation, the identities are automatically created or deleted as part of the user on boarding or exit process.
- One challenge of linking the organisation identity system with the cloud service provider could be complexity. But, the advantages are by far more than the complexity involved and hence several organisations prefer this approach.

3. Identity Broker

- The final model of identity management could be using a third-party which is also known as identity broker. Identity brokers manage the entire lifecycle of the identity for both within the organisation use and also the cloud service provider. Identity brokers provide several features apart from the regular identity management, authentication and authorisation.
- They integrate with several systems making your life easy by not having to duplicate the identity information on every system. They could also provide additional features such as identity audit and multi-factor authentication.
- One disadvantage of this approach could be that it could be a single point of failure. Since all your identities are managed by one identity broker and if that identity broker has any disruption, it could mean disruption for your users as well. You won't be able to use any system until the identity broker is up again.

Comparison between Identity Providers

Comparison Attribute	Cloud Provider	Cloud Consumer	Identity Broker
Duplicate identity information	Yes	No	No
Complexity	Low	Medium	Medium
Effort of managing identities	High	Low	Low
Cost	Negligible	Negligible	Comes at a cost
Single point of failure	Unlikely	Unlikely	Likely

5.6.4 Best Practices for IAM in the Cloud

- Here are some of the recommended best practices for identity and access management in the cloud.

1. Secure your cloud accounts using multi-factor authentication.
2. Do not share your account details with anyone.

3. Create individual accounts for users and do not create shared accounts.
4. Grant least privileges based on the job required.
5. Use roles wherever possible instead of assigning permissions directly to the user.
6. Rotate credentials periodically.
7. Monitor your account logins and cloud usage patterns.
8. Protect access keys used for programmatic accesses.
9. Check out the vendor documentation and hardening guide respective to the cloud service provider that you use.

- You can use vendor documentation or the cloud service provider specific hardening guides for understanding and following the best practices for your cloud IAM. For example, following are the recommendations from the CIS Security Benchmark for AWS.

S. No.	Recommendation for Identity and Access Management
1.1	Avoid the use of the "root" account
1.2	Ensure multi-factor authentication (MFA) is enabled for all IAM users that have a console password
1.3	Ensure credentials unused for 90 days or greater are disabled
1.4	Ensure access keys are rotated every 90 days or less
1.5	Ensure IAM password policy requires at least one uppercase letter
1.6	Ensure IAM password policy require at least one lowercase letter
1.7	Ensure IAM password policy require at least one symbol
1.8	Ensure IAM password policy require at least one number
1.9	Ensure IAM password policy requires minimum length of 14 or greater
1.10	Ensure IAM password policy prevents password reuse
1.11	Ensure IAM password policy expires passwords within 90 days or less
1.12	Ensure no root account access key exists
1.13	Ensure MFA is enabled for the "root" account
1.14	Ensure hardware MFA is enabled for the "root" account
1.15	Ensure security questions are registered in the AWS account
1.16	Ensure IAM policies are attached only to groups or roles
1.17	Maintain current contact details
1.18	Ensure security contact information is registered
1.19	Ensure IAM instance roles are used for AWS resource access from instances
1.20	Ensure a support role has been created to manage incidents with AWS Support
1.21	Do not setup access keys during initial user setup for all IAM users that have a console password
1.22	Ensure IAM policies that allow full "*:*" administrative privileges are not created

5.7 Data Protection in Cloud

- Mostly you will agree with me if I say that the data is lifeblood of any business.
- For example, imagine
  - Can YouTube survive the loss of all its videos forever?
  - What if a bank exposed all the account records publicly?
  - Will you be comfortable if your Aadhar card, phone number, name, address and other personal details are publicly available?
- The answers to such questions might be too hard but one thing might be clear that the data is extremely crucial to remain protected at all times.
- Increasingly businesses are adopting cloud services and moving their applications and data in the cloud. The data ownership is with the business, but it resides with the cloud service provider. The responsibilities for data protection need to be carefully understood and agreed upon.
- Following are some of the major data exposures on AWS storage service named S3.

Timeline	Data Exposure	Company
May-2017	Battlefield imagery and administrator credentials to sensitive systems	Booz Allen Hamilton
Jun-2017	Personal data about 198 million American voters	U.S. Voter Records
Jul-2017	Personally identifiable information for 2.2 million people	Dow Jones & Co
Jul-2017	Personally identifiable information about over 3 million wrestling fans	WWE
Jul-2017	Personally identifiable information about 6 million people and sensitive corporate information about IT systems, including login credentials	Verizon Wireless
Sep-2017	Personally identifiable information about 4 million customers, proprietary code, and administrator credentials	Time Warner Cable
Nov-2017	Terabytes of information from spying archive, resume for intelligence positions--including security clearance and operations history, credentials and metadata from an intra-agency intelligence sharing platform.	The U.S. Department of Defense
Oct-2017	Master access keys for Accenture's account with AWS Key Management system, plaintext customer password databases, and proprietary API data	Accenture
Dec-2017	111GB of detailed financial information--including full credit reports--about 47,000 people	National Credit Federation
Dec-2017	Personal information about 123 million American households	Alteryx

ry, isn't it? Let's learn about how data might be protected in the cloud.

5.7.1 Data Security Concerns in the Cloud

Fig. 5.7.1 shows some of the major data security concerns in the cloud.

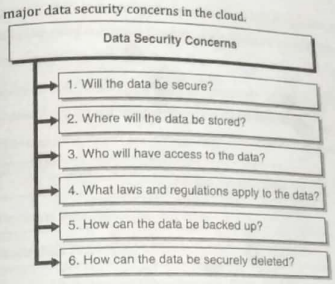


Fig. 5.7.1

1. **Will the data be secure?**

The first and foremost concern when moving to the cloud is that will your data be secure. Will it get the same level of protection that it used to get when it was in your own datacentre? Specifically speaking, you need to ensure the following.

(i) **Confidentiality :** Will your data be transmitted and stored in a way that it is not exposed to unauthorised entities? Can you apply the regular encryption mechanisms to safeguard your data as it is transmitted or stored? In the cloud, you need to ensure that the data is encrypted as well.

(ii) **Integrity :** You also need to ensure that the data integrity is maintained. It must be hard to alter the file contents or modify database entries. It should be protected from the public as well as the insiders from your organisation as well as the cloud service provider organisation.

(iii) **Availability :** The data should be available for business continuity. The cloud service provider should guarantee a minimum level of assurance of data availability. Usually availability is given in terms of number of nines (9s). For example, a cloud service providers can claim the data availability to be 99.999% which means a downtime of just 5.26 minutes in a year!

(iv) **Privacy :** Your business might be processing user's personal data. When you put their personal data on the cloud, you need to ensure that it is not exposed. Also, the cloud service provider or other tenants should not have any visibility into that data, and neither should be able to sell it to a third party for any other motive.

2. **Where will the data be stored?**

- Unlike your own datacentre where the location of operation is fixed, you can consume cloud services globally. Cloud service providers typically have various regions in which they provide services tuned for local market in terms of pricing and features. For example, look at the following map of Google Cloud Platform. It provides services distributed across various regions around the globe. This map is as of May 2019.

- Given that you can choose the location of operation yourself, it is your responsibility to choose where the data is kept. The cloud service provider would not move your data or place your data at their own will.
- Data placement in a particular region is important for various reasons.
- Some of them could be

(i) Data Sovereignty : The data might be subject to government laws and regulations. For example, European Union has put certain restrictions on the countries where the personal data of EU citizens can be stored.

(ii) Data Security : The cloud service provider might not have the same security services available in all the regions equally. If you have a dependency on using a particular security service, you will have to choose the region where that particular service is available.

(iii) Latency : Your application might be processing huge amount of user data. If the data is kept farther from the user location, it might increase the response time of your application.

- So, understanding where your data is stored in the cloud is crucial to ensuring its security.

3. Who will have access to the data?

- Having the right visibility for who all can access your data in the cloud is very critical. Remember our discussion from previous section where private and sensitive data was exposed publicly on AWS S3? Ensuring that the data has appropriate access control mechanisms established can go a long way in ensuring that it is not exposed to unauthorised entities.
- When considering access, review the following entities:
  - (i) Who, from your organisation, has access? What can they do with the access [read, write, update, delete]?
  - (ii) Who, from cloud service provider, has access? What can they do with it?

  - (iii) Will government have access to the data, if needed?
  - (iv) Is your data publicly accessible? Does it need to be public? What level of public access granted?
  - (v) Access control for your data in the cloud can really be worrisome if you do not plan it well.

4. What laws and regulations apply to the data?

- I talked about data sovereignty in brief earlier. Various businesses are under the control of laws and regulations that determine how the business should carry out its operations and keep its data.
- Some of the data regulations around the world are as following.

Region	Law / Protection
Argentina	Personal Data Protection Act 2000
Australia	Australia's Privacy Act 1988
Canada	Personal Information Protection and Electronic Documents Act (PIPEDA) 2000
France	France's Data Protection Act 2
European Union	General Data Protection Regulation (GDPR)
India	Personal Data Protection Bill, 2018

- Apart from the laws, there could also be compliance regulations such as PCIDSS and HIPPA that may require you to protect the information and could lead to heavy fines and penalties if data is found to be breached.
- When moving your data to the cloud, you should be aware of which laws and regulations you need to be compliant with and how that compliance level can be achieved after migration. Careful planning and risk assessment would ensure that you can meet those data protection requirements.

5. How can the data be backed up?

- Data backup is essential to ensure business continuity. The regular data backup process that you might have established for your datacentre may not be adequate in the cloud environment. The data that requires backup could be from virtual machines, cloud storage, databases or other forms of cloud applications.
- You must identify the various types of data that you have in the cloud and how each one can be adequately backed up.
- Cloud service providers typically provide several options for automating and centralising your data backup. You should additionally take precautions that the backup has same level of protection applied. You would not want to expose backed up data and must ensure that the data is secured all times whether in use or in backup.

6. How data can be securely deleted?

- You might want to delete data when it reaches the end of its useful lifecycle. But, as you might already know, when you delete a file, its content still resides on the disk until overwritten by the content of another file.
- As per NIST 800-88, Guidelines for Media Sanitization, there are several mechanisms for completing destroying the information stored on a media.

Method	Description
Clear	One method to sanitize media is to use software or hardware products to overwrite storage space on the media with non-sensitive data. This process may include overwriting not only the logical storage location of a file(s) (e.g., file allocation table) but also may include all addressable locations. The security goal of the overwriting process is to replace written data with random data. Overwriting cannot be used for media that are damaged or not rewriteable.
Purge	Degaussing and executing the firmware Secure Erase command are acceptable methods for purging. Degaussing is exposing the magnetic media to a strong magnetic field in order to disrupt the recorded magnetic domains. A degausser is a device that generates a magnetic field used to sanitize magnetic media. Degaussing can be an effective method for purging damaged or inoperative media, for purging media with exceptionally large storage capacities, or for quickly purging diskettes.
Destroy	There are many different types, techniques, and procedures for media destruction. If destruction is decided on because of the high security categorization of the information, then after the destruction, the media should be able to withstand a laboratory attack. *Disintegration, Pulverization, Melting, and Incineration.* These sanitization methods are designed to completely destroy the media. They are typically carried out at an outsourced metal destruction or licensed incineration facility with the specific capabilities to perform these activities effectively, securely, and safely. *Shredding.* Paper shredders can be used to destroy flexible media such as diskettes once the media are physically removed from their outer containers. The shred size of the refuse should be small enough that there is reasonable assurance in proportion to the data confidentiality that the data cannot be reconstructed.

- In the cloud, your data resides on shared storage and it may not be possible to physically destroy the media. However, the cloud service providers have special secure wipe routines that can be used to permanently erase data. These techniques are compatible with standard media sanitisation guidelines such as NIST 800-88.
- You could also encrypt your entire data and then delete the keys. This is also a good way to ensure that your data is gone forever and cannot be restored and used without your knowledge.

5.7.2 Data Encryption in the Cloud (Cloud Digital Persona and Data security)

Data encryption is perhaps the most important control for protecting data. In the cloud environment as well, the cloud service providers have several options for encrypting data and managing the encryption key lifecycle. Let's understand the various aspects of data encryption in the cloud.

Note : I would be mostly focusing on data at rest encryption. Data in-transit encryption is typically provided by TLS connection that you are already familiar with.

5.7.2(A) Shared Responsibility for Encryption based on Type of Cloud Service

It is important to understand that cloud services are majorly of three types – IaaS, PaaS and SaaS. You have already learnt about them in the previous sections. Based on type of service in reference, the data encryption responsibility and key management varies. Let's understand them.

1. Encryption Responsibility for Infrastructure as a Service (IaaS) (Virtual Machine Security)

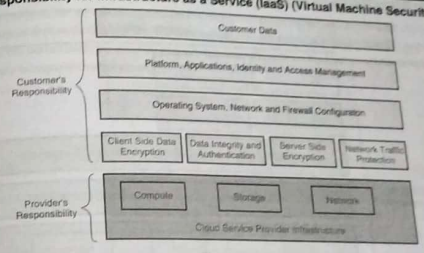


Fig. 5.7.2

For IaaS, the customer has the entire responsibility of configuring data encryption. The cloud service provider may provide features and mechanisms to help the customer configure encryption and manage the keys. But, it is completely on the customer to secure the data using those encryption mechanisms. Cloud service provider has more or less no control over the data encryption for IaaS. An example of IaaS service is virtual machine. The customer must secure the virtual machine disk by encrypting it.

2. Encryption Responsibility for Platform as a Service (PaaS) (Application Security)

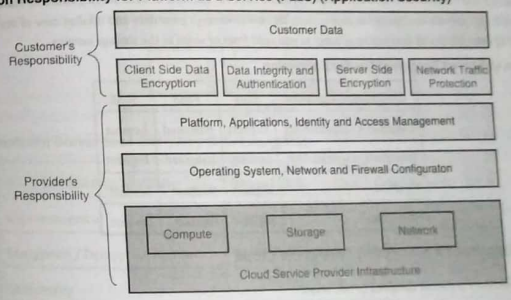


Fig. 5.7.3

For PaaS, the cloud service provider shares more responsibility than IaaS. It takes care of the operating system where the PaaS service is hosted as well as any PaaS service specific security configuration that you choose to configure. An example of PaaS service could be Database service. The cloud service provider can ensure that the disk, on which the database records are stored, is entirely encrypted. You, as a user, do not have control over the OS of the PaaS service. However, you can configure PaaS service specific security configuration. For example, you can provide the key that could be used for encrypting the database records.

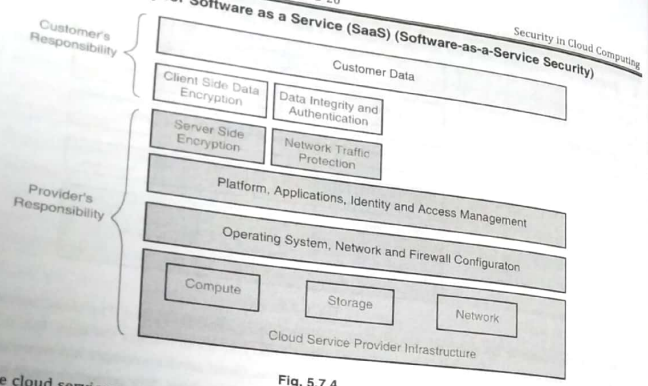


Fig. 5.7.4

For SaaS, the cloud service provider takes the maximum responsibility. It ensures that the service is entirely secured. The disk, the OS, the service configuration, network traffic, etc. are completely taken care by the service provider. You could just choose to use the controls provided and can encrypt the data. An example of SaaS service could be storage service. You could just use it and provide configuration instructions on how to encrypt your data and it takes care of managing the keys and automating encryption and decryption process as you read from or write to the storage service.

Comparison of Shared Responsibility for Encryption

Comparison Attribute	IaaS	PaaS	SaaS
Customer's responsibility	Highest	Balanced	Lowest
Provider's responsibility	Lowest	Balanced	Highest
Flexibility	Highest	Medium	Lowest
Complexity	Highest	Medium	Lowest

5.7.2(B) Mechanisms for Encrypting Data in the Cloud

You could encrypt the data in the cloud in two ways.

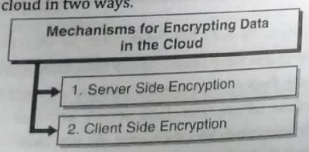


Fig. 5.7.5

1. Server Side Encryption

In this type of encryption, the data is encrypted by the server before being written to the disk. The client sends the data in unencrypted format (but securely over the TLS connection) and the server applies encryption to it. When the client wants to read the data, the server decrypts the data and sends it securely over to the client. The encryption and decryption processes are completely transparent to the client.

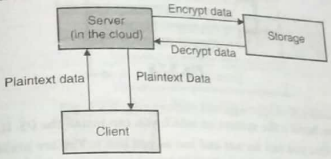


Fig. 5.7.6

2. Client Side Encryption

In this type of encryption, the client encrypts the data before sending it to the server. The server just writes the data as received from the client. When reading the data, the server passes the encrypted data back to the client. The client must decrypt it before it can be read in plaintext.

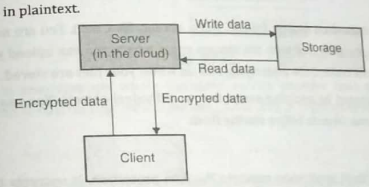


Fig. 5.7.7

Comparison between Server Side and Client Side Encryption

Comparison Attribute	Server Side Encryption	Client Side Encryption
Keys managed at	Server	At each client
Encryption / Decryption Process	Carried out by server	Carried out by client
Complexity	Low	High
Data needs to be protected in-transit	Yes	No

5.7.2(C) Types of Cloud Storage Requiring Encryption (Content Level Security)

In the cloud, majorly there are three kinds of data storage that require encryption.

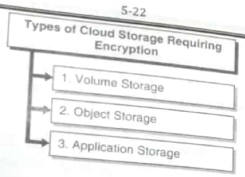


Fig. 5.7.8

1. **Volume Storage**

- Volume storage allows you to build a file system on which you can install the OS. It is a block level storage. Volume storage behaves likes a disk that you can format and use as you like it. You are presented with the raw disk and you can carry out disk operations as you do on your regular computer.
- Volume storage can be encrypted per virtual machine instance. You can use the full disk encryption mechanism from the cloud service provider. You can also use the regular OS-based encryption mechanisms such as Windows® BitLocker® for Windows® based OS or Linux dm-crypt.

2. **Object Storage**

- Object storage stores the individual objects (music, video, photos, files, etc.). You are not presented with a block level device (disk). Instead you are provided with the storage space where you can upload your files. For example, Google Drive or Dropbox. You do not control the underlying disk on which your files are stored.
- Object storage can be encrypted by applying encryption to individual objects. You can use server-side or client-side encryption for encrypting your objects before storing them.

3. **Application Storage**

In application storage, the cloud application manages the data protection. It encrypts the data before writing to the disk and decrypts it on user demand. The encryption and decryption processes are completely transparent to the user and the application handles the process in the background seamlessly. For example, if you use a database application as a service in the cloud, you can turn on the encryption for this service. The database service can then manage the entire encryption and decryption processes and also the keys. You can optionally manage the keys as well.

5.7.2(D) Encryption Management in the Cloud

- Encryption depends on three things:

1. Data to encrypt
2. Encryption Keys
3. Encryption process or algorithm

- By now, you understand that you, as the cloud consumer, are always responsible for protecting your data. The remaining two parameters, keys and the process, can be managed either by you, shared between you and the cloud service provider or by the cloud service provider.
- Typically a Key Management Infrastructure (KMI) manages the two aspects of keys:

(i) Securely storing the keys

(ii) Key Management : For example, allowing key usage or managing the key lifecycle

- There are four models in which you can manage the keys and the encryption process.

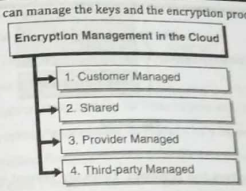


Fig. 5.7.9 : Encryption Management in the Cloud

1. **Customer Managed**

- In the customer managed model, you, as the customer, manage the key lifecycle, key storage and also the encryption / decryption process. You encrypt and decrypt your data entirely and the cloud service provider has no active role in it.

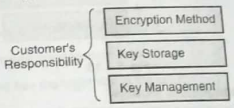


Fig. 5.7.10

- This is typically done in environments where the cloud service provider does not provide suitable encryption mechanisms or if you need to control your own processes due to laws or regulations or for any other security and trust reasons.

2. **Shared**

- In the shared model, you and the cloud service provider share the responsibility for encryption. Who does what can be mutually agreed upon? Usually, this is done to offload general tasks to the cloud service provider and you being in control of what matters – the encryption keys. The cloud service provider may not ever have access to the keys and may not manage its lifecycle or its usage.
- Following is an example of what responsibilities can be shared. Based on your agreement with the cloud service provider, there could be other responsibilities that it can take.

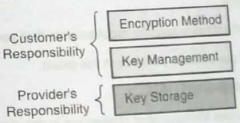


Fig. 5.7.11

3. **Provider Managed**

In the provider managed model, the entire responsibility of key lifecycle management, key storage, key usage and the encryption process can be handled by the cloud service provider. For you, it becomes extremely easy and convenient to ensure that your data is adequately protected without you having to manage any complexities behind it.

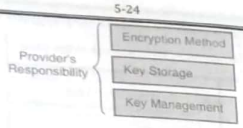


Fig. 5.7.12

4. Third-party Managed

A third-party can also manage your keys and processes. There are several vendors who specialize in key lifecycle management and have been traditionally providing key management solutions and encryption solutions. You could use them to control your encryption processes and manage your keys.

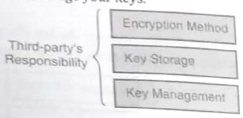


Fig. 5.7.13

Comparison between Encryption Management Models

Comparison Attribute	Customer Managed	Shared	Provider Managed	Third-party Managed
Complexity	High	Medium	Low	Low
Flexibility	High	Medium	Low	Low
Control	High	Medium	Low	Low

Review Questions

Here are a few review questions to help you gauge your understanding of this chapter. Try to attempt these questions and ensure that you can recall the points mentioned in the chapter.

[A] Basic Terms and Concepts

Q. 1 Describe principles for cloud security. [6 Marks]

Q. 2 What are some of the typical cloud software security requirements that you would consider. [6 Marks]

Q. 3 Write a short note on security and governance services in the cloud. [4 Marks]

[B] Cloud Identity and Access Management (IAM)

Q. 4 Explain the various Identity and Access Management challenges in the cloud. [6 Marks]

Q. 5 Describe the identity management lifecycle. [4 Marks]

Q. 6 What are the types of identity providers used in the cloud? [6 Marks]

Q. 7 Write a short note on Identity Broker. [4 Marks]

Q. 8 Compare the types of identity providers in the cloud. [6 Marks]

Q. 9 Suggest some of the best practices for cloud IAM. [4 Marks]

[C] Data Protection in Cloud

Q. 10 List the Data Security Concerns in the Cloud and briefly explain each. [8 Marks]

Q. 11 Where will the data be stored in the cloud" is a key data security concern. Explain in detail. [8 Marks]

Q. 12 How can you securely delete the data in the cloud? [4 Marks]

Q. 13 With a stack diagram, explain "Encryption Responsibility for Infrastructure as a Service (IaaS)". [6 Marks]

Q. 14 With a stack diagram, explain "Encryption Responsibility for Platform as a Service (PaaS)". [6 Marks]

Q. 15 With a stack diagram, explain "Encryption Responsibility for Software as a Service (SaaS)". [6 Marks]

Q. 16 Compare Shared Responsibility for Data Encryption based on type of cloud service. [5 Marks]

Q. 17 With block diagrams, explain the mechanisms for encrypting data in the cloud. [8 Marks]

Q. 18 Write a short note on Server-side encryption. [4 Marks]

Q. 19 Write a short note on Client-side encryption. [4 Marks]

Q. 19 Compare server side and client side encryption. [5 Marks]

Q. 20 Briefly describe the types of cloud storage that require encryption. [5 Marks]

Q. 21 Describe the models of encryption and key management in the cloud. [8 Marks]

Q. 22 Compare the models of encryption and key management in the cloud. [5 Marks]

6 Advanced Techniques in Cloud Computing

Syllabus

At the end of this unit, you should be able to understand and comprehend the following syllabus topics :

- Future Tends in cloud Computing
- Mobile Cloud
- Automatic Cloud Computing: Comet Cloud
- Multimedia Cloud: IPTV
- Energy Aware Cloud Computing
- Jungle Computing
- Distributed Cloud Computing Vs Edge Computing
- Containers
  - Docker
  - Kubernetes
  - Introduction to DevOps
- IOT and Cloud Convergence
  - The Cloud and IoT in your Home
  - The IOT and cloud in your Automobile
  - PERSONAL: IoT in Healthcare

6.1 Multimedia Cloud: IPTV

- If you have ever used Netflix, Amazon Prime, Apple TV, Hotstar or any other internal based media streaming service to watch content on your TV, smartphone, tablet or laptop, you have already experienced what an IPTV is!

✍ ***Definition :*** *IPTV is a media streaming service that allows you to play media content on any device over the internet.*

- This is also called Over-The-Top (OTT) platform. You can play content from various media providers. You can watch movies, play songs, watch sports or any other entertainment channels. Increasingly viewers are moving towards IPTV to get better control over what they watch.
- IPTV provides the following benefits (or applications) over traditional TV watching experience.

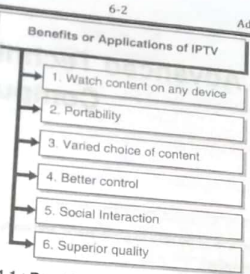


Fig. 6.1.1 : Benefits of Applications of IPTV

1. **Watch content on any device :** The content from IPTV based service can be watched on any device. You can play it on large screens or on your tiny smartphone screen. The device just needs to be able to connect to the internet and have the capability to play video and audio.
2. **Portability :** Because you can play content on any device, you are no more required to be physically hooked to the TV screen. You can watch the content on the move in taxis, trains or while waiting in a queue at a billing counter.
3. **Varied choice of content :** IPTV service providers get content from various media sources. Additionally, they produce special series based on customer demands and preferences. You have more choices than regular TV channels. Additionally, you can automatically convert the language as per your preference. For example, if a TV show is originally broadcasted in English, then you can change its language (as you watch) to Hindi or as per your personal language preference.
4. **Better Control :** With IPTV, you have a completely interactive experience with TV. You can pause, record, download live TV shows, movies, etc. and watch them as per your comfort and convenience. So, for example, if you are watching sports and the doorbell rings, you do not need to miss the sports action. You can just pause the live TV and resume it exactly from the same point. Another example could be that you can download a movie and store it for certain period of time on your device. You can then watch the movie even at places, where you do not have active internet, such as flights.
5. **Social Interaction :** With IPTV, you can comment and provide reviews on shows. You can socially engage with other viewers on the IPTV platform and make your watching experience more collective and involved.
6. **Superior quality :** IPTV can provide superior video and audio quality if you have good internet connectivity. The streaming services let you control the quality of the content to match your viewing experience, internet connectivity speed and optimising the cost for internet data consumption.

.2 Future Trends in Cloud Computing

The dictionary meaning of ubiquitous is "existing or being everywhere at the same time". Gone are the days when you required a physical and dedicated machine for computing. Now computing is available in various form factors such as watch, mobile phones, vehicles, doorbells, smart TV and what not. Anything and everything these days comes with a "smart" connectivity.

***Definition** :Ubiquitous computing refers to the concept of modern computing that does not require any special form factor, such as desktop or server, for computing.*

- Computing can be achieved on any device, anytime and anywhere. Ubiquitous computing is also sometimes shortened to be written as UbiCom. It is also called as pervasive computing. The dictionary meaning of the word pervasive is "existing in or spreading through every part of something". These devices are usually connected over network through various protocols such as Wi-Fi, Bluetooth or ZigBee. The core idea behind ubiquitous computing, as proposed by its father, Mark Weiser, is to make computing "invisible". A computer should help you to become efficient and carry out day to day tasks without you feeling its presence or your routine interrupted by it.
- There are several examples of ubiquitous computing. Some of them are smart watches, smart bulbs, smart thermostat, smart TV, smart refrigerators, and perhaps anything that comes with a prefix of "smart". These devices are increasingly being used and serve various purposes such as health monitoring, regulating temperature, capturing video or audio intelligently, etc. These devices are also commonly called as Internet of Things or IoT in short.
- Various cloud computing enabling technologies along with several developments from various vendors such as AWS, Microsoft, Google, VMware and IBM has made it further possible to enable and fuel the vision of ubiquitous computing. Today, several vendors provide specific as well as general IoT solutions to meet various ubiquitous computing requirements.
- For example, the following snapshot from AWS as shown in Fig. 6.2.1 provides various IoT services that it offers on its cloud computing environment.

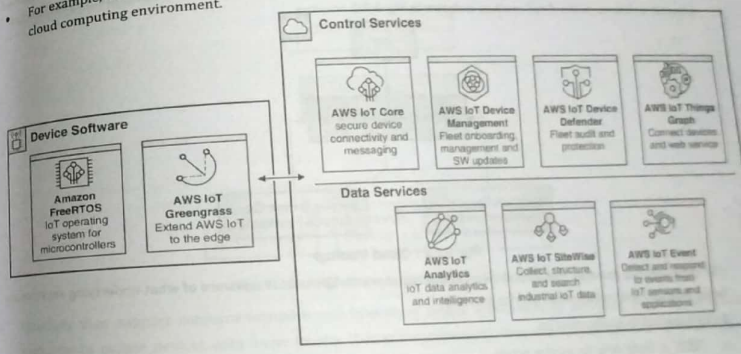


Fig. 6.2.1 : Snapshot of various IoT Services

6.2.1 Cloud Mashup

University Question

Q. Explain the concept of cloud mashup. Draw and explain architecture of IoT. **SPPU - May 19, 8 M**

- You would have heard mashup songs right? What is a mashup? The dictionary meaning of mashup is "so created by combining elements from two or more sources".
- For example, you might have used app-based taxi service such as Uber and OLA. How do you see the live lo the taxi on Google Map within the app? That's an example of a mashup. The app provider lays its data over Go and shows you a great view of how the taxi you booked is approaching you and in how much time. Another could be that you can use tweets from Twitter and plot them on Google Map to show the areas from where being sent.

- Mashups are typically created to provide combination, aggregation and visualisation.
  - **Combination :** In combination mashups, the data from various sources are collected and shown within one application. One example could be a flight booking or hotel booking website that compare the prices from various other websites and help you choose the cheapest or most relevant option.
  - **Aggregation :** In aggregation mashups, the collected data is further analysed (or processed) to provide more useful information. For example, a news app could show the most read news from around the world from various other news apps.
  - **Visualisation :** In visualisation mashups, the data is presented in a user-friendly way either on maps or any other preferred layout. For example, you could build an app to combine available rental flats from several real estate websites and plot them on Google Map.
- A similar approach is taken for creating a cloud mashup.

Definition : *A cloud mashup combines functionalities from various cloud services to create a new service.*

- You could combine a service from Google Cloud Platform and another service from Amazon Web Services to create a new service. The new service would consume the services from the respective cloud service providers as architected.
- A typical architecture of a cloud mashup is shown in Fig. 6.2.2.

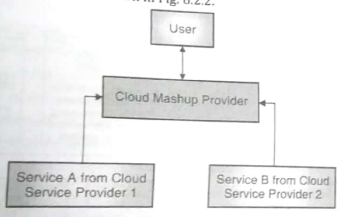


Fig. 6.2.2 : Cloud Mashup

- **The cloud mashup provider combines the services as appropriate. The user is unaware of what underlying services are being used and from which cloud service provider.**
- **A cloud mashup typically relies on,**
  - **REST or SOAP APIs for service access**
  - **JSON or XML for data exchange**

6.2.1(A) Advantages of Cloud Mashup

1. **You can create a new service combining several independent services.**
2. **The resulting mashup service might be more user-friendly.**
3. **Mashup can provide higher scalability and performance.**

6.2.1(B) Disadvantages of Cloud Mashup

1. **It requires a separate development time.**
2. **The cloud service provider may not maintain backward compatibility as its services evolve.**
3. **Different cloud service providers may have different SLAs. The overall SLA of your mashup service would then be the minimum of all the SLAs for the services that you use.**

6.2.2 Mobile Cloud Computing

University Questions

Q. Explain architecture of Mobile cloud computing with diagram. **SPPU - May 18, 10 Marks**

Q. Explain the concept of cloudlet. **SPPU - May 19, 8 Marks**

There are two potential issues with the number of mobile devices (think IoT as well) and the computing requirements.

1. There are various types of IoT, and mobile devices and they have limited computing power.
2. The cloud services might not meet the high response requirements of these devices.

Cloudlet solves these two problems by being an intermediary between the devices and the cloud.

Definition : *Cloudlet brings the cloud "closer" to the devices by providing an interface between the mobile devices and the cloud.*

Fig. 6.2.3 shows high-level architecture of cloudlet.

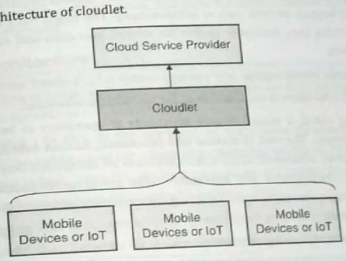


Fig. 6.2.3 : High-level Architecture of Cloudlet

- Cloudlets thus support resource-intensive and interactive mobile applications by providing powerful computing resources to mobile devices with lower latency (better response time). They synchronise the data with the regular cloud service continuously.
- Cloudlet typically runs the same service that is running in the cloud. It is just that it is closer to the user and thus provides a better response time to the user when compared with the distant cloud service accessed over the internet. Cloudlet is usually available over Wi-Fi or LAN and thus provides much better response. Cloudlet is also referred as "data centre-in-a-box" because it is self-contained and self-managed. Think of cloudlet as a mini-cloud operated by the business itself.

6.2.3 Comparison between Cloudlet and Cloud

University Question

Q. Differentiate between cloud and cloudlet. **SPPU - May 19, 8 Marks**

The Table 6.2.1 provides a comparison between cloudlet and cloud.

Table 6.2.1 : Comparison between Cloudlet and Cloud

Comparison Attribute	Cloud	Cloudlet
Managed by	The cloud service provider	The business itself
Connectivity	Over the Internet	Over LAN or Wi-Fi
Users	Several users worldwide	Just local users
State of data	Real and consistent data	Temporarily Cached data

6.3 Autonomic Cloud Engine

University Question

Q. Explain Autonomic cloud engine in detail. **SPPU - May 19, 6 Marks**

- The dictionary meaning of the word autonomic is "acting or occurring involuntarily". For example, your heart beats, your lungs breathe, and your eyelids shut and open on their own without requiring any thinking or action from your side. These actions just happen on their own or autonomously. You might have also heard about other autonomous systems such as self-driving cars, auto-pilot system for aircraft operation and robotics in factory. Autonomic cloud engine is a similar concept.

Definition : *Autonomic cloud engine is a concept under which cloud operations and management can be completely automated and there is minimal need for human involvement.*

- Operating at the scale of cloud, it would require thousands of operators to be available continuously to ensure uninterrupted user experience. The cost of manual jobs (slowness), human oversight and human errors could make cloud very expensive. Instead, autonomic cloud engine provides an automated way to deal with cloud operations and management. The regular maintenance and operations tasks are programmatically automated. Such autonomous handling requires a few human resources and do not require a constant supervision thus improving the overall system efficiency at an optimised cost.
- Autonomic cloud engine has the following characteristics.

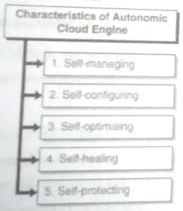


Fig. 6.3.1 : Characteristics of Autonomic Cloud Engine

1. **Self-managing :** Autonomic cloud engine can manage itself. System administrators are not required to intervene every time. For example, if a new upgrade is available, the system can evaluate if it would benefit from the upgrade and upgrade itself. If there are error or problems after upgrade, it can automatically revert itself to the last known good state.
2. **Self-configuring :** Autonomic cloud engine can configure themselves for best settings, security, business policies and with other performance enhancing parameters. For example, when a new module is added to the autonomous system, it can automatically understand its functionality and start to make the best use of it without requiring any configuration externally.

3. **Self-optimising :** Complex systems such as cloud can have thousands of settings. Various combinations of settings can alter the behaviour of the system. Autonomic systems can automatically choose the optimum configuration depending upon the situation. For example, if the demand is high, it can choose high performance settings but when the demand is low, it can choose more conservative settings.
4. **Self-healing :** It takes time for identifying, tracing, and determining the root cause of failures in complex computing systems. Autonomous systems, however, can detect, diagnose, and repair problems resulting from bugs or failures in software and hardware. Using the knowledge about the system configuration, it can analyse information from log files and other sources of information about the problem. It can then identify the solution to the diagnosed problem and self-heal itself.
5. **Self-protection :** Autonomous systems can proactively detect future problems and carry out maintenance tasks before the system actually breaks down. For example, if the hard disk is filling at a rate such that it would fill up 100% in 3 days, it can notify the administrator to add more hard disk capacity or if the hard disk capacity already exists, it can expand the file system automatically and use the extra hard disk space.

- I want to clarify here that autonomic cloud engine is a concept and something to continuously aim for. Truly autonomic cloud engine does not exist today, though; automation is heavily used by the cloud service providers to operate the cloud most efficiently.

6.4 Comet Cloud Architecture

- Comet cloud is an integration of public and private cloud. It supports on-demand scaling using public cloud. Basically, it uses two concepts for its operations :

1. **Autonomic Cloud Bursting :** In this, as the workload on the private datacentre (or private cloud) increases, it automatically starts to utilise public cloud for supporting additional workload. Public cloud resources are consumed until the workload demand is high. Once the situation normalises, the public cloud resources are no more used, and the operations only rely upon the private datacentre (or private cloud).
2. **Autonomic Cloud Bridging :** In this, instead of only switching to public cloud during high workload demand, a constant connection is established between the public cloud provider and the private datacentre. Cloud bridges enable connectivity between the datacentre and multiple cloud environments and extend the transparent network access to varied resources across different computing environments.

- The Fig. 6.4.1 a high-level block diagram of Comet Cloud Architecture.

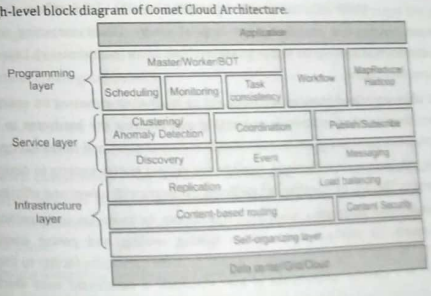


Fig. 6.4.1

- It has three layers.

1. **Infrastructure Layer :** This is the bottom most layer that houses the physical components such as servers and networking as well as software components such as load balancing.
2. **Service Layer :** The service layer has several functions such as event management, message notifications and coordination of tasks.
3. **Programming Layer :** The programming layer provides the basic framework for application development and management. It supports various frameworks for automation. For example, master servers can generate tasks and worker servers can consume them.

6.5 Energy Aware Cloud Computing

University Questions

Q. List and explain the key issues related to cloud computing energy efficiency.

Q. Explain key issues related to energy efficiency in cloud computing. SPPU - May 18, 8 Marks SPPU - May 19, 8 Marks

- Cloud providers use massive data centres that consume a lot of energy. However, if their energy consumption is distributed per user, it would turn out to be comparatively less if these users were to setup their own data centres.
- Energy efficiency in cloud computing comes from the following factors.

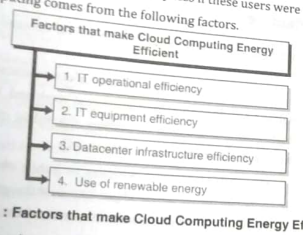


Fig. 6.5.1 : Factors that make Cloud Computing Energy Efficient

1. **IT operational efficiency :** Cloud computing provides IT operational efficiency through dynamic provisioning, multitenancy and higher server utilisation. If each organisation were to setup its own data centre, it would mean additional unused capacity, over provisioning and waste of energy. Cloud computing achieves better hardware utilisation by distributing and sharing the resources with multiple users (multitenancy). Less equipment on the planet means less energy consumption.
2. **IT equipment efficiency :** Cloud providers are motivated to lower their expenses on energy to reduce the overall operational cost. Consequently, they invest in customising and tailoring the hardware to have only the required components and choose higher energy efficient specifications over cheaper and lower energy efficient specifications. They collaborate with hardware manufacturers, suppliers and subject matter experts to design the servers and other datacentre equipment. They also recycle older and inefficient hardware with newer and efficient ones periodically.
3. **Datacentre infrastructure efficiency :** Advanced infrastructure technologies in hyperscale datacentres reduce electricity requirements for overhead tasks such as lighting, cooling, and power conditioning. Power usage effectiveness (PUE) - the ratio of overall electricity consumption at the datacentre facility to the electricity delivered to the IT hardware - is a common measurement of how efficiently a datacentre uses electricity. The hyperscale datacentres that power the cloud are able to achieve better PUEs than typical enterprise datacentres.

- **Use of renewable energy :** Increasingly the cloud service providers are shifting to renewable sources of energy such as solar and wind. They are setting up such solar and wind farms to further lower their impact on the environment and make cloud computing a greener choice. Such greener sources of energy reduce the dependence on non-renewable sources such as carbon and nuclear power plants

6.6 Jungle Computing

University Questions

Q. What is Jungle Computing? Explain why there is need of Jungle Computing? SPPU - Dec.18, 8 Marks

Q. Explain Jungle Computing in detail. SPPU - May 19, 6 Marks

- Various scientific research platforms require high performance computing resources.
- ***Definition :*** *Jungle computing is a technique to distribute the computational requirements across clusters, grids, supercomputers and cloud computing such that to balance performance and cost.*
- Based on the computational requirements, a jungle computing platform can automatically choose the target system to place the workload. For example, the results that are not needed immediately can be computed on a standalone machine whereas if you have to process terabytes of cancer cell research data you could use cloud computing platform or supercomputers.
- Following are some of the desired characteristics of jungle computing systems.

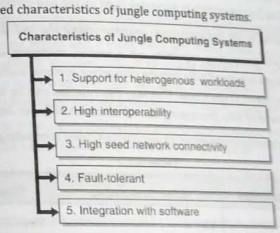


Fig. 6.6.1 : Characteristics of Jungle Computing Systems

1. **Support for heterogenous workloads :** As you read earlier, the computational needs could be very different from user to user and application to application. Jungle computing system should be able to run these workloads on any target platform without creating dependency.
2. **High interoperability :** The results from one computing resource could be fed into another computing resource. For example, you could transfer the results from a cloud computing environment to a standalone system for specific analysis and study. Hence, the computing resources in a jungle computing system should be interoperable.
3. **High speed network connectivity :** Any scientific research often involves massive datasets. High speed network connectivity is hence required to input and output the data sets. Also, as multiple systems can work together, high speed connectivity between them ensures that the systems can send and receive data from each other as required in a processing workflow.
4. **Fault-tolerant :** Jungle computing system should be able to keep track of which resources are available for computing and which are not. It should be able to handle failures without disrupting the workflow and processing. It should continuously watch out for the health of the resources and notify the operators if any failures are detected.

5. **Integration with software :** Scientific research often requires common software such as data processing, trend analytics and machine learning. Jungle computing systems should provide integration with such software so that the users do not have to setup the software before using the system. They should be able to just load the data and begin the computation.

6.7 Distributed Cloud Computing Vs Edge Computing

Definition : *A distributed computer system consists of multiple software components that are on multiple computers but run as a single system.*

- The computers that are in a distributed system can be physically close together and connected by a local network, or they can be geographically distant and connected by a wide area network. A distributed system can consist of any number of possible configurations, such as mainframes, personal computers, workstations, minicomputers, and so on. The goal of distributed computing is to make such a network work as a single computer. Today, cloud computing is largest form of distributed computing model.
- As there are further enhancements in silicon technology, the computing capabilities on even tiny devices are becoming quite powerful and could be handy to perform computation even further closer to actual data generating device than sending the entire data to the cloud or the distributed systems. The end-devices could perform some low-level computing tasks such as taking smart decisions by performing local analysis. This shift of computing to the actual end-device is called edge computing (as computing is occurring right on the end-device itself).

Definition : *Edge computing is the network layer encompassing the end-devices and their users.*

- Edge computing provides local computing capability on a sensor, metering or some other end-devices that are network-accessible. Edge systems are essentially remote computing systems such as smartphones, network gateways, or smart objects that work on behalf of the cloud. This way information can be shared quickly, securely and without latency. Plus it improves the speed of data processing as a direct result of lower dependency on the cloud.

The Table 6.7.1 provides a quick comparison between distributed cloud computing and edge computing.

Table 6.7.1

Comparison Attributes	Edge Computing	Cloud Computing
App Hosting	No	Yes
Flexible computing power	No	Yes
Resource Pooling	No	Yes
Real-time response	Yes	No
Fault Tolerance	No	Yes
Device dependent	Yes	No
Entire domain awareness	No	Yes
Cloud awareness	No	Yes
Controllers	Specific to edge	General
Security Scope	Limited to device	Up to Cloud Layer
Big Data Analytics	No	Yes

Comparison Attributes	Edge Computing	Cloud Computing
Scalability	Low	High
Use of virtualisation	No	Yes
Data Storage	No	Yes, Nearly unlimited

6.8 Docker at a Glance

University Question SPPU - May 18, Dec. 18, May 19, 2 Marks

Q. What is Docker?

Before you learn anything about Docker, you should understand how any application software was traditionally developed and run in the production.

Earlier,

1. Various developers used to write code and put it in code version control system such as GitHub.
2. Then, a separate build process would build the code.
3. Once the code build was ready, it was handed over to the operations team to deploy it in staging (pre-production testing environment).
4. Once the code passed the testing phase on staging environment, it would be deployed in the production.

- The entire process could take several weeks and if there were any issues found during testing or production, it was difficult to roll back the changes again. It was also difficult to ensure that the code would run exactly the same as the developer wanted. The operations team controlled the execution environment (OS, libraries, runtime environment, etc.) and often developer would be frustrated that what works on her laptop, does not work on the target system.
- To speed up the entire developer to production deployment process and to ensure that the application software can be shipped without worrying about the execution environment, Docker was born!

Definition : *Docker is a platform for developers and sysadmins to develop, deploy, and run applications with containers.*

- The use of Linux containers to deploy applications is called containerization. Containers are not new, but their use for easily deploying applications is. Linux containers use kernel namespace for isolating one container's runtime from another container's runtime environment. The isolation is provided for,

1. Inter-process communication (IPC)
2. Unix timesharing (UTS)
3. Mount points
4. Processes (PID)
5. Network and
6. User

6.8.1 Architecture of Docker

University Questions

Q. Draw and explain Docker deployment workflow. SPPU - May 18, 6 Marks

Q. Draw architecture for Docker and explain its components. SPPU - Dec. 18, 8 Marks

- Containerization is a virtualization technology at the OS level. Docker provides the virtualization layer on top of the host operating system. Unlike VMs, it does not require a hypervisor layer. However, the host OS can itself be virtualized or may not be virtualized – does not matter for Docker.
- The Fig. 6.8.1 shows a high-level block diagram for Docker. I have also placed a diagram for VM side-by-side for your clarity.

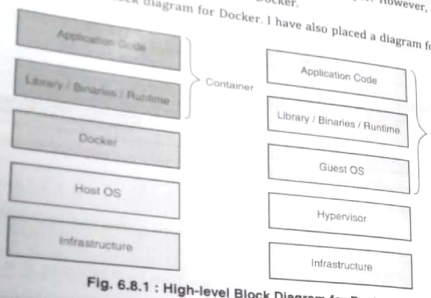


Fig. 6.8.1 : High-level Block Diagram for Docker

- **A container runs natively on Linux and shares the kernel of the host machine with other containers. It runs as a lightweight process taking limited resources.**
- **A Virtual Machine (VM) runs a full-blown guest operating system with virtual access to host's hardware resources through the hypervisor.**

6.8.2 Building a Docker Image

- **You need to build a Docker image with all the runtime libraries and your application code. You also need to choose the base OS runtime image that your application would use. The base OS image provides the execution environment for your application.**
- **Let's create very simple Docker image.**

```
FROM centos
MAINTAINER Pravin Goyal
LABEL Description="This is a sample image for cloud computing book readers."
ENTRYPOINT echo "I am running inside a container."
```

- **You should name the file as *Dockerfile* and place it in the directory from where you will execute the *docker build command*. Let's build the Docker image. I want to call my Docker image as *sample_docker*.**

```
PS:> docker build . -t sample_docker
Sending build context to Docker daemon  6.123MB
Step 1/4 : FROM centos
 ---> 9f38484d220f
Step 2/4 : MAINTAINER Pravin Goyal
 ---> e37781ee2090
```



```
 ---> 40ee92b27e52
Step 4/4 : ENTRYPOINT echo "I am running inside a container."
 ---> b2865856c41b
Successfully built b2865856c41b
Successfully tagged sample_docker:latest
```

- As you notice, the Docker image consists of several layers – one for each line in the Dockerfile.
- Once the image is created, you can see it using the *docker images* command. You can then run it using the *docker run* command.

```
PS> docker images
REPOSITORY   TAG   IMAGE ID      CREATED         SIZE
sample_docker latest b2865856c41b  2 minutes ago   202MB
PS> docker run sample_docker
I am running inside a container.
PS>
```

- Note here that I have presented a highly simplified example here. The real life Dockerfile could have 500+ lines and the container image when run could run the entire app! Also, there are several commands and options for defining Dockerfile, building the image and then running it.

6.8.3 Docker Workflow

University Question

Q. Draw and explain Docker deployment workflow. **SPPU - Dec. 18, May 19, 6 Marks**

- With Docker, the entire pipeline, from the developer's code check-in to production deployment, can be automated. Typically, Docker-based applications have the following workflow for development, testing and deployment. This is also called CI/CD pipeline that you read about earlier in this chapter under "Faster Time to Market for Software Applications".
- The Fig. 6.8.2 shows a high-level diagram for Docker Workflow.

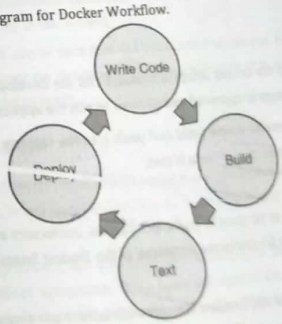


Fig. 6.8.2 : Docker Workflow

through each of the steps and learn what goes-in.

Write Code

As usual, you start writing code for your application. You can choose any preferred programming language and any combination of development stack based on your application requirements. Docker containers are extremely popular, and it is hard to find a popular development stack or language not having Docker container support. You use code version control systems such as GitHub, Bitbucket, or cloud based service such as AWS Code Commit to manage your code. You can also use your own code version control system in-house.

2. Build

Once you have a version of code that you believe is good to go, you start the build process. The build process is very similar to the one that you learnt about earlier using *Dockerfile* and *docker build* command. In the build stage, you typically carry out the following activities.

(a) Get the latest copy of your code that you want to build.

(b) Get the list of all dependencies, binaries, libraries and runtime environment information.

(c) Write the *Dockerfile* with the build instructions.

(d) Build the Docker image.

(e) Push the Docker image to a registry (storage location) from where next steps of workflow can be carried out.

3. Test

Once your Docker image is ready, you deploy the image for testing your application. Deploying the image means creating containers and running them (as you learnt earlier using *docker run* command). The application can undergo several tests as you may have defined to ensure that it meets the quality and performance criteria. It could be,

(a) Functional testing

(b) White-box testing

(c) Black-box testing

(d) Regression testing

(e) Scale testing

(f) Security testing

If there are bugs identified, then the docker image is rejected and the developer is notified of the bugs or issues that must be resolved before the Docker image is approved and allowed to run the application in the production environment.

Once you fix the bugs, you again create a new build and push it to the registry to begin the testing process again and proceed to the next step if the fit-for-production criteria is met.

4. Deploy

Once the Docker image is approved to be good to go for production, containers are created and run as per the desired scale. It could be that you want to deploy 50 container instances of the Docker image to appropriately meet the expected application workload.

If the image has the new version of the application, the older containers are terminated and replaced with the newer containers while the application is live. You can upgrade the application automatedly without downtime.

Docker containers also provide a mechanism to roll back the application update. If any issues are identified in the production, the new deployment can be quickly rolled back, and the previously approved version of the Docker image can be used to create new containers running the previously approved application version.

6.8.4 Process Simplification

As you learnt from the Docker workflow, it has simplified the way the applications are developed, built, shipped, tested and deployed. Docker has brought several process simplifications.

Some of them are as following.

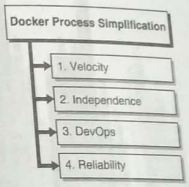


Fig. 6.8.3 : Docker Process Simplification

1. **Velocity :** The entire pipeline from code development to deployment can be automated. What used to take days to deploy can now be deployed in minutes if not seconds. You have already learnt about the CI/CD pipeline and how can it be used to automate the application deployment process.
2. **Independence :** The dependency on the operations team to deploy the application is over. Operations team, now, just provides the infrastructure and the developers deploy and manage the applications themselves.
3. **DevOps :** A new working model called DevOps has combined development and operations. DevOps is the combination of cultural philosophies, practices, and tools that increases an organisation's ability to deliver applications and services at high velocity: evolving and improving products at a faster pace than organisations using traditional software development and infrastructure management processes. This speed enables organisations to better serve their customers and compete more effectively in the market.
4. **Reliability :** With Docker, developers can be sure that the application would run exactly the same as it runs on their development machine. The dependency on the target runtime environment is over. They package the application along with all the required runtime components..

6.8.5 Broad Support and Adoption

- Docker containers have become extremely popular with application developers and organisations. All the major public cloud service providers support it. It has become the new de facto way to run and host applications specially in the cloud environment.
- Following are some facts and Fig. 6.8.4, 6.8.5 and 6.8.6 that should give you a quick understanding of the broad support and adoption of the container ecosystem. These facts and figures are collected from Cloud Native Computing Foundation (CNCF) Survey 2018 available at https://www.cncf.io/blog/2018/08/29/cncf-survey-use-of-cloud-native-technologies-in-production-has-grown-over-200-percent/.

1. Adoption of Container Deployment Environment

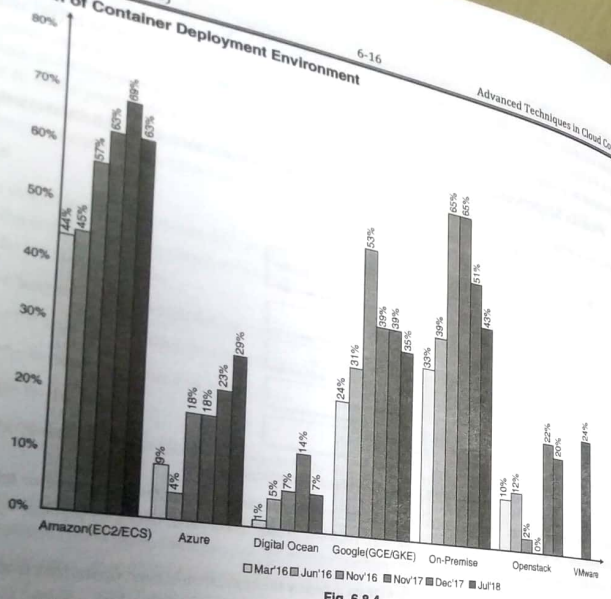


Fig. 6.8.4

2. Number of Containers Organisations Typically Run

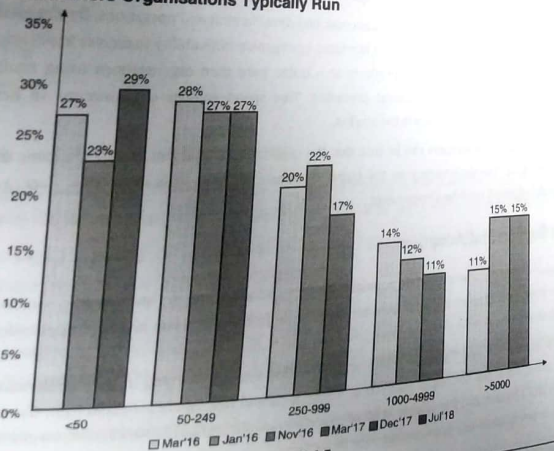


Fig. 6.4.5

3. Organisations Typically Manage Containers with,

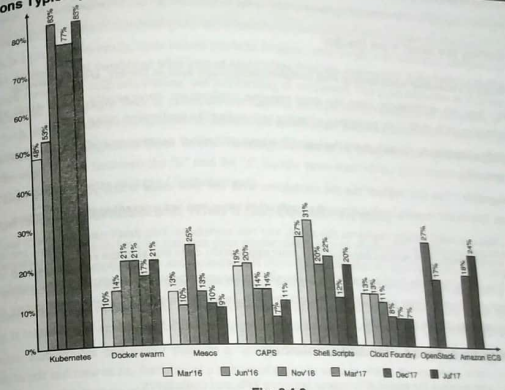


Fig. 6.4.6

4. Docker Hosted Images

Docker hosts the largest repositories of official Docker images across various categories.

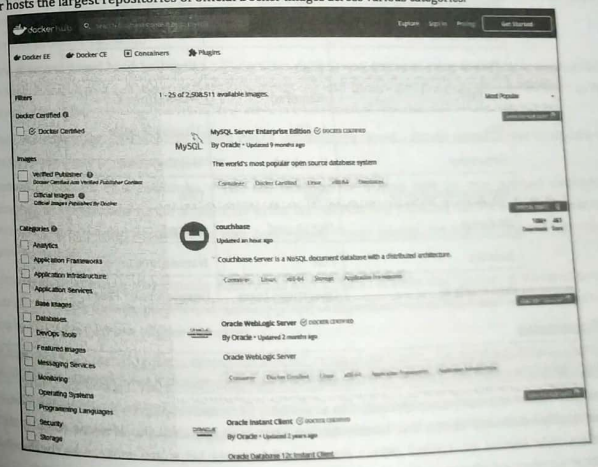


Fig. 6.4.7

You can find images for almost anything that is worth using in software applications today

6.8.6 Getting the Most from Docker

A few things to remember when dealing with container technology, such as Docker, are as following,

1. **Flexible :** You can containerize even the most complex applications. Increasingly, lot of applications are getting containerized to get benefit from packaging, shipping and scaling the application using Docker.
2. **Lightweight :** Containers use and share the host OS kernel with other containers and the OS. Containers are quick to deploy and scale.
3. **Interchangeable :** You can replace the old containers with the new ones when you have a new version of your application. You do not require to shut down the application when you are upgrading the application.
4. **Portable :** You can build locally, deploy to the cloud, and run anywhere. You don't need to worry too much about the execution environment since you package not only your application code but also the required runtime environment.
5. **Scalable :** It is easier to scale containers either manually or automatically. For example, if the number of requests to your application increases, you can add more containers to serve the load and when the number of requests goes down, you can terminate the containers.

6.8.7 Comparison between VMs and Containers

The Table 6.8.1 summarises the major differences between VMs and containers.

Table 6.8.1 : Comparison between VMs and Containers

Comparison Attribute	VMs	Containers
Virtualization Level	Hardware	OS
Time to create, start and stop	High	Very low
Full OS	Required	Not Required
Size	In GBs	Usually in MBs
Hypervisor	Required	Not Required
Resource consumption	High	Low
OS Kernel	Not Shared	Shared
Isolation	Full	Namespace level
Used mostly by	Operations team	Development Team

6.9 Kubernetes

Note: Kubernetes is a huge topic that demands its own books. The coverage here is for your reference only.

- With modern web services, users expect applications to be available 24/7, and developers expect to deploy new versions of those applications several times a day. Containerization helps package software to serve these goals, enabling applications to be released and updated without downtime. Kubernetes helps you make sure those containerized applications run where and when you want and helps them find the resources and tools they need to work.

- **Definition :** *Kubernetes is an open source container orchestration platform developed by Google for managing microservices or containerized applications across a distributed cluster of nodes.*
- Kubernetes is a production-ready, open source platform designed with Google's accumulated experience over 15 years in container orchestration, combined with best-of-breed ideas from the community. Kubernetes is highly resilient and supports zero downtime, rollback, scaling, and self-healing of containers. The main objective of Kubernetes is to hide the complexity of managing a fleet of containers. It can run on bare metal machines or on public or private cloud platforms such as AWS, Azure and OpenStack. Kubernetes architecture follows a client-server architecture.
- The name Kubernetes originates from Greek, meaning helmsman or pilot. K8s as an abbreviation results from counting the eight letters between the "K" and the "s". Google open-sourced the Kubernetes project in 2014. You can learn more about Kubernetes at https://kubernetes.io.

6.9.1 Why you need Kubernetes and What it Can Do (Advantages)

- Containers are a good way to bundle and run your applications. In a production environment, you need to manage the containers that run the applications and ensure that there is no downtime. For example, if a container goes down, another container needs to start. Wouldn't it be easier if this behaviour was automatically handled by a system?
- That's where Kubernetes comes to the rescue! Kubernetes provides you with a framework to run distributed systems resiliently. It takes care of scaling and failover for your application, provides deployment patterns, and more.
- For example, Kubernetes can easily manage a canary deployment for your system. A canary deployment pattern allows you to roll out new code/features to a subset of users as an initial test. For example, if you want to check if your users like the new changes on your website or not, then you can direct, for example, 20% of your users to new website and 80% to the previous ones. As you gather more feedback, you can begin changing this traffic re-direction and can direct more and more users to new website. If your users don't like the new website, you can re-direct the traffic back to previous one.
- Kubernetes provides you with:

1. **Service discovery and load balancing :** Kubernetes can expose a container using the DNS name or using their own IP address. If traffic to a container is high, Kubernetes can load balance and distribute the network traffic so that the deployment is stable.
2. **Storage orchestration :** Kubernetes allows you to automatically mount a storage system of your choice, such as local storages, public cloud providers, and more.
3. **Automated rollouts and rollbacks :** You can describe the desired state for your deployed containers using Kubernetes, and it can change the actual state to the desired state at a controlled rate. For example, you can automate Kubernetes to create new containers for your deployment, remove existing containers and adopt all their resources to the new container.
4. **Automatic bin packing :** You provide Kubernetes with a cluster of nodes that it can use to run containerized tasks. You tell Kubernetes how much CPU and memory (RAM) each container needs. Kubernetes can fit containers onto your nodes to make the best use of your resources.
5. **Self-healing :** Kubernetes restarts containers that fail, replaces containers, kills containers that don't respond to your user-defined health check, and doesn't advertise them to clients until they are ready to serve. You can also set policies such as that anytime at least 5 container instances for a service must be running. If any container instance crashes Kubernetes can automatically bring up new container instances to match the desired number of container instances

6. **Secret and configuration management** : Kubernetes lets you store and manage sensitive information, such as passwords, OAuth tokens, and SSH keys. You can deploy and update secrets and application configuration without rebuilding your container images, and without exposing secrets in your stack configuration.

6.9.2 Architecture / Components of Kubernetes

The Fig. 6.9.1 shows high-level architecture of Kubernetes.

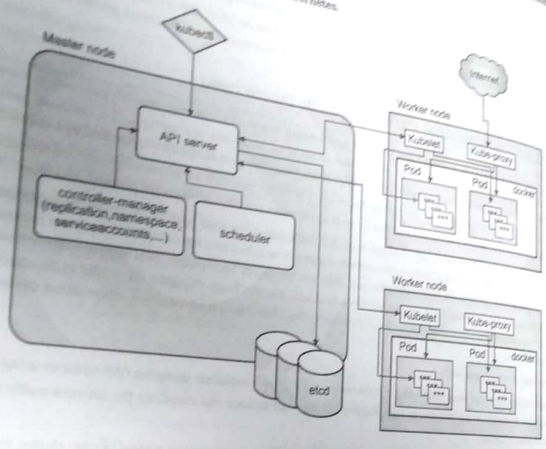


Fig. 6.9.1

- When you deploy Kubernetes, you get a cluster. A Kubernetes cluster consists of a set of worker machines, called nodes, that run containerized applications. Every cluster has at least one worker node. The worker node(s) host the Pods that are the components of the application workload. The master node manages the worker nodes and the Pods in the cluster. In production (real-life) environments, master node components control plane usually run across multiple computers and a cluster usually runs multiple nodes, providing fault-tolerance and high availability.
- Master node typically has the following components.
  - **etcd cluster** : a distributed key value storage that stores Kubernetes cluster data
  - **kube-apiserver** : the central management entity that receives all REST requests for modifications to cluster elements
  - **kube-controller-manager** : runs controller processes like replication controller (sets number of replicas in a pod) and endpoints controller (populates services, pods and other objects)
  - **cloud-controller-manager** : responsible for managing controller processes with dependencies on the underlying cloud provider

- **kube-scheduler** : helps schedule the pods (a co-located group of containers inside which our application processes are running) on the cluster nodes based on resource utilization

- Worker node typically has the following components.
  - **kubelet** : the main service on a node, taking in new or modified pod specifications from kube-apiserver, and ensuring that pods and containers are healthy and running
  - **kube-proxy** : runs on each worker node to deal with individual host subnetting and expose services
- kubectl is a command line tool that interacts with kube-apiserver and send commands to the master node. Each command is converted into an API call.

6.10 Introduction to DevOps

DevOps is a set of practices, tools, and a cultural philosophy that automate and integrate the processes between software development and IT teams. It emphasizes team empowerment, cross-team communication and collaboration, and technology automation. The DevOps movement began around 2007 when the software development and IT operations communities raised concerns about the traditional software development model, where developers who wrote code worked apart from operations who deployed and supported the code. The term DevOps, a combination of the words - development and operations, reflects the process of integrating these disciplines into one, continuous process.

6.10.1 How does a DevOps Team Work?

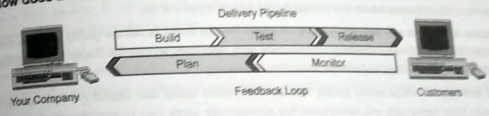


Fig. 6.10.1

- A DevOps team includes developers and IT operations working collaboratively throughout the product lifecycle, in order to increase the speed and quality of software deployment. It is a new way of working, a cultural shift, that has significant implications for teams and the organizations they work for.
- Under a DevOps model, development and operations teams are no longer "siloed." Sometimes, these two teams merge into a single team where the engineers work across the entire application lifecycle — from development and test to deployment and operations — and have a range of multidisciplinary skills.
- DevOps teams use tools to automate and accelerate processes, which helps to increase reliability. A DevOps toolchain helps teams tackle important DevOps fundamentals including continuous integration, continuous delivery, automation, and collaboration.
- DevOps values are sometimes applied to teams other than development. When security teams adopt a DevOps approach, security is an active and integrated part of the development process. This is called DevSecOps.

6.10.2 The DevOps Lifecycle

- Because of the continuous nature of DevOps, practitioners use the infinity loop to show how the phases of the DevOps lifecycle relate to each other. Despite appearing to flow sequentially, the loop symbolises the need for constant collaboration and iterative improvement throughout the entire lifecycle.

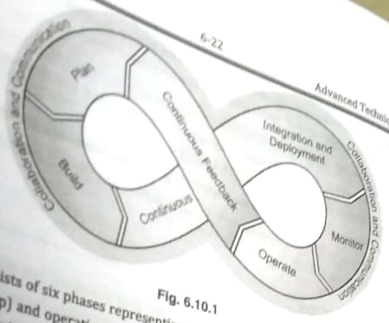


Fig. 6.10.1

- The DevOps lifecycle consists of six phases representing the processes, capabilities, and tools needed for development (on the left side of the loop) and operations (on the right side of the loop). Throughout each phase, teams collaborate and communicate to maintain alignment, velocity, and quality.

1. **Plan :** DevOps teams adopt agile practices to improve speed and quality. Agile is an iterative approach to project management and software development that helps teams break work into smaller pieces to deliver incremental value.
2. **Build :** Git is a free and open source version control system. It offers excellent support for branching, merging, and rewriting repository history, which has led to many innovative and powerful workflows and tools for the development build process. DevOps teams can use any version control system in place of Git.
3. **Continuous integration and delivery (CI/CD) :** It allows DevOps teams to release quality products frequently and predictably, from source code repository to production with automated workflows. Teams can merge code changes frequently, deploy feature flags, and incorporate end-to-end testing.
4. **Monitor and alert :** DevOps teams quickly identify and resolve issues that impact product uptime, speed, and functionality. They deploy tools and mechanisms that automatically notify your team of changes, high-risk actions, or failures, so you can keep services on.
5. **Operate :** DevOps teams manage the end-to-end delivery of IT services to customers. This includes the practices involved in design, implementation, configuration, deployment, and maintenance of all IT infrastructure that supports an organization's services.
6. **Continuous feedback :** DevOps teams evaluate each release and generate reports to improve future releases. By gathering continuous feedback, teams can improve their processes and incorporate customer feedback to improve the next release.

6.10.3 Advantages of DevOps

The DevOps model of providing IT services provides the following benefits.

1. **Speed :** Teams that practice DevOps release deliverables more frequently, with higher quality and stability. In fact, the DORA 2019 State of DevOps report found that elite teams deploy 208 times more frequently and 106 times faster than low-performing teams. Continuous delivery allows teams to build, test, and deliver software with automated tools.
2. **Improved Collaboration :** The foundation of DevOps is a culture of collaboration between developers and operations teams, who share responsibilities and combine work. This makes teams more efficient and saves time related to work handoffs and creating code that is designed for the environment where it runs.

3. **Rapid Deployment :** By increasing the frequency and velocity of releases, DevOps teams improve products rapidly. A competitive advantage can be gained by quickly releasing new features and repairing bugs. The quicker you can release new features and fix bugs, the faster you can respond to your customers' needs and build competitive advantage. Continuous integration and continuous delivery are practices that automate the software release process, from build to deploy.
4. **Quality and reliability :** Ensure the quality of application updates and infrastructure changes so you can reliably deliver at a more rapid pace while maintaining a positive experience for end users. Use practices like continuous integration and continuous delivery to test that each change is functional and safe. Monitoring and logging practices help you stay informed of performance in real-time.
5. **Security :** By integrating security into a continuous integration, continuous delivery, and continuous deployment pipeline, DevSecOps is an active, integrated part of the development process. Security is built into the product by integrating active security audits and security testing into agile development and DevOps workflows.

6.10.4 Challenges of Adopting DevOps

Some of the challenges of adoption DevOps model are as following.

1. **Old habits and mindset :** Habits are hard to break. Teams rooted in siloed (confined to themselves) ways of working can struggle with, or even be resistant to, overhauling team structures to embrace DevOps practices.
2. **Over focus only on tools :** Some teams may mistakenly believe that new tools are sufficient to adopt DevOps. Yet, DevOps is a combination of people, tools, and culture. Everyone on a DevOps team must understand the entire value stream — from ideation, to development, to the end user experience. It requires breaking down silos in order to collaborate throughout the product lifecycle.
3. **Operational workload :** Moving from a legacy infrastructure to using Infrastructure as Code (IaC) and microservices can offer faster development and innovation, but the increased operational workload can be challenging. It is best to build out a strong foundation of automation, configuration management, and continuous delivery practices to help ease the load.

6.10.5 DevOps Practices

There are a few key practices that help organisations innovate faster through automating and streamlining the software development and infrastructure management processes. Most of these practices are accomplished with proper tooling. Let's learn about them in brief.

1. **Continuous integration(CI) :** Continuous integration is the practice of automating the integration of code changes into a software project. It allows developers to frequently merge code changes into a central repository where builds and tests are executed. This helps DevOps teams address bugs quicker, improve software quality, and reduce the time it takes to validate and release new software updates.
2. **Continuous delivery(CD) :** Continuous delivery is a software development practice where code changes are automatically built, tested, and prepared for a release to production. It expands upon continuous integration by deploying all code changes to a testing environment and/or a production environment after the build stage. When continuous delivery is implemented properly, developers will always have a deployment-ready build artifact that has passed through a standardized test process.
3. **Microservices :** The microservices architecture is a design approach to build a single application as a set of small services. Each service runs in its own process and communicates with other services through a well-defined interface using a lightweight mechanism, typically an HTTP-based application programming interface (API). Microservices are built around business capabilities; each service is scoped to a single purpose. You can use different frameworks or programming languages to write microservices and deploy them independently, as a single service, or as a group of services.

4. **Situational awareness :** It is vital for every member of the organisation to have access to the data they need to do their job as effectively and quickly as possible. Team members need to be alerted of failures in the deployment pipeline — whether systemic or due to failed tests and receive timely updates on the health and performance of applications running in production. Metrics, logs, traces, monitoring, and alerts are all essential sources of feedback teams need to inform their work.
5. **Infrastructure as Code :** Infrastructure as code is a practice in which infrastructure is provisioned and managed using code and software development techniques, such as version control and continuous integration. The cloud's API-driven model enables developers and system administrators to interact with infrastructure programmatically, and at scale, instead of needing to manually set up and configure resources. Thus, engineers can interface with infrastructure using code-based tools and treat infrastructure in a manner similar to how they treat application code. Because they are defined by code, infrastructure and servers can quickly be deployed using standardised patterns, updated with the latest patches and versions, or duplicated in repeatable ways.
   a) **Configuration Management :** Developers and system administrators use code to automate operating system and host configuration, operational tasks, and more. The use of code makes configuration changes repeatable and standardized. It frees developers and systems administrators from manually configuring operating systems, system applications, or server software.
   b) **Policy as Code :** With infrastructure and its configuration codified with the cloud, organisations can monitor and enforce compliance dynamically and at scale. Infrastructure that is described by code can thus be tracked, validated, and reconfigured in an automated way. This makes it easier for organisations to govern changes over resources and ensure that security measures are properly enforced in a distributed manner (e.g. information security or compliance with PCI-DSS or HIPAA). This allows teams within an organisation to move at higher velocity since non-compliant resources can be automatically flagged for further investigation or even automatically brought back into compliance.
6. **Monitoring and Logging :** Organisations monitor metrics and logs to see how application and infrastructure performance impacts the experience of their product's end user. By capturing, categorising, and then analysing data and logs generated by applications and infrastructure, organisations understand how changes or updates impact users, shedding insights into the root causes of problems or unexpected changes. Active monitoring becomes increasingly important as services must be available 24/7 and as application and infrastructure update frequency increases. Creating alerts or performing real-time analysis of this data also helps organizations more proactively monitor their services.
7. **Communication and Collaboration :** Increased communication and collaboration in an organisation is one of the key cultural aspects of DevOps. The use of DevOps tooling and automation of the software delivery process establishes collaboration by physically bringing together the workflows and responsibilities of development and operations. Building on top of that, these teams set strong cultural norms around information sharing and facilitating communication through the use of chat applications, issue or project tracking systems, and wikis. This helps speed up communication across developers, operations, and even other teams like marketing or sales, allowing all parts of the organization to align more closely on goals and projects.

6.11 Internet of Things (IoT)

- I am sure you would have heard of several smart devices such as smart watches, smart bulbs, smart thermostat, smart TV, smart refrigerators, and perhaps anything that comes with a prefix of "smart". These devices are increasingly being used and serve various purposes such as health monitoring, regulating temperature, capturing video or audio intelligently, etc. These devices are also commonly called as Internet of Things or IoT in short.

Definition : The Internet of Things (IoT) is a collection of diverse technologies that interact with the physical world.

- In a nutshell, IoT refers to not only smart devices but also the underlying technologies that enable and connect these devices to the internet. There are billions of such devices in homes, schools, organisations, factories, oil wells, hospitals, cars, and several other places. With the explosion of these connected devices, you increasingly need solutions to connect them, and collect, store, and analyse device data to gain powerful and actionable insights.
- The Fig. 6.11.1 shows the broad spectrum of common IoT devices used either directly by the consumer or in an industry setting.

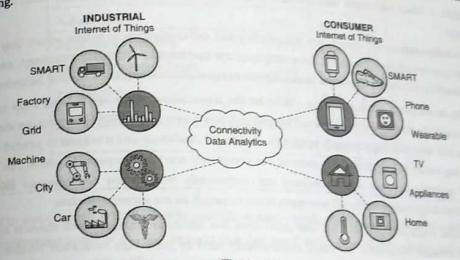


Fig. 6.11.1

- In this section, you will learn about several technologies, protocols and architectural solutions that make IoT possible.

6.11.1 Characteristics of IoT

- The major characteristics of IoT are as shown in Fig. 6.11.2.

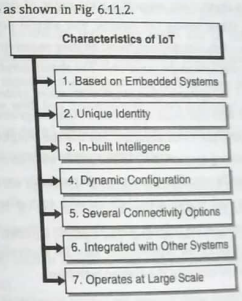


Fig. 6.11.2

1. **Based on Embedded Systems :** IoT devices are typically based on embedded systems and inherit several of their characteristics such as real-timeliness, low power consumption, high reliability, tolerance for harsh environments and small form factor. These devices perform specific functions such as reading temperature or controlling lights. They typically run specific Operating Systems (OS) as appropriate for the task they need to carry out.

4. **Situational awareness :** It is vital for every member of the organisation to have access to the data they need to do their job as effectively and quickly as possible. Team members need to be alerted of failures in the deployment pipeline — whether systemic or due to failed tests and receive timely updates on the health and performance of applications running in production. Metrics, logs, traces, monitoring, and alerts are all essential sources of feedback teams need to inform their work.
5. **Infrastructure as Code :** Infrastructure as code is a practice in which infrastructure is provisioned and managed using code and software development techniques, such as version control and continuous integration. The cloud's API-driven model enables developers and system administrators to interact with infrastructure programmatically, and at scale, instead of needing to manually set up and configure resources. Thus, engineers can interface with infrastructure using code-based tools and treat infrastructure in a manner similar to how they treat application code. Because they are defined by code, infrastructure and servers can quickly be deployed using standardised patterns, updated with the latest patches and versions, or duplicated in repeatable ways.
   a) **Configuration Management :** Developers and system administrators use code to automate operating system and host configuration, operational tasks, and more. The use of code makes configuration changes repeatable and standardized. It frees developers and systems administrators from manually configuring operating systems, system applications, or server software.
   b) **Policy as Code :** With infrastructure and its configuration codified with the cloud, organisations can monitor and enforce compliance dynamically and at scale. Infrastructure that is described by code can thus be tracked, validated, and reconfigured in an automated way. This makes it easier for organisations to govern changes over resources and ensure that security measures are properly enforced in a distributed manner (e.g. information security or compliance with PCI-DSS or HIPAA). This allows teams within an organisation to move at higher velocity since non-compliant resources can be automatically flagged for further investigation or even automatically brought back into compliance.
6. **Monitoring and Logging :** Organisations monitor metrics and logs to see how application and infrastructure performance impacts the experience of their product's end user. By capturing, categorising, and then analysing data and logs generated by applications and infrastructure, organisations understand how changes or updates impact users, shedding insights into the root causes of problems or unexpected changes. Active monitoring becomes increasingly important as services must be available 24/7 and as application and infrastructure update frequency increases. Creating alerts or performing real-time analysis of this data also helps organizations more proactively monitor their services.
7. **Communication and Collaboration :** Increased communication and collaboration in an organisation is one of the key cultural aspects of DevOps. The use of DevOps tooling and automation of the software delivery process establishes collaboration by physically bringing together the workflows and responsibilities of development and operations. Building on top of that, these teams set strong cultural norms around information sharing and facilitating communication through the use of chat applications, issue or project tracking systems, and wikis. This helps speed up communication across developers, operations, and even other teams like marketing or sales, allowing all parts of the organization to align more closely on goals and projects.

6.11 Internet of Things (IoT)

- I am sure you would have heard of several smart devices such as smart watches, smart bulbs, smart thermostat, smart TV, smart refrigerators, and perhaps anything that comes with a prefix of "smart". These devices are increasingly being used and serve various purposes such as health monitoring, regulating temperature, capturing video or audio intelligently, etc. These devices are also commonly called as Internet of Things or IoT in short.

Definition : *The Internet of Things (IoT) is a collection of diverse technologies that interact with the physical world.*

- In a nutshell, IoT refers to not only smart devices but also the underlying technologies that enable and connect these devices to the internet. There are billions of such devices in homes, schools, organisations, factories, oil wells, hospitals, cars, and several other places. With the explosion of these connected devices, you increasingly need solutions to connect them, and collect, store, and analyse device data to gain powerful and actionable insights.
- The Fig. 6.11.1 shows the broad spectrum of common IoT devices used either directly by the consumer or in an industry setting.

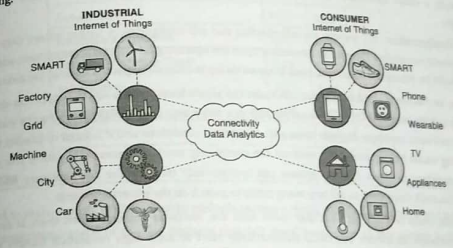


Fig. 6.11.1

- In this section, you will learn about several technologies, protocols and architectural solutions that make IoT possible.

6.11.1 Characteristics of IoT

- The major characteristics of IoT are as shown in Fig. 6.11.2.

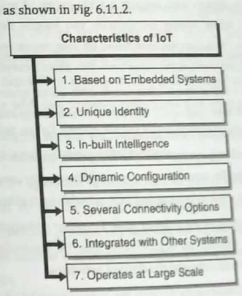


Fig. 6.11.2

1. **Based on Embedded Systems :** IoT devices are typically based on embedded systems and inherit several of their characteristics such as real-timeliness, low power consumption, high reliability, tolerance for harsh environments and small form factor. These devices perform specific functions such as reading temperature or controlling lights. They typically run specific Operating Systems (OS) as appropriate for the task they need to carry out.

Unique Identity : You need to control, operate, and manage IoT devices remotely via various connectivity options such as Wi-Fi and Bluetooth. Hence, these devices have unique identities such as IP addresses, MAC addresses, serial numbers, etc. using which you can precisely address a specific device and can either read output from the device (say temperature reading) or provide input (to turn off or turn on) to the device.

3. **In-built Intelligence :** As you understand that IoT devices are built on top of embedded systems, these devices are packed with in-built intelligence in the form of specific algorithms, situation-to-action mapping, and other intelligent mechanisms that can sense the environmental conditions and take specific actions or provide the desired response. For example, a smart bulb could automatically detect when you enter and leave the room and power on or off accordingly.

4. **Dynamic Configuration :** IoT devices are self-adapting and self-configuring in nature where they can appropriately and dynamically reconfigure themselves depending upon the operating environment, situations, and conditions. Their operation could be automated where you need not provide input controls to get the desired behaviour.

For example, an automatic watering system can water the plants based on the soil moisture. On sunny days, the soil might dry up quickly leading to frequent watering and on a non-sunny day, it might only require watering twice a day. How fast the soil loses moisture depends upon a lot of external conditions and it is hard to predict the precise timing when watering would be required. Such systems can completely work autonomously where you could just set the desired soil moisture to retain and the system can continuously monitor the soil moisture and adjust watering accordingly. You don't really need to tell how many times to switch on the watering system.

Another example could be CCTV cameras that could send low resolution images when there is no movement near them to save network bandwidth but could automatically start to send high resolution images when they sense movement around them.

5. **Several Connectivity Options :** IoT devices typically support various network connectivity options that you could use as per your requirements. Some of the common connectivity options are 2G, 3G, 4G, LTE, 5G, Wi-Fi, Bluetooth, NFC, ZigBee, USB, Ethernet, etc. These devices also support a wide range of network communication protocols such as HTTP, CoAP, MQTT, AMQP, etc. When buying IoT devices, you should determine what connectivity options are available on the devices and make a judicious decision based on your requirements. All devices may not support all the connectivity options and network protocols at the same time. You should carefully choose what you need.

6. **Integrated with Other Systems :** IoT systems often work with other systems taking an integrated approach. IoT systems could provide the desired data that could make rest of the integrated systems work together. For example, you could have an IoT based fire alarm that could automatically use the telephone system to call fire station and other emergency responders, send the building evacuation orders to public announcement system, could automatically call a few ambulances and doctors to handle casualties, and most importantly could activate water sprinkler system to extinguish the fire. IoT devices often work in groups and interact with each other to achieve the desired outcome.

7. **Operates at large scale :** IoT devices operate at large scale. Consider a shopping mall or a factory. The total number of sensors could be in thousands. Overall, an organisation may be controlling, managing, and operating tens of thousands of devices that co-operate and carry out the desired activities.

Also, the total number of IoT devices could easily surpass the human population and any other forms of computing. As per the Gartner report published on Aug 2019 (https://www.gartner.com/en/newsroom/press-releases/2019-08-29-gartner-says-5-8-billion-enterprise-and-automotive-io), approximately 5.8 Billion Enterprise and Automotive IoT Endpoints would be in use by end of 2020. The Table 6.11.1 from the report summarises the IoT endpoint market by segment, 2018-2020, worldwide (installed base, billions of units).

Table 6.11.1

Segment	2018	2019	2020
Utilities	0.98	1.17	1.37
Government	0.40	0.53	0.70
Building Automation	0.23	0.31	0.44
Physical Security	0.83	0.95	1.09
Manufacturing & Natural Resources	0.33	0.40	0.49
Automotive	0.27	0.36	0.47
Healthcare Providers	0.21	0.28	0.36
Retail & Wholesale Trade	0.29	0.36	0.44
Information	0.37	0.37	0.37
Transportation	0.06	0.07	0.08
Total	**3.96**	**4.81**	**5.81**

6.12 IoT Vision

IoT is expected to make a positive difference to multiple sectors – retail, cities, offices, homes, and perhaps everything around you. Let's understand some of the key impacts that IoT could have. Most of this section builds on the report, published by McKinsey & Company, titled "The Internet Of Things: Mapping The Value Beyond The Hype".

1. **Human :** For human-beings, the IoT applications fall into two broad categories – improving health and raising productivity. Unlike other IoT applications, where a reading from a sensor might initiate a specific action, in the human setting, sensor data provide information that people will use to guide their actions and decisions. Using IoT systems to convince healthy people to change their living habits and to help sick patients adhere to doctors' prescriptions would be a true breakthrough. Emerging applications have the potential to transform a wide range of health-care therapies. Implantables, ingestibles and injectables, such as smart pills and nanobots, have the potential eventually to replace many surgeries with less invasive procedures that could offer faster recovery, reduced risk of complications, and lower cost. Imagine an IoT device, placed into a human body, that can continuously monitor blood glucose level and automatically inject insulin dosage appropriately.

2. **Home :** IoT applications relating to the operation of homes, such as energy management, security, and automation of domestic chores are the most useful. By far, the largest opportunity in the home setting is in automating domestic chores. This work is not counted in national productivity data but has an enormous impact on how people spend time and money. The household activities such as cleaning, washing, preparing food, gardening, caring for pets, and so on take a good amount of time and effort.

You could automatically have the home temperature controlled, house cleaned by a robotic vacuum cleaner, refrigerators or shelves automatically ordering groceries based on what is placed on them, clothes washed, dried, and ironed and several other home chores that could be automated.

...using which you can precisely address a specific device and can either read output from the device ...temperature reading) or provide input (to turn off or turn on) to the device.

3. **In-built Intelligence :** As you understand that IoT devices are built on top of embedded systems, these devices are packed with in-built intelligence in the form of specific algorithms, situation-to-action mapping, and other intelligent mechanisms that can sense the environmental conditions and take specific actions or provide the desired response. For example, a smart bulb could automatically detect when you enter and leave the room and power on or off accordingly.

4. **Dynamic Configuration :** IoT devices are self-adapting and self-configuring in nature where they can appropriately and dynamically reconfigure themselves depending upon the operating environment, situations, and conditions. There operation could be automated where you need not provide input controls to get the desired behaviour.

For example, an automatic watering system can water the plants based on the soil moisture. On sunny days, the soil might dry up quickly leading to frequent watering and on a non-sunny day, it might only require watering twice a day. How fast the soil loses moisture depends upon a lot of external conditions and it is hard to predict the precise timing when watering would be required. Such systems can completely work autonomously where you could just set the desired soil moisture to retain and the system can continuously monitor the soil moisture and adjust watering accordingly. You don't really need to tell how many times to switch on the watering system.

Another example could be CCTV cameras that could send low resolution images when there is no movement near them to save network bandwidth but could automatically start to send high resolution images when they sense movement around them.

5. **Several Connectivity Options :** IoT devices typically support various network connectivity options that you could use as per your requirements. Some of the common connectivity options are 2G, 3G, 4G, LTE, 5G, Wi-Fi, Bluetooth, NFC, ZigBee, USB, Ethernet, etc. These devices also support a wide range of network communication protocols such as HTTP, CoAP, MQTT, AMQP, etc. When buying IoT devices, you should determine what connectivity options are available on the devices and make a judicious decision based on your requirements. All devices may not support all the connectivity options and network protocols at the same time. You should carefully choose what you need.

6. **Integrated with Other Systems :** IoT systems often work with other systems taking an integrated approach. IoT systems could provide the desired data that could make rest of the integrated systems work together. For example, you could have an IoT based fire alarm that could automatically use the telephone system to call fire station and other emergency responders, send the building evacuation orders to public announcement system, could automatically call a few ambulances and doctors to handle casualties, and most importantly could activate water sprinkler system to extinguish the fire. IoT devices often work in groups and interact with each other to achieve the desired outcome.

7. **Operates at large scale :** IoT devices operate at large scale. Consider a shopping mall or a factory. The total number of sensors could be in thousands. Overall, an organisation may be controlling, managing, and operating tens of thousands of devices that co-operate and carry out the desired activities.

Also, the total number of IoT devices could easily surpass the human population and any other forms of computing. As per the Gartner report published on Aug 2019 (https://www.gartner.com/en/newsroom/press-releases/2019-08-29-gartner-says-5-8-billion-enterprise-and-automotive-io), approximately 5.8 Billion Enterprise and Automotive IoT Endpoints would be in use by end of 2020. The Table 6.11.1 from the report summarises the IoT endpoint market by segment, 2018-2020, worldwide (installed base, billions of units).

Table 6.11.1

Segment	2018	2019	2020
Utilities	0.98	1.17	1.37
Government	0.40	0.53	0.70
Building Automation	0.23	0.31	0.44
Physical Security	0.83	0.95	1.09
Manufacturing & Natural Resources	0.33	0.40	0.49
Automotive	0.27	0.36	0.47
Healthcare Providers	0.21	0.28	0.36
Retail & Wholesale Trade	0.29	0.36	0.44
Information	0.37	0.37	0.37
Transportation	0.06	0.07	0.08
Total	**3.96**	**4.81**	**5.81**

6.12 IoT Vision

IoT is expected to make a positive difference to multiple sectors – retail, cities, offices, homes, and perhaps everything around you. Let's understand some of the key impacts that IoT could have. Most of this section builds on the report, published by McKinsey & Company, titled "The Internet Of Things: Mapping The Value Beyond The Hype".

1. **Human :** For human-beings, the IoT applications fall into two broad categories – improving health and raising productivity. Unlike other IoT applications, where a reading from a sensor might initiate a specific action, in the human setting, sensor data provide information that people will use to guide their actions and decisions. Using IoT systems to convince healthy people to change their living habits and to help sick patients adhere to doctors' prescriptions would be a true breakthrough. Emerging applications have the potential to transform a wide range of health-care therapies. Implantables, ingestibles and injectables, such as smart pills and nanobots, have the potential eventually to replace many surgeries with less invasive procedures that could offer faster recovery, reduced risk of complications, and lower cost. Imagine an IoT device, placed into a human body, that can continuously monitor blood glucose level and automatically inject insulin dosage appropriately.

2. **Home :** IoT applications relating to the operation of homes, such as energy management, security, and automation of domestic chores are the most useful. By far, the largest opportunity in the home setting is in automating domestic chores. This work is not counted in national productivity data but has an enormous impact on how people spend time and money. The household activities such as cleaning, washing, preparing food, gardening, caring for pets, and so on take a good amount of time and effort.

You could automatically have the home temperature controlled, house cleaned by a robotic vacuum cleaner, refrigerators or shelves automatically ordering groceries based on what is placed on them, clothes washed, dried, and ironed and several other home chores that could be automated.

Various IoT applications in homes

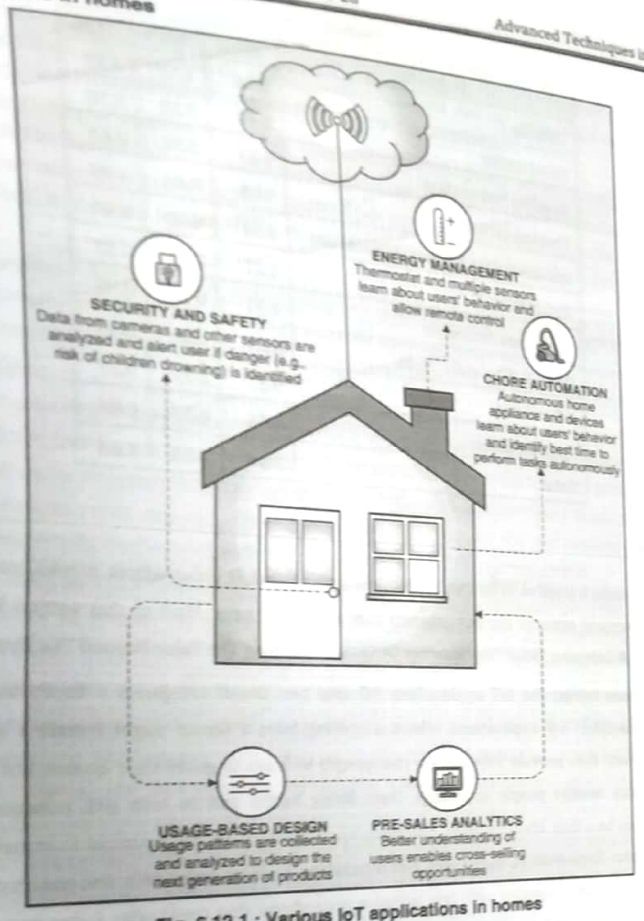


Fig. 6.12.1 : Various IoT applications in homes

3. **Retail :** Retail environments have undergone significant change over the past two decades due to the introduction of information technologies, including the rise of online shopping. The Internet of Things has the potential to cause even greater disruption, but IoT can also provide traditional retailers with the tools to compete and coexist with the online retail world as "omni-channel" shopping erases the distinction between online and offline shops. The Internet of Things, for example, can guide the shopper to the item she has been looking at online when she enters the store and text her a personalised coupon to make the purchase in-store that day. IoT technology can also provide data to optimise store layouts, enable fully automated checkout, and fine-tune inventory management. These and other innovations could enable new business models and allow retailers to improve productivity, reduce costs, and raise sales.

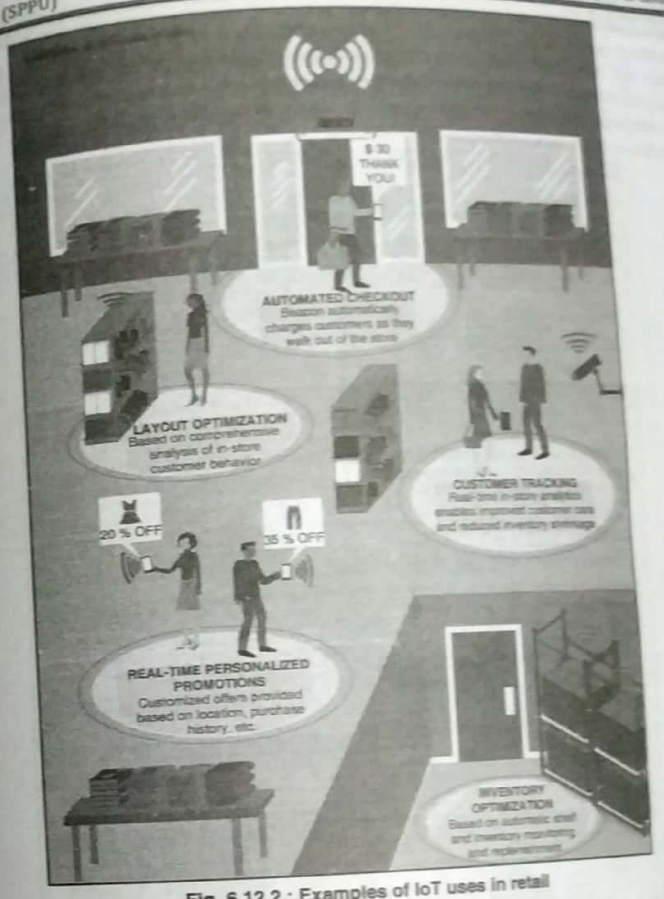


Fig. 6.12.2 : Examples of IoT uses in retail

Amazon Go is an interesting example of how IoT has drastically changed retail shopping experience. You would learn about it later in the chapter.

4. **Offices :** IoT has applications in the office setting that are similar to those in the home setting such as managing energy and security systems. Quite a few office spaces are conserving natural resources by regulating the usage of electricity, water, garbage disposal, etc. and becoming "green" buildings.

5. **Factories :** IoT would play a major role in the next phase of factory automation. It would help in full digitization of production processes, monitor and control all tools of production, and use the data collected to improve productivity and quality. The greatest potential for creating value would be in operations optimisation by making the various processes within the factory more efficient. This includes using sensors, rather than human judgment (and human error), to adjust the performance of machinery. It also involves use of data from production machinery to adjust workflows. This is done by remotely tracking, monitoring, and adjusting machinery based on sensor data from different parts of the plant (and even across plants).

most valuable applications of IoT in the factory setting are predictive maintenance. Predictive maintenance involves using sensors to monitor machinery breakdowns and determine then maintenance will be required, rather than relying on regularly maintenance routines. IoT can improve inventory management by automatically restocking parts based on weight or height data recorded by sensors.

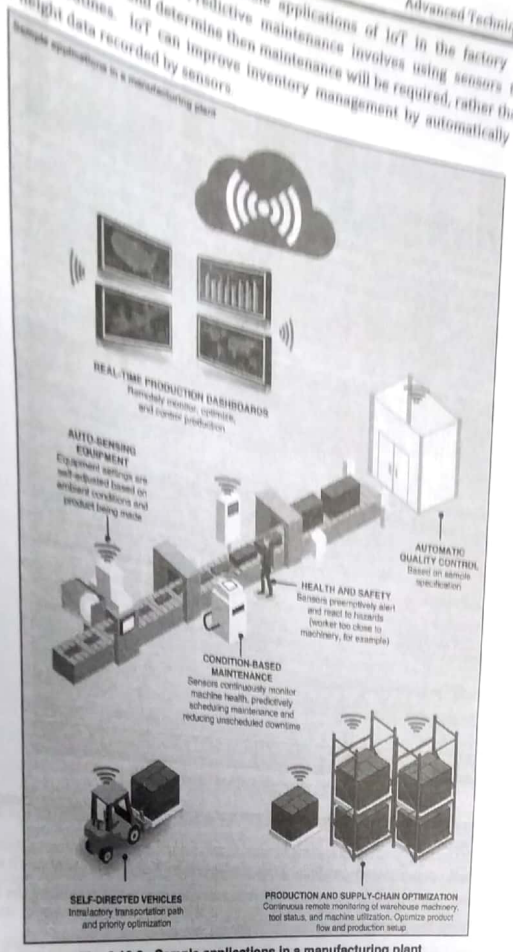


Fig. 6.12.3 : Sample applications in a manufacturing plant

6. **Vehicles :** IoT would play a major role in how vehicles such as cars, trains, trucks, even aircraft are used. The IoT sensors and connectivity could improve how vehicles are serviced, maintained, and designed. The same types of sensors and wireless connections that make it possible to create a self-driving car can be used to monitor how vehicles are performing to enable condition-based maintenance routines that are far more cost-effective than periodic maintenance or performing maintenance after a problem occurs. Tracking performance data can also help vehicle manufacturers design more reliable products and discover other ways to serve customers. Insurance companies could monitor your driving habits and patterns and could provide insurance quotation specifically for you instead of charging you fixed premiums.

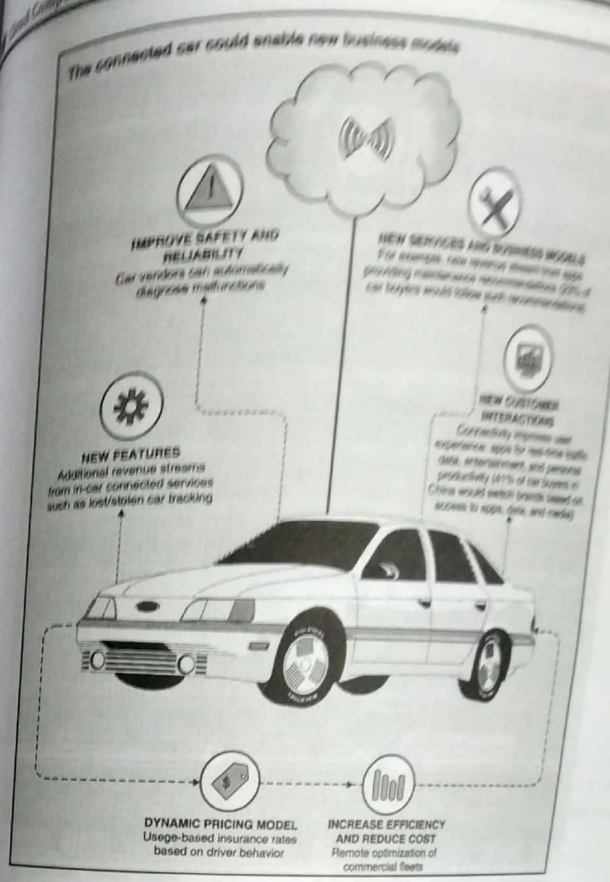


Fig. 6.12.4 : Connected Car

7. **Cities :** IoT has kicked off several "smart city" initiatives around the world. These cities are experimenting with IoT applications to improve services, relieve traffic congestion, conserve water and energy, and improve quality of life. Large, concentrated populations and complex infrastructure make cities a target-rich environment for IoT applications. IoT applications in cities are focusing on public health and safety, transportation, and resource management. IoT applications in public safety and health include air and water quality monitoring.

Transportation applications range from traffic-control systems to smart parking meters to autonomous vehicles. Resource and infrastructure management uses include sensors and smart meters to better manage water and electric infrastructure.

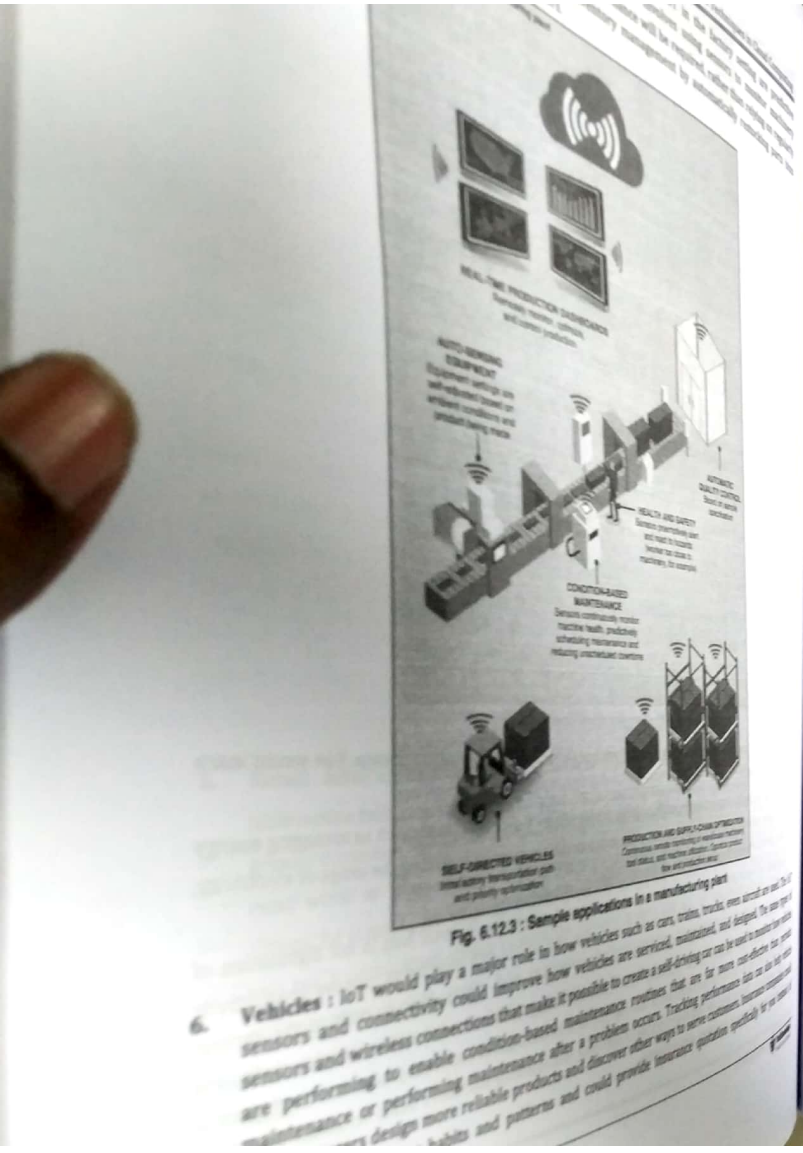


Fig. 6.12.3 : Sample applications in a manufacturing plant

6. **Vehicles :** IoT would play a major role in how vehicles such as cars, trains, trucks, even aircraft and ships are serviced, maintained, and designed. The use of sensors and connectivity could improve how vehicles are serviced, maintained, and designed. The use of sensors and wireless connections that make it possible to create a self-driving car can be used to monitor the car while it is performing to enable condition-based maintenance routines that are far more cost-effective than maintenance or performing maintenance after a problem occurs. Tracking performance data on an ongoing basis can help designers design more reliable products and discover other ways to serve customers. Insurance companies can track driving habits and patterns and could provide insurance quotation specifically to an individual driver.

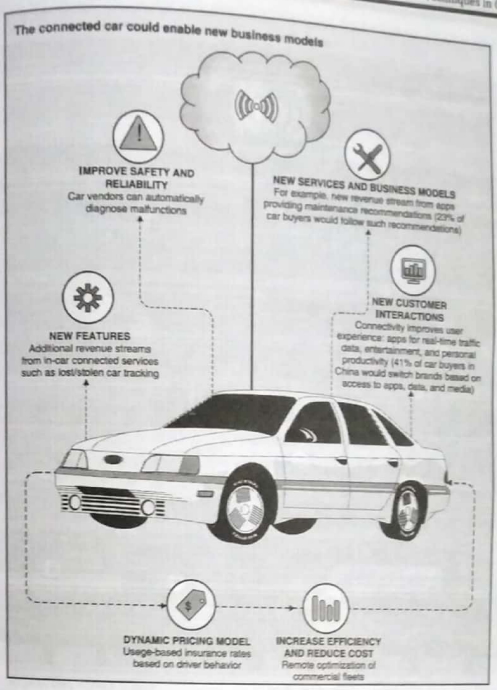


Fig. 6.12.4 : Connected Car

7. **Cities :** IoT has kicked off several "smart city" initiatives around the world. These cities are experimenting with IoT applications to improve services, relieve traffic congestion, conserve water and energy, and improve quality of life. Large, concentrated populations and complex infrastructure make cities a target-rich environment for IoT applications. IoT applications in cities are focusing on public health and safety, transportation, and resource management. IoT applications in public safety and health include air and water quality monitoring.

Transportation applications range from traffic-control systems to smart parking meters to autonomous vehicles. Resource and infrastructure management uses include sensors and smart meters to better manage water and electric infrastructure.

Fig. 6.12.5 : IoT applications in cities

To summarise, IoT applications could provide significant value in the areas which are shown in Fig. 6.12.6.

Internet of Things applications create value in nine settings

Setting	Description	Use-case
Cities	Urban environments	Public safety and health, traffic control, resource management
Offices	Employment locations for knowledge workers	Organizational modeling and worker monitoring, augmented reality for training
Retail	Locations where consumers engage in commerce	Self-checkout, layout optimization, smart customer relationship management
Work sites	Custom production environments	Operations management, equipment maintenance, health and safety
Factories	Standardized production environments	Operations management, predictive maintenance
Outside	Between urban environments (and outside other settings)	Logistics routing, autonomous (self-driving) vehicles, navigation
Home	Building where people live	Energy management, safety and security, chore automation
Human	Devices attached to or inside human body	Monitoring and managing illness, improving wellness
Automobile	Systems inside moving vehicles	Condition-based maintenance, determining insurance premiums

Fig. 6.12.6 : Major IoT Application Areas

6.13 Economic Significance of IoT (Impact of IoT)

As per the McKinsey report stated earlier, The IoT market has the potential to generate about $4 trillion to $11 trillion in economic value by 2025. The Fig. 6.13.1 shows the various areas of potential for IoT with predictions for low and high impact.

Fig. 6.13.1

6.14 Technical Building Blocks (High-Level Architecture of IoT)

The Fig. 6.14.1 shows block diagram provides a high-level architectural view of IoT.

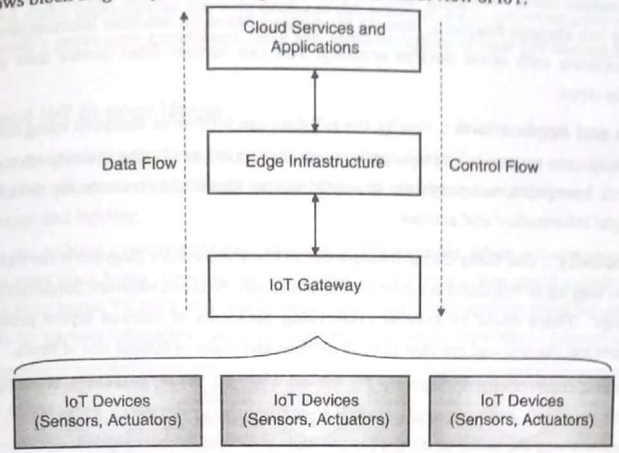


Fig. 6.14.1 : High-level architecture of IoT

IoT architecture consists of four major components as shown in the architecture Fig. 6.14.1.

1. **IoT Devices :** These are the IoT devices such as thermostats, cameras, watch, refrigerator, cars, etc. These devices continuously record and track various data as per their design. For example, your smart watch continuously tracks your pulse. These devices have various connectivity options such as ZigBee, Bluetooth, and Wi-Fi.

   IoT devices typically have

   (a) Sensors : These are electronic circuits that continuously read the data as designed. These could be RFID, GPS, Accelerometers, Gyroscope, etc.

   (b) Actuators : These are electronic circuits that can not only read data but also take actions. For example, if the room temperature goes beyond a certain limit, an actuator can automatically power on or power off air conditioning system.

   You have already learnt about how sensors and actuators work in detail in the "Architecture of a Real-time System" section previously.

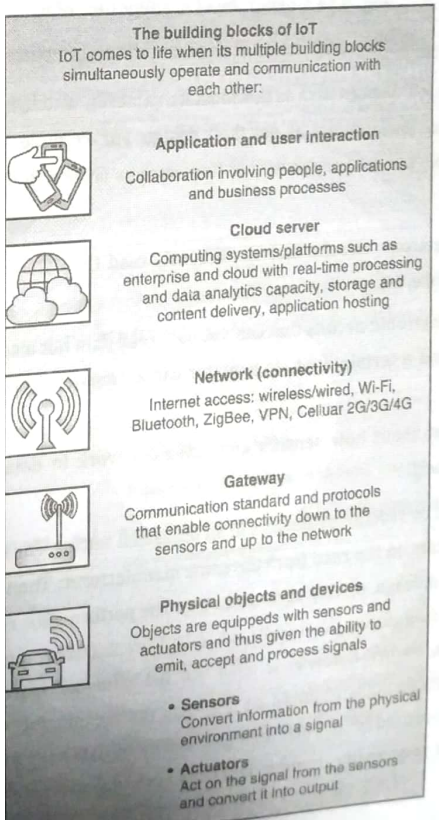
2. **IoT Gateway :** There could be several IoT devices in an industrial setting. For example, if you are using a connected car, there could be several cars on the road from the same manufacturer. The various data points from the connected cars such as engine health, mileage, driving style and other car performance related parameters could be tracked by the manufacturer for providing preventive maintenance and breakdown support. To optimise the number of direct connections to the devices, an IoT Gateway is used. An IoT gateway aggregates data from several devices and processes them. The pre-processing is required so that only the meaningful information is sent further. IoT devices could collect data at a milli-second level. Such high granularity of data may not be required to be sent further in its entirety. IoT gateways, thus, process the granular data from several devices and send them further.

e IoT data is synthesised at the IoT gateway, it is sent to edge infrastructure. Edge
omputing resources closer to the user to avoid sending large volume of data over
functions, execute predictions based on machine learning models, keep device dat
other devices securely. You can further filter device data and only send necess

tions : Finally, the IoT data can further be analysed using cloud services and
al IoT based applications such as car predictive maintenance, health monitoring, tra
nation system or anything else that could consume the data from the IoT devices
n and actions.

thing that is transparent in the architecture diagram is the network connectivity fro
e cloud services and applications. Without network connectivity, there is no interne
uld be several networking protocols at various layers providing seamless netw
and manage the IoT devices and make optimum use of them.

ity options are 2G, 3G, 4G, LTE, 5G, Wi-Fi, Bluetooth, NFC, ZigBee, USB, Ethernet,
ide range of network communication protocols such as HTTP, CoAP, MQTT, AMQP,
building blocks (or the high-level architecture) of IoT.



6.15 IoT and Cloud Convergence

So, as you understand, cloud technologies are very useful in the world of IoT to collect massive amount of sensor data and process them to provide a great user experience. Let's see some examples of how IoT backed by cloud computing is changing our lives.

6.15.1 The Cloud and IoT in your Home

- There are several IoT devices available in the market today that help you to automate mundane tasks at home. There are various categories of products under home automation such as assistance, safety and security, entertainment, connectivity, and energy and lighting.
- These products help you to have a connected home experience. For example, while you are away from your home, you could see who came near your home door via the connected CCTV camera. You could automatically play the video from your mobile to your home TV once you reach home. Following are snapshots from Google Nest range of home automation products for various categories. You could also visit the Google store site to explore more.

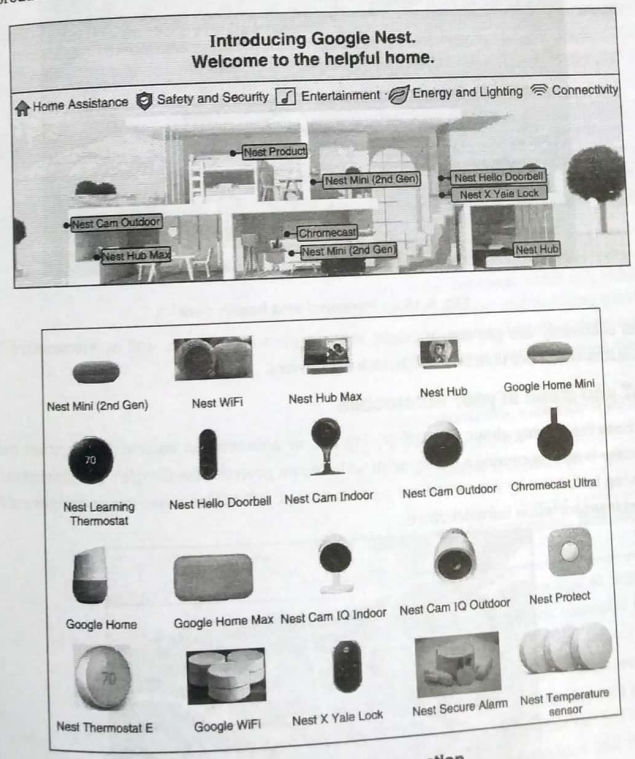


Fig. 6.15.1 : Home Automation

- ...re are quite a few IoT devices into personal and health care segment. These devices track and report every part of your day including activity, exercise, food, weight and sleep to help you stay fit, stay motivated, and make an overall healthy impact on your lifestyle.
- For example, Fit bit provides several of these devices that you can just make them part of your regular life. These could be smart watches, trackers, and forms of wearables such as shoes and T-shirts.

Fig. 6.15.2 : Personal and health care - 1

- These devices commonly use gas sensors, light sensors, pressure sensors, and accelerometers. Various companies make these sensors which can then be used in such IoT devices.

6.15.3 The IoT and cloud in your Automobile

- People have been fantasising about the self-driving car, or autonomous vehicle, in literature and film for decades. While this fantasy is now becoming a reality with well-known projects like Google's self-driving car, Tesla and several other self-driving initiatives by renowned car manufacturers, IoT is also a necessary component for implementing a fully connected transportation infrastructure.

Fig. 6.15.3

- IoT is going to allow self-driving vehicles to better interact with the transportation system around them through bidirectional data exchanges while also providing important data to the riders. Self-driving vehicles need always-on, reliable communications and data from other transportation-related sensors to reach their full potential. Connected roadways is the term associated with both the driver and driverless cars fully integrating with the surrounding transportation infrastructure. Basic sensors reside in cars already. They monitor oil pressure, tyre pressure, temperature, and other operating conditions, and provide data around the core car functions. From behind the steering wheel, the driver can access this data while also controlling the car using equipment such as a steering wheel, pedals, and so on. The need for all this sensory information and control is obvious. The driver must be able to understand, handle, and make critical decisions while concentrating on driving safely. The Internet of Things is replicating this concept on a much larger scale.
- Today, you are seeing automobiles produced with thousands of sensors, to measure everything from fuel consumption to location to the entertainment your family is watching during the ride. As automobile manufacturers strive to reinvent the driving experience, these sensors are becoming IP-enabled to allow easy communication with other systems both inside and outside the car. In addition, new sensors and communication technologies are being developed to allow vehicles to "talk" to other vehicles, traffic signals, school zones, and other elements of the transportation infrastructure. We are now starting to realize a truly connected transportation solution. Most connected roadways solutions focus on resolving today's transportation challenges. These challenges can be classified into the three categories highlighted in the Table 6.15.1.

Table 6.15.1 : Transportation challenges

Challenge	Supporting data
Safety	According to the US department of Transportation. 5.6 million crashes were reported in 2012 alone resulting in more than 33,000 fatalities IoT and the enablement of connected vehicle technologies will empower drivers with the tools they need to anticipate potential crashes and significantly reduce the number of lives lost each year.
Mobility	More than a billion cars are on the roads worldwide. Connected vehicle mobility applications can enable system operators and drivers to make more informed decisions, which can, in turn, reduce travel delays. Congestion causes 5.5 billion hours of travel delay per year, and reducing travel delays is more critical than ever before. In addition, communication between mass transit, emergency response vehicles, and traffic management infrastructures help optimize the routing of vehicles, further reducing potential delays.
Environment	According to the American Public Transportation Association, each year transit systems can collectively reduce carbon diode (CO_2) emissions by 16.2 million metric tons by reducing private vehicle miles. Connected vehicle environmental applications will give all travellers the real time information they need to make "green" transportation choices.

Sources : Traffic Safety Facts, 2010 : National Highway Traffic Safety Administration, June 2012; and WHO Global Status Report on Road Safety; 2013.

- By addressing these challenges, connected roadways will bring many benefits to society. These benefits include reduced traffic jams and urban congestion, decreased casualties and fatalities, increased response time for emergency vehicles, and reduced vehicle emissions.
- For example, with IoT-connected roadways, a concept known as Intersection Movement Assist (IMA) is possible. This application warns a driver (or triggers the appropriate response in a self-driving car) when it is not safe to enter an intersection due to a high probability of a collision - perhaps because another car has run a stop sign or strayed into the wrong lane. Thanks to the communications system between the vehicles and the infrastructure, this sort of scenario can be handled quickly and safely. The Fig. 6.15.4 illustrates IMA.

6.15.2 Personal : IoT in Healthcare

- There are quite a few IoT devices into personal and health care segment. These devices track and report every part of your day including activity, exercise, food, weight and sleep to help you stay fit, stay motivated, and make an overall healthy impact on your lifestyle.
- For example, Fit bit provides several of these devices that you can just make them part of your regular life. These could be smart watches, trackers, and forms of wearables such as shoes and T-shirts.

Fig. 6.15.2 : Personal and health care - 1

- These devices commonly use gas sensors, light sensors, pressure sensors, and accelerometers. Various companies make these sensors which can then be used in such IoT devices.

6.15.3 The IoT and cloud in your Automobile

- People have been fantasising about the self-driving car, or autonomous vehicle, in literature and film for decades. While this fantasy is now becoming a reality with well-known projects like Google's self-driving car, Tesla and several other self-driving initiatives by renowned car manufacturers, IoT is also a necessary component for implementing a fully connected transportation infrastructure.

Fig. 6.15.3

- IoT is going to allow self-driving vehicles to better interact with the transportation system around them through bidirectional data exchanges while also providing important data to the riders. Self-driving vehicles need always-on, reliable communications and data from other transportation-related sensors to reach their full potential. Connected roadways is the term associated with both the driver and driverless cars fully integrating with the surrounding transportation infrastructure. Basic sensors reside in cars already. They monitor oil pressure, tyre pressure, temperature, and other operating conditions, and provide data around the core car functions. From behind the steering wheel, the driver can access this data while also controlling the car using equipment such as a steering wheel, pedals, and so on. The need for all this sensory information and control is obvious. The driver must be able to understand, handle, and make critical decisions while concentrating on driving safely. The Internet of Things is replicating this concept on a much larger scale.
- Today, you are seeing automobiles produced with thousands of sensors, to measure everything from fuel consumption to location to the entertainment your family is watching during the ride. As automobile manufacturers strive to reinvent the driving experience, these sensors are becoming IP-enabled to allow easy communication with other systems both inside and outside the car. In addition, new sensors and communication technologies are being developed to allow vehicles to "talk" to other vehicles, traffic signals, school zones, and other elements of the transportation infrastructure. We are now starting to realize a truly connected transportation solution. Most connected roadways solutions focus on resolving today's transportation challenges. These challenges can be classified into the three categories highlighted in the Table 6.15.1.

Table 6.15.1 : Transportation challenges

Challenge	Supporting data
Safety	According to the US department of Transportation. 5.6 million crashes were reported in 2012 alone resulting in more than 33,000 fatalities IoT and the enablement of connected vehicle technologies will empower drivers with the tools they need to anticipate potential crashes and significantly reduce the number of lives lost each year.
Mobility	More than a billion cars are on the roads worldwide. Connected vehicle mobility applications can enable system operators and drivers to make more informed decisions, which can, in turn, reduce travel delays. Congestion causes 5.5 billion hours of travel delay per year, and reducing travel delays is more critical than ever before. In addition, communication between mass transit, emergency response vehicles, and traffic management infrastructures help optimize the routing of vehicles, further reducing potential delays.
Environment	According to the American Public Transportation Association, each year transit systems can collectively reduce carbon diode (CO_2) emissions by 16.2 million metric tons by reducing private vehicle miles. Connected vehicle environmental applications will give all travellers the real time information they need to make "green" transportation choices.

Sources : *Traffic Safety Facts, 2010 : National Highway Traffic Safety Administration, June 2012; and WHO Global Status Report on Road Safety; 2013.*

- By addressing these challenges, connected roadways will bring many benefits to society. These benefits include reduced traffic jams and urban congestion, decreased casualties and fatalities, increased response time for emergency vehicles, and reduced vehicle emissions.
- For example, with IoT-connected roadways, a concept known as Intersection Movement Assist (IMA) is possible. This application warns a driver (or triggers the appropriate response in a self-driving car) when it is not safe to enter an intersection due to a high probability of a collision - perhaps because another car has run a stop sign or strayed into the wrong lane. Thanks to the communications system between the vehicles and the infrastructure, this sort of scenario can be handled quickly and safely. The Fig. 6.15.4 illustrates IMA.

Fig. 6.15.4

- **IMA is one of many possible roadway solutions that emerge when we start to integrate IoT with both traditional and self-driving vehicles. Other solutions include automated vehicle tracking, cargo management, and road weather communications.**
- **With automated vehicle tracking, a vehicle's location is used for notification of arrival times, theft prevention, or highway assistance. Cargo management provides precise positioning of cargo as it is enroute so that notification alerts can be sent to a dispatcher and routes can be optimized for congestion and weather. Road weather communications use sensors and data from satellites, roads, and bridges to warn vehicles of dangerous conditions or inclement weather on the current route.**
- **Today's typical road car utilizes more than a million lines of code and this only scratches the surface of the data potential. As cars continue to become more connected and capable of generating continuous data streams related to location, performance, driver behaviour, and much more, the data generation potential of a single car is staggering. It is estimated that a fully connected car will generate more than 25 gigabytes of data per hour, much of which will be sent to the cloud.**
- **To put this in perspective, that's equivalent to a dozen HD movies sent to the cloud every hour—by your car! Multiply that by the number of hours a car is driven per year and again by the number of cars on the road, and you see that the amount of connected car data generated, transmitted, and stored in the cloud will be in the zettabytes range per year (more than a billion petabytes per year). The Fig. 6.15.5 provides an overview of the sort of sensors and connectivity that you will find in a connected car.**

Fig. 6.15.5

- **Cloud computing roadways are undergoing massive disruption as in how the data generated by a car will be used by third parties. Clearly, the data generated by your car needs to be handled in a secure and reliable way. It needs to be highly available. But who will use all this data? Automobile data is extremely useful to a wide range of interested parties. For example, tyre companies can collect data related to use and durability of their products in a range of environments in real time. Automobile manufacturers can collect information from motors to better understand how the cars are being driven, when parts are starting to fail, or whether the car has been involved in a collision. All that will help them build better cars in the future. This becomes especially true as autonomous vehicles are introduced, which are sure to be driven in a completely different way than the traditional family car.**
- **In the future, car sensors will be able to interact with third-party applications, such as GPS/maps, to enable dynamic routing to avoid traffic, accidents, and other hazards. Similarly, Internet-based entertainment, including music, movies, and other streaming or downloads, can be personalised and customised to optimize a road trip. This data will also be used for targeted advertising. As GPS navigation systems become more integrated with sensors and way finding applications, it will become possible for personalized routing suggestions to be made. For example, if it is known that you prefer a certain coffee shop, through the use of a cloud-based data connector, the navigation system will be able to provide routing suggestions that have you drive your car past the right coffee shop.**
- **All these data opportunities bring into play a new technology: the IoT data broker. Imagine the many different types of data generated by an automobile and the plethora of different parties interested in this data. This poses a significant business opportunity. In a very real sense, the data generated by the car and driver becomes a valuable commodity that can be bought and sold. While the data transmitted from the car will likely go to one initial location in the cloud, from there the data can be separated and sold selectively by the data broker.**
- **For example, tire companies will pay for information from sensors related to your tires, but they won't get anything else. While information brokers have been around a long time, the technology used to aggregate and separate the data from connected cars in a secure and governed manner is rapidly developing and will continue to be a major focus of the IoT industry for years to come.**
- **Connected roadways are likely to be one of the biggest growth areas for innovation. Automobiles and the roads they use have seen incredible change over the past century, but the changes ahead of us are going to be just as astounding. In the past few years alone, we have seen highway systems around the world adopt sophisticated sensors that can detect seismic vibrations, car accidents, severe weather conditions, traffic congestion, and more. Recent advancements in roadway fibre-optic sensing technology is now able to record not only how many cars are passing but their speed and type.**

Review Questions

Here are a few review questions to help you gauge your understanding of this chapter. Try to attempt these questions and ensure that you can recall the points mentioned in the chapter.

Q.1 Why is IPTV better than traditional TV? (4 Marks)

Q.2 Write a short note on cloud mashups. (3 Marks)

Q.3 Describe cloud mashup. List its advantages and disadvantages. (4 Marks)

Q.4 Write a short note on Mobile Cloud Computing. (4 Marks)

Q.5 Write a short note on cloudlet. (4 Marks)

Q.6 Differentiate between cloud and cloudlet. (4 Marks)