ELECTIVE

5.4.1 CLOUD COMPUTING **L T P 3 - 3**

RATIONALE

This course offers a good understanding of cloud computing concepts and challenges faced in implementation of cloud computing.

LEARNING OUTCOMES

After undergoing the subject, the students would be able to:

- Explain core concepts of cloud computing paradigm.
- Explain various Service Models
- Explain various Deployment Models.
- Describe SLA management in Cloud Computing
- Explain and apply the concept of virtualization.
- Describe the scheduling of tasks in cloud.
- Illustrate the fundamental concepts of cloud storage.
- Describe various security issues in the cloud.
- Make use of cloud.

DETAILED CONTENTS

1. Introduction (6 Periods)

Evolution of Cloud Computing, Cloud Computing Overview, Characteristics, Applications, Benefits,

Challenges.

2. Service and Deployment Models (6 Periods)

- Cloud Computing Service Models: Infrastructure as a Service, Platform as a Service, Software as a Service;
- Cloud Computing Deployment Models: Private Cloud; Public Cloud, Community Cloud, Hybrid Cloud, Major Cloud Service providers.

3. Service Level Agreement (SLA) Management (4 Periods)

Overview of SLA, Types of SLA, SLA Life Cycle, SLA Management Process.

4. Virtualization Concepts (8 Periods)

Overview of Virtualization, Types of Virtualization, Benefits of Virtualization, Hypervisors.

5. Cloud Security (6 Periods)

Infrastructure Security, Data Security & Privacy Issues, Legal Issues in Cloud Computing.

6. Cloud Storage (6 Periods)

Overview; Storage as a Service, Benefits and Challenges, Storage Area Networks (SANs).

7. Scheduling in Cloud (12 Periods)

Overview of Scheduling problem, Different types of scheduling, Scheduling for independent and dependent tasks, Static vs. Dynamic scheduling.

LIST OF PRACTICALS

1. Introduction to Cloud Vendors: Amazon, Microsoft, IBM.

2. Setting up Virtualization using Virtualbox/VMWare Hypervisor

3. Introduction to OwnCloud

4. Installation and configuration of OwnCloud software for SaaS

5. Accessing Microsoft AZURE cloud-services

6. Cloud Simulation Software Introduction: CloudSim

INSTRUCTIONAL STRATEGY

In addition to cassroom teaching, the teacher should demonstrate the practical usage of cloud using real cloud services.

MEANS OF ASSESSMENT

- Assignments and Quiz/class tests, mid-term and end-term written tests
- Actual laboratory and practical work and Viva-Voce

RECOMMENDED BOOKS

1. Rajkumar Buyya, James Broberg, Andrzej Goscinski (Editors): Cloud Computing: Principles and Paradigms, Wiley, 2011

2. Kumar Saurabh, Cloud Computing, Wiley, 2012.

3. Barrie Sosinsky: Cloud Computing Bible, Wiley, 2011.

4. Judith Hurwitz, Robin Bloor, Marcia Kaufman,Fern Halper: Cloud Computing for Dummies, Wiley, 2010

5. e-books/e-tools/relevant software to be used as recommended by AICTE/HSBTE/NITTTR.

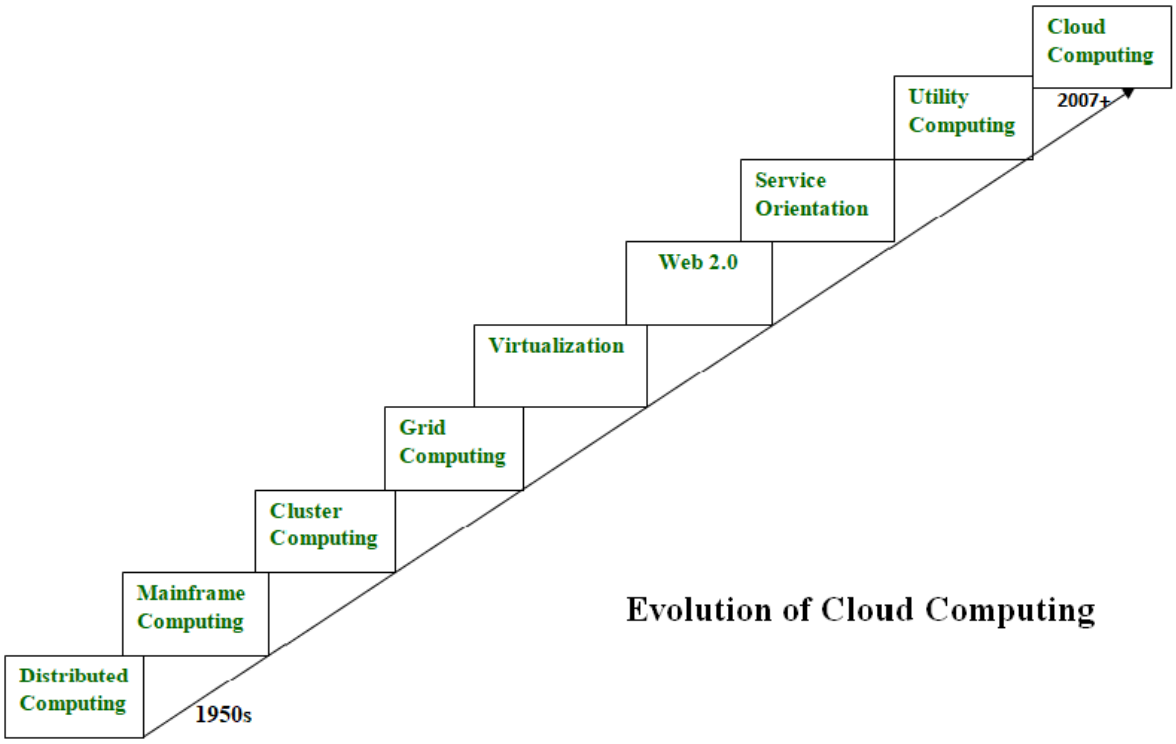
Websites for Reference:

http://swayam.gov.in

Evolution of Cloud Computing

Cloud computing is all about renting computing services. This idea first came in the 1950s. In making cloud computing what it is today, five technologies played a vital role. These are **distributed systems and its peripherals, virtualization, web 2.0, service orientation, and utility computing**.

I. **Distributed Systems and its peripherals**
II. **Virtualization**
III. **Web 2.0**
IV. **Service Orientation**
V. **Utility Computing**



Evolution of Cloud Computing

1) **Distributed Systems:**
   It is a composition of multiple independent systems but all of them are depicted as a single entity to the users. The purpose of distributed systems is to share resources and also use them effectively and efficiently. Distributed systems possess characteristics such as scalability, concurrency, continuous availability, heterogeneity, and independence in failures. But the main problem with this system was that all the systems were required to be present at the same geographical location. Thus to solve this problem, distributed

computing led to three more types of computing and they were- Mainframe computing, cluster computing, and grid computing.

2) **Mainframe computing:**
   Mainframes which first came into existence in 1951 are highly powerful and reliable computing machines. These are responsible for handling large data such as massive input-output operations. Even today these are used for bulk processing tasks such as online transactions etc. These systems have almost no downtime with high fault tolerance. After distributed computing, these increased the processing capabilities of the system. But these were very expensive. To reduce this cost, cluster computing came as an alternative to mainframe technology.
3) **Cluster computing:**
   In 1980s, cluster computing came as an alternative to mainframe computing. Each machine in the cluster was connected to each other by a network with high bandwidth. These were way cheaper than those mainframe systems. These were equally capable of high computations. Also, new nodes could easily be added to the cluster if it was required. Thus, the problem of the cost was solved to some extent but the problem related to geographical restrictions still pertained. To solve this, the concept of grid computing was introduced.

4) **Grid computing:**
   In 1990s, the concept of grid computing was introduced. It means that different systems were placed at entirely different geographical locations and these all were connected via the internet. These systems belonged to different organizations and thus the grid consisted of heterogeneous nodes. Although it solved some problems but new problems emerged as the distance between the nodes increased. The main problem which was encountered was the low availability of high bandwidth connectivity and with it other network associated issues. Thus cloud computing is often referred to as “Successor of grid computing”.

5) **Virtualization:**
   It was introduced nearly 40 years back. It refers to the process of creating a virtual layer over the hardware which allows the user to run multiple instances simultaneously on the hardware. It is a key technology used in **cloud computing**. It is the base on which major cloud computing services such as **Amazon EC2**, **VMware vCloud**, etc work on. Hardware virtualization is still one of the most common types of virtualization.

6) **Web 2.0:**
   It is the interface through which the cloud computing services interact with the clients. It is because of Web 2.0 that we have interactive and dynamic web pages. It also increases flexibility among web pages. Popular examples of web 2.0 include Google Maps, Facebook, Twitter, etc. Needless to say, social media is possible because of this technology only. In gained major popularity in 2004.

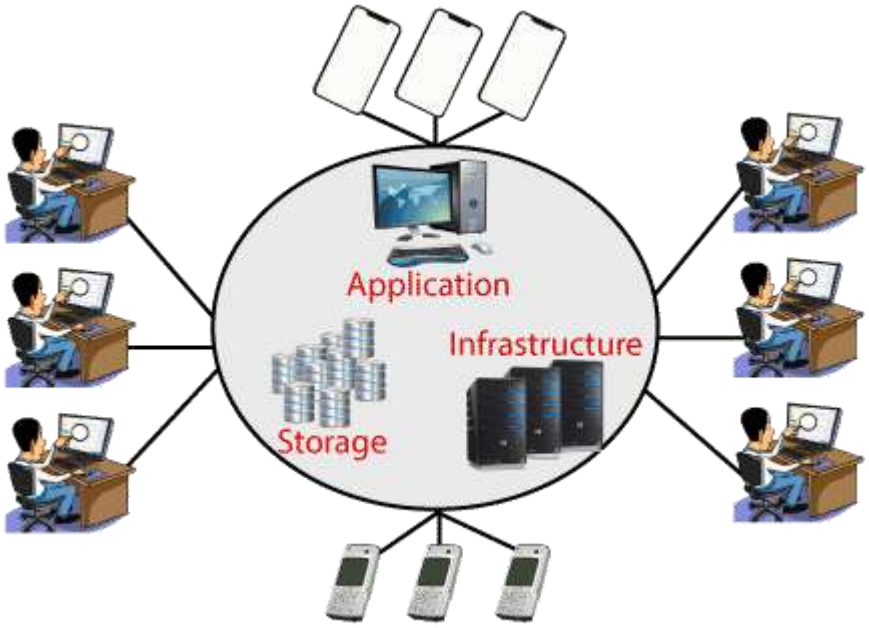
7) **Service orientation:**
   It acts as a reference model for cloud computing. It supports low-cost, flexible, and evolvable applications. Two important concepts were introduced in this computing model. These were **Quality of Service (QoS)** which also includes the **SLA (Service Level Agreement)** and **Software as a Service (SaaS)**.

8) **Utility computing:**
   It is a computing model that defines service provisioning techniques for services such as compute services along with other major services such as **storage**, **infrastructure**, etc which are provisioned on a **pay-per-use basis**.

Thus, the above technologies contributed to the making of cloud computing.

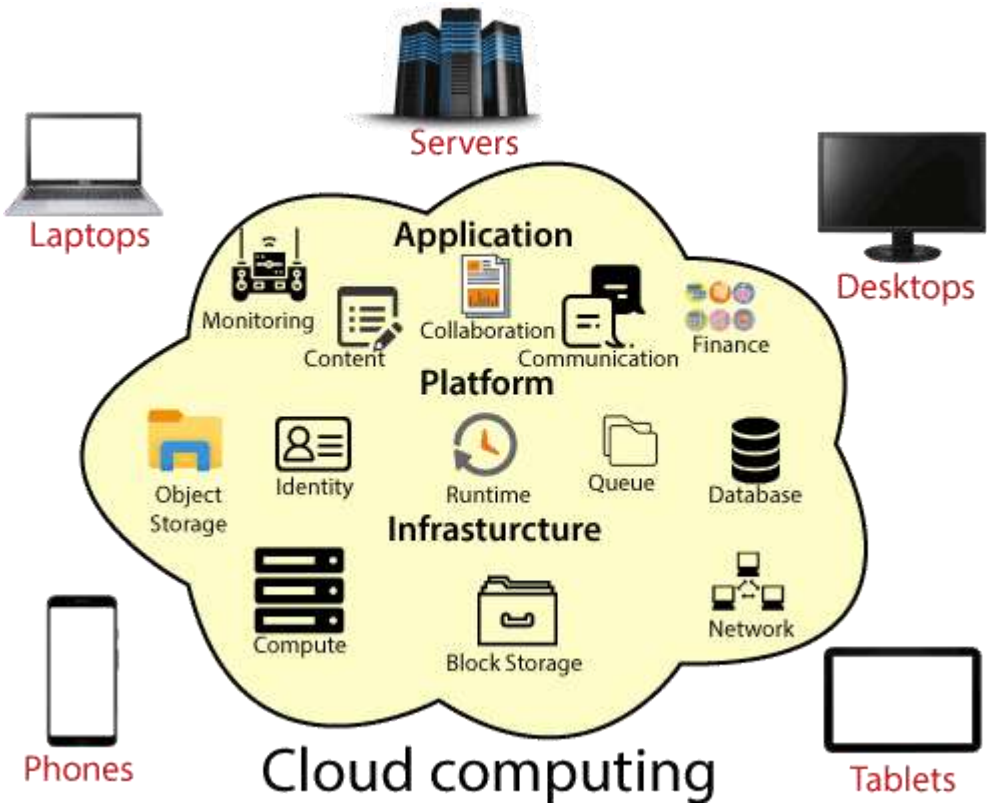
Introduction to Cloud Computing

Cloud Computing is the delivery of computing services such as servers, storage, databases, networking, software, analytics, intelligence, and more, over the Cloud (Internet).



Cloud Computing provides an alternative to the on-premises datacentre. With an on-premises datacentre, we have to manage everything, such as purchasing and installing hardware, virtualization, installing the operating system, and any other required applications, setting up the network, configuring the firewall, and setting up storage for data. After doing all the set-up, we become responsible for maintaining it through its entire lifecycle.

But if we choose Cloud Computing, a cloud vendor is responsible for the hardware purchase and maintenance. They also provide a wide variety of software and platform as a service. We can take any required services on rent. The cloud computing services will be charged based on usage.



The cloud environment provides an easily accessible online portal that makes handy for the user to manage the compute, storage, network, and application resources. Some cloud service providers are in the following figure.

Main Characteristics of Cloud Computing

The five major characteristics of the Cloud Computing are as follows:

1. **On-demand self-service** - The service of the cloud is available round the clock and provides computing capabilities on-demand of the user automatically.
2. **Broad network access** - Users can access the services via different modes as the heterogeneous thin and thick client platforms.
3. **Resource pooling** - The feature of multi-tenancy where users are assigned resources dynamically, based on demands.
4. **Rapid elasticity**– The service is flexible and can be scaled up or down to suit the business requirements. Resources and programs can be used based on requirement and the user is billed only for the usage.
5. **Measured service** – Usage metering is available and you pay only for what you use. You need not pay for any infrastructure that you do not use.

Advantages of cloud computing

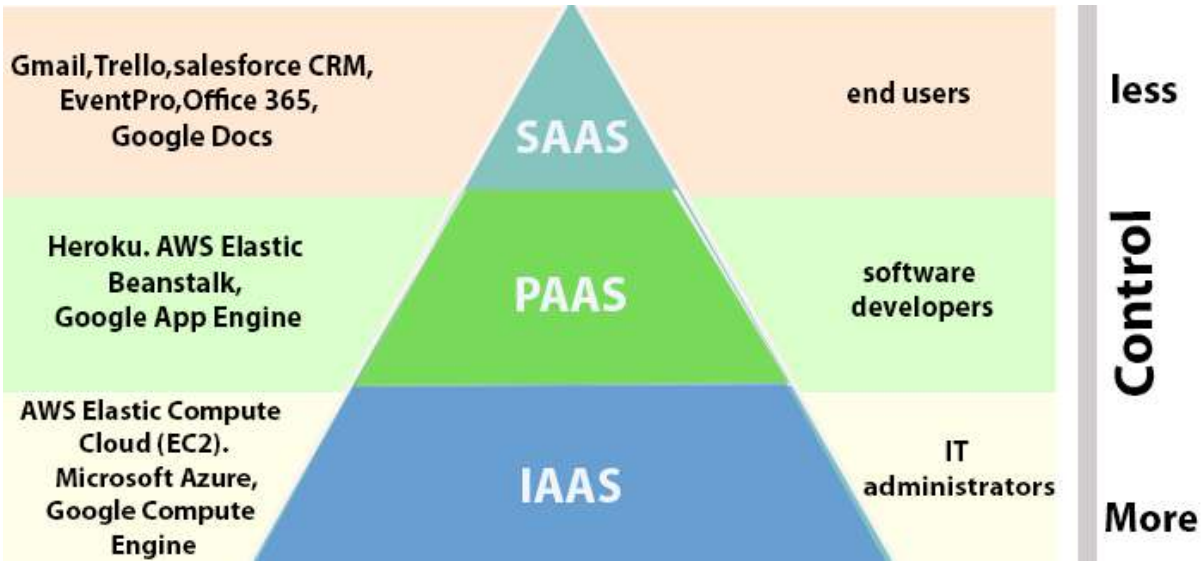
- **Cost:** It reduces the huge capital costs of buying hardware and software.
- **Speed:** Resources can be accessed in minutes, typically within a few clicks.
- **Scalability:** We can increase or decrease the requirement of resources according to the business requirements.
- **Productivity:** While using cloud computing, we put less operational effort. We do not need to apply patching, as well as no need to maintain hardware and software. So, in this way, the IT team can be more productive and focus on achieving business goals.
- **Reliability:** Backup and recovery of data are less expensive and very fast for business continuity.
- **Security:** Many cloud vendors offer a broad set of policies, technologies, and controls that strengthen our data security.

Unit 2: Cloud Computing Service Models:

Various Cloud Computing Service model are as:

1) **Infrastructure as a Service (IaaS)**
2) **Platform as a Service (PaaS)**
3) **Software as a Service(SaaS)**

Types of Cloud Services



1. **Infrastructure as a Service (IaaS):** In IaaS, we can rent IT infrastructures like servers and virtual machines (VMs), storage, networks, operating systems from a cloud service vendor. We can create VM running Windows or Linux and install anything we want on it. Using IaaS, we don't need to care about the hardware or virtualization software, but other than that, we do have to manage everything else. Using IaaS, we

get maximum flexibility, but still, we need to put more effort into maintenance.

2. **Platform as a Service (PaaS):** This service provides an on-demand environment for developing, testing, delivering, and managing software applications. The developer is responsible for the application, and the PaaS vendor provides the ability to deploy and run it. Using PaaS, the flexibility gets reduce, but the management of the environment is taken care of by the cloud vendors.
3. **Software as a Service (SaaS):** It provides a centrally hosted and managed software services to the end-users. It delivers software over the internet, on-demand, and typically on a subscription basis. E.g., Microsoft One Drive, Dropbox, WordPress, Office 365, and Amazon Kindle. SaaS is used to minimize the operational cost to the maximum extent.

Cloud Computing Deployment Models

The cloud services can be deployed in different methods. The deployment model is based on the service model, organizational structure, location, user base, and so on. The four most commonly used deployment models are as follows:

1. **Public Cloud**
2. **Private Cloud**
3. **Community Cloud**
4. **Hybrid Cloud**

1. Public Cloud

In this model, the infrastructure is accessible to the public and it is owned by a vendor, who offers the services of the cloud to the users. The cloud vendor shares the cloud resources with the end users. The resource pool is huge and the services are shared by lots of users. The services of this cloud model can be free or available for nominal charges. Google uses a public cloud deployment model. With this model, users need not purchase any infrastructure but can use that of the vendor. A drawback of the public cloud model is that it poses a security threat. If you have very confidential data running in your network, it is not safe to use the public cloud model.

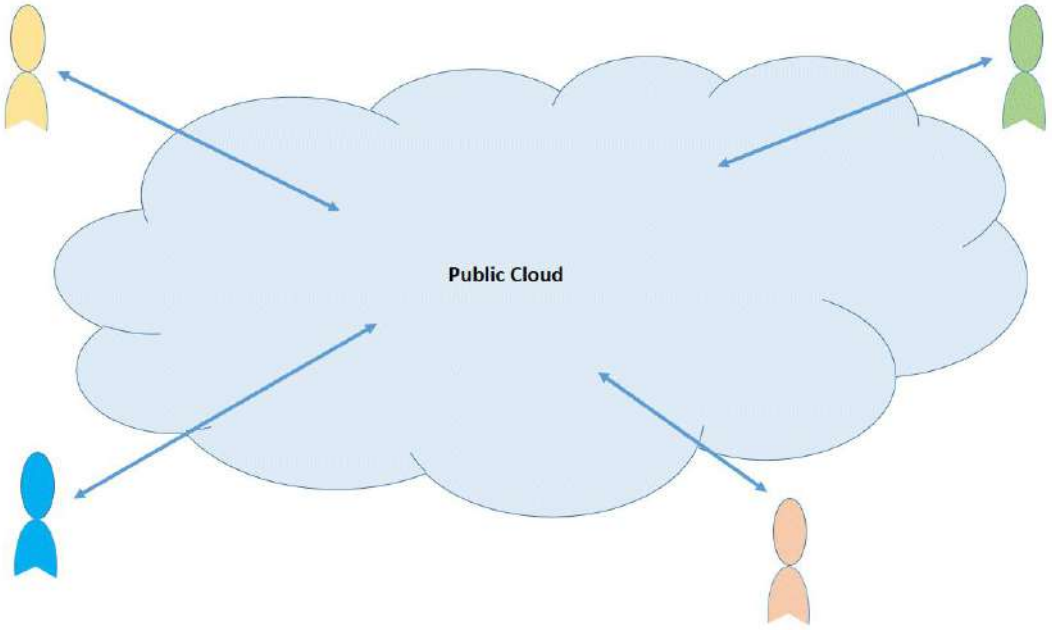


Fig2.2.1: Public Cloud

1. Private Cloud

As the name suggests, this would be a privately owned cloud. Here, the user or organization owns the cloud and only the user or employees of the company have access to the cloud, thereby making data and transactions secure. There is more control over resources when compared to the Public Cloud model. The Private cloud model uses the Virtualization solution and the data centers belong to the company. The major advantage of this model is the security and the control that the users have over the resources and application. However, the drawback is that more financial investment is required and the offering is not as big scale as that of a public cloud model.

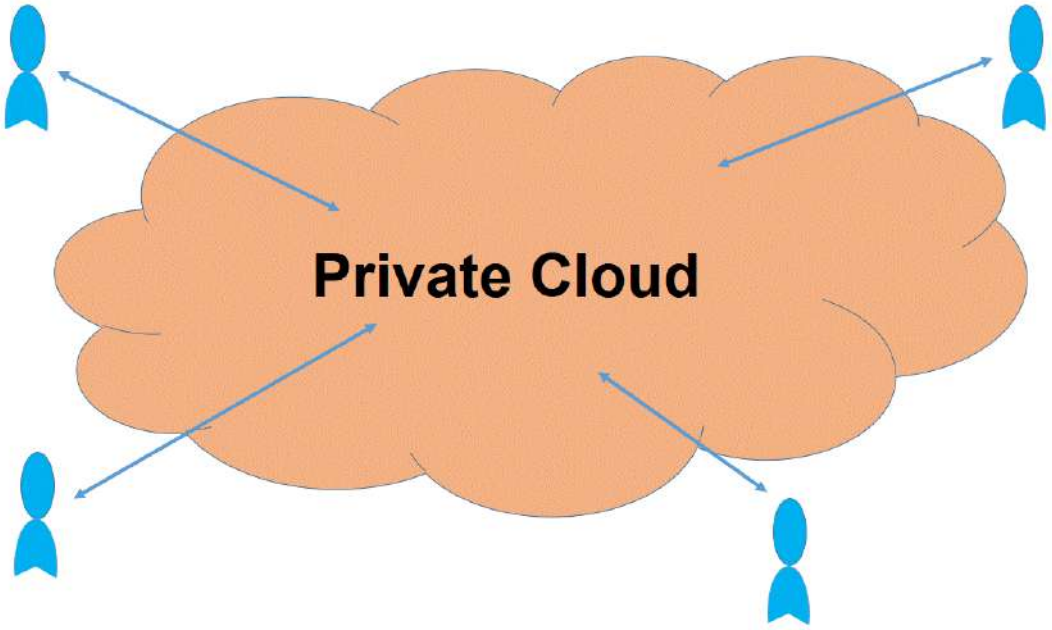


Fig2.2.2: Private Cloud

1. Community Cloud

In the Community Cloud model, the infrastructure is owned jointly by different organizations. The organizations may have a similar set of requirements, policies, and customer base. So, they can combine the offerings and make the customer base even bigger. Duplication of same or similar applications and resources are avoided. This model helps reduce the costs, which would otherwise be higher if the organization deploys the Private Cloud model. This is again a classification of the Private Cloud, as it is available to only a certain group of users.

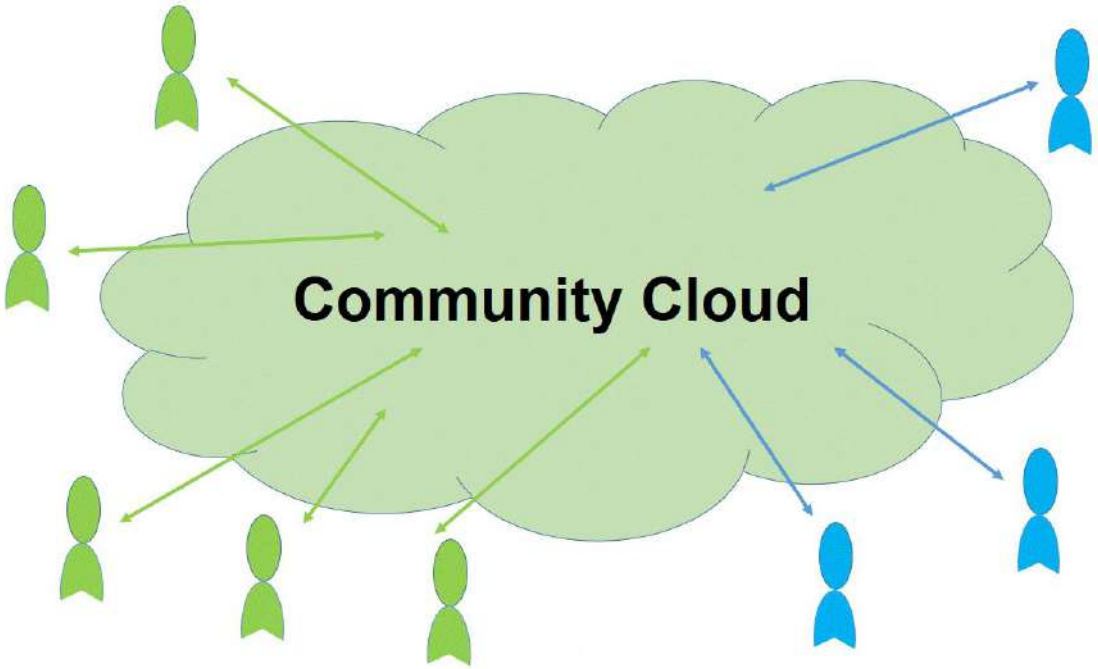


Fig2.2.3: Community Cloud

2. Hybrid Cloud

The Hybrid Cloud deployment model comprises of two or more clouds. This can be a combination of the other three cloud types – public, private, or community. The hybrid deployment is complex compared to the other three owing to the execution and management tasks involved. An example scenario of this model can be where an organization is on the private cloud but there are load spikes which the private cloud cannot handle. For this the organization depends on the public cloud to support the load. The shift from the private to the public cloud and back will be seamless to the end user

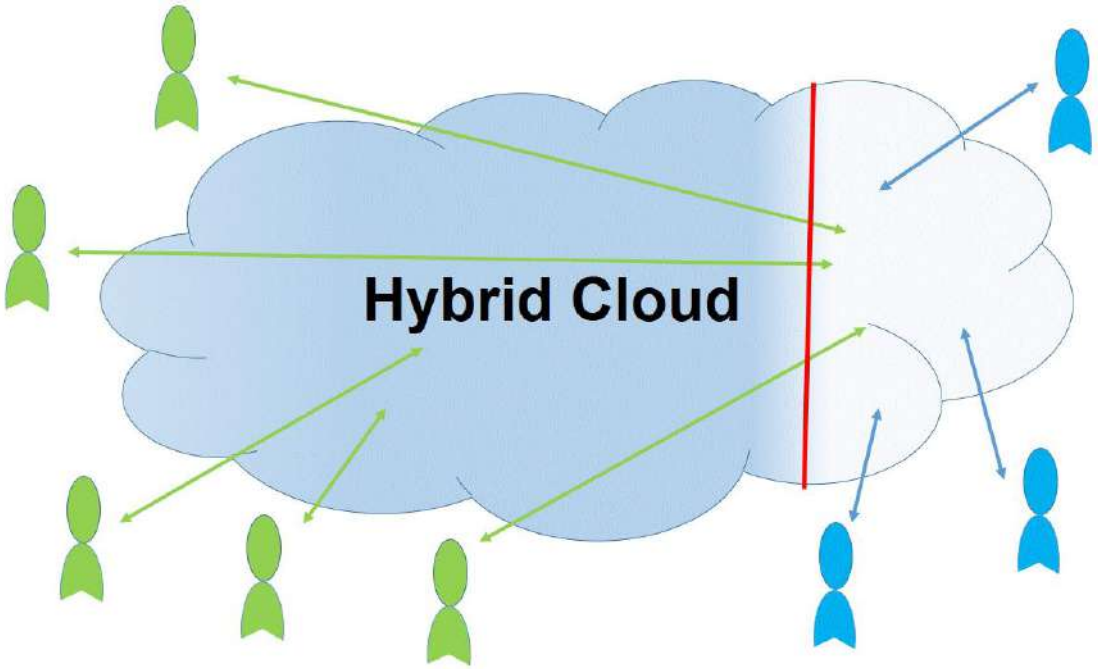


Fig2.2.4: Hybrid Cloud

Major Cloud Service Providers

- **Kamatera**
- **Softchoice**
- **Prolifics**
- **ScienceSoft**
- **phoenixNAP**
- **Cloudways**
- **pCloud**
- **Amazon Web Services**
- **Microsoft Azure**
- **Google Cloud Platform**
- **Adobe**
- **VMware**
- **IBM Cloud**
- **Rackspace**
- **Red Hat**
- **Salesforce**
- **Oracle Cloud**
- **SAP**
- **Verizon Cloud**
- **Navisite**
- **Dropbox**
- **Egnyte**

The following table summarizes the top 3 key players and their offerings in the cloud computing world:

	AWS	Azure	Google Cloud
Company	**AWS Inc.**	**Microsoft**	**Google**
Launch year	2006	2010	2008
Geographical Regions	25	54	21
Availability Zones	78	140 (countries)	61
Key offerings	Compute, storage, database, analytics, networking, machine learning, and AI, mobile, developer tools, IoT, security, enterprise applications, blockchain.	Compute, storage, mobile, data management, messaging, media services, CDN, machine learning and AI, developer tools, security, blockchain, functions, IoT.	Compute, storage, databases, networking, big data, cloud AI, management tools, Identity and security, IoT, API platform
Compliance Certificates	46	90	
Annual Revenue	$33 billion	$35 billion	$8 billion

Unit 2: Cloud Computing Deployment Models

Types of Cloud Computing Deployment Models are as :

1. Public Cloud
2. Private Cloud
3. Hybrid Cloud
4. Community Cloud

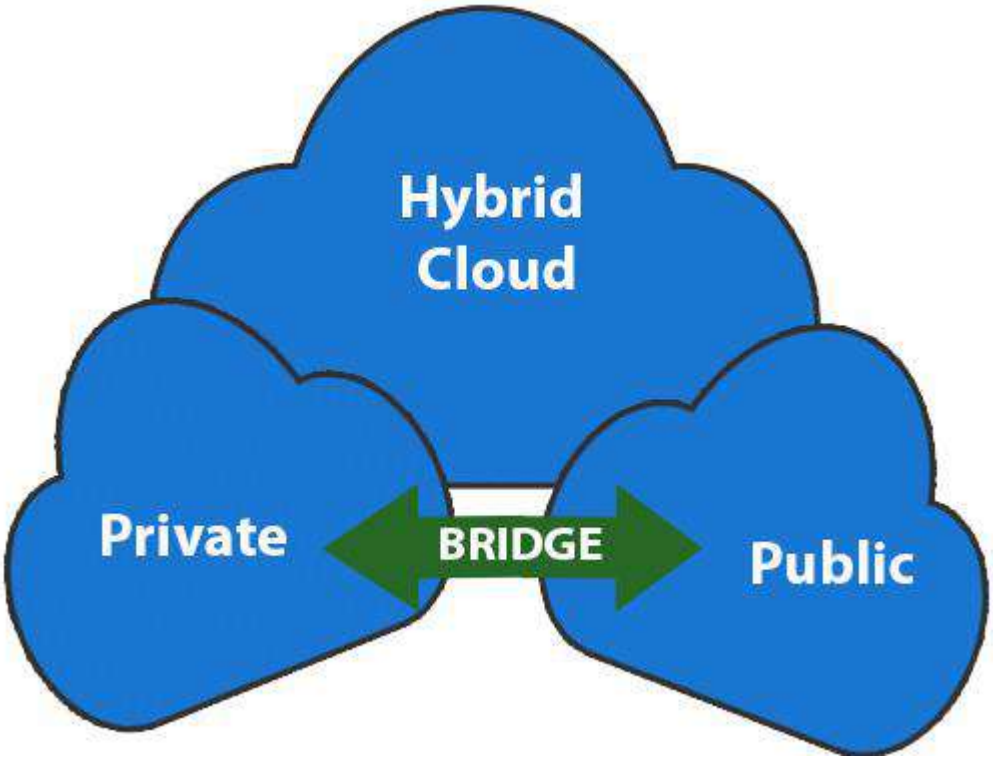


Fig: Types of Cloud Computing

1. **Public Cloud:** The cloud resources that are owned and operated by a third-party cloud service provider are termed as public clouds. It delivers computing resources such as servers, software, and storage over the internet
2. **Private Cloud:** The cloud computing resources that are exclusively used inside a single business or organization are termed as a private cloud. A private cloud may physically be located on the company's on-site data centre or hosted by a third-party service provider.
3. **Hybrid Cloud:** It is the combination of public and private clouds, which is bounded together by technology that allows data applications to be

shared between them. Hybrid cloud provides flexibility and more deployment options to the business.

Cloud SLA (cloud service-level agreement)

A cloud SLA (cloud service-level agreement) is an agreement between a cloud service provider and a customer that ensures a minimum level of service is maintained. It guarantees levels of reliability, availability and responsiveness to systems and applications, while also specifying who will govern when there is a service interruption.

A cloud infrastructure can span geographies, networks and systems that are both physical and virtual. While the exact metrics of a cloud SLA can vary by service provider, the areas covered are uniform: volume and quality of work -- including precision and accuracy -- speed, responsiveness and efficiency. The document aims to establish a mutual understanding of the services, prioritized areas, responsibilities, guarantees and warranties provided by the service provider.

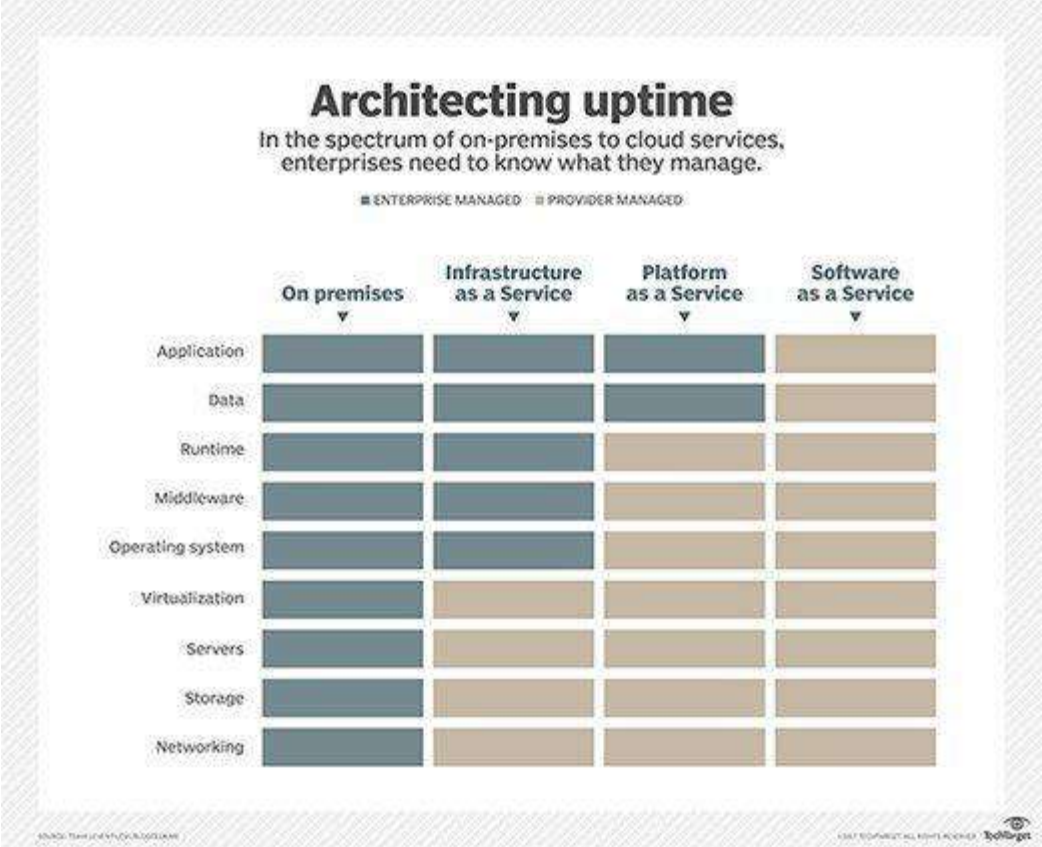
Metrics and responsibilities among the parties involved in cloud configurations are clearly outlined, such as the specific amount of response time for reporting or addressing system failures.

Financial penalties a provider must pay for failing to live up to the guaranteed terms are also included. These penalties are often in the form of credits for service time.

What to look for in a cloud SLA

Service-level agreements have become more important as organizations move their systems, applications and data to the cloud. A cloud SLA ensures that cloud providers meet certain enterprise-level requirements and provide customers with a clearly defined set of deliverables.

The defined level of services should be specific and measureable in each area. This allows the quality of service (QoS) to be benchmarked and, if stipulated by the agreement, rewarded or penalized accordingly.



An SLA will commonly use technical definitions that quantify the level of service, such as mean time between failures (MTBF) or mean time to repair (MTTR), which specifies a target or minimum value for service-level performance.

A typical compute and cloud SLA articulates precise levels of service, as well as the recourse or compensation the user is entitled to should the provider fail to deliver the service as described. Another area to consider carefully is service availability, which specifies the maximum amount of time a read request can take; how many retries are allowed; and so on.

The SLA should also define compensation for users if the specifications aren't met. A cloud storage service provider usually offers a tiered service credit plan that gives users credits based on the discrepancy between SLA specifications and the actual service levels delivered.

Most public cloud storage services provide details of the service levels that users can expect on their websites, and these will likely be the same for all users. However, an enterprise establishing a service with a private cloud storage

provider may be able to negotiate a more customized deal. In this case, the cloud SLA might include specifications for retention policies, the number of copies that will be retained, storage locations and so on.

Cloud service-level agreements may be more detailed to cover governance, security specifications, compliance, and performance and uptime statistics. They should address security and encryption practices for data privacy, disaster recovery expectations, data location, as well as data access and portability.

Data protection processes, such as backup and disaster recovery, should also be addressed. The agreement should outline the responsibilities of each party, the acceptable performance parameters, a description of the applications and services covered under the agreement, procedures for monitoring service levels, and a schedule for the remediation of outages.

Examine the ramifications of the cloud SLA before signing. For example, 99.9% uptime, a common stipulation, translates to nine hours of outage per year. For some mission-critical data, that may not be adequate. You should also check to see how terms are defined.

SLAs that scale

Most SLAs are negotiated to meet the needs of the customer at the time of signing, but many businesses change dramatically in size over time. A solid cloud service-level agreement outlines intervals for reviewing a contract so that it meets the changing needs of an organization.

Some vendors even build in notification workflows that indicate when a cloud service-level agreement is close to being breached so new negotiations can be initiated based on the changes in scale. When entering any cloud SLA negotiation, it's important to protect the business by clarifying uptimes. A good SLA protects both the customer and supplier from missed expectations.

Finally, the cloud SLA should include an exit strategy that outlines the expectations of the provider to ensure a smooth transition.

Service Level Agreements

https://www.youtube.com/watch?v=9Btb72NZwg4

Service Level Agreements in the Cloud: Who cares?

Service Level Agreements (SLAs) in the Cloud. There have been many articles written on the topic, but still there is confusion about the importance of SLAs. Most people require a blueprint for architects and contractors to start building a new home and similarly would expect a new car to come with a warranty. An SLA serves as both the blueprint and warranty for cloud computing.

There is an article written for Educause Quarterly by Thomas J. Trappler called "If It's in the Cloud, Get it on Paper: Cloud Computing Contract Issues". In his paper he recommends that the contract:

- Codifies the specific parameters and minimum levels required for each element of the service, as well as remedies for failure to meet those requirements.
- Affirms your institution's ownership of its data stored on the service provider's system, and specifies your rights to get it back.
- Details the system infrastructure and security standards to be maintained by the service provider, along with your rights to audit their compliance.
- Specifies your rights and cost to continue and discontinue using the service.

Using Trappler's thoughts, let's delve into why a SLA is important to ensuring the cloud meets the requirements of the enterprise.

In order to survive in today's world, one must be able to expect the unexpected as there are always new, unanticipated challenges. The only way to consistently overcome these challenges is to create a strong initial set of ground rules, and plan for exceptions from the start. Challenges can come from many fronts, such as networks, security, storage, processing power, database/software availability or even legislation or regulatory changes. As cloud customers, we operate in an environment that can spans geographies, networks, and systems. It only makes sense to agree on the desired service level for your customers and measure the real results. It only makes sense to set out a plan for when things go badly, so that a minimum level of service is maintained. Businesses depend on computing systems to survive.

In some sense, the SLA sets expectations for both parties and acts as the roadmap for change in the cloud service – both expected changes and surprises. Just as any IT project would have a roadmap with clearly defined deliverables, an SLA is equally critical for working with cloud infrastructure. That raises the next question in the journey: what should be in the SLA?

In order to consistently develop an effective SLA, a list of important criteria needs to be established. Let's start with an initial list:

- Availability (e.g. 99.99% during work days, 99.9% for nights/weekends)
- Performance (e.g. maximum response times)

- Security / privacy of the data (e.g. encrypting all stored and transmitted data)
- Disaster Recovery expectations (e.g. worse case recovery commitment)
- Location of the data (e.g. consistent with local legislation)
- Access to the data (e.g. data retrievable from provider in readable format)
- Portability of the data (e.g. ability to move data to a different provider)
- Process to identify problems and resolution expectations (e.g. call center)
- Change Management process (e.g. changes – updates or new services)
- Dispute mediation process (e.g. escalation process, consequences)
- Exit Strategy with expectations on the provider to ensure smooth transition

With a core set of criteria established, the next step is to evaluate the criticality of the cloud service and associated data. Nearly any computing system can be made extremely reliable, but the costs may be too high. Not every system needs the same degree of reliability as NASA designed for the space shuttles, and few could afford the costs.

For example, providing a read-only catalogue for customers is fairly simple. While the catalogue may be very high value, it is fairly easy to restore from backup with minimal customer impact. However, if the same service has an online shopping with financial transactions and customer data, then the risk level and also importance to the business just increased. The nature of the service is integral to determining the right SLA.

For every new cloud service, an SLA assessment process should be done. The SLA is a living agreement though and as services change, the SLA should be reassessed.

The SLA should act as a guide for handling potential problems. We need to look at the SLA as a tool for protecting the stability of the service, protecting the assets of the company and minimizing the expense should drastic actions be required. As an example, changing service providers and undoing the contracts in place, should be a last resort; it's a very expensive and painful solution. Nonetheless, it needs to be covered in the SLA so that both parties can disengage a lawsuit.

Lessons that have been learned with respect to SLAs:

1. Read your cloud provider's SLA very carefully – The almost four-day Amazon outage of April 2011 did not breach Amazon's EC2 SLA, which as a FAQ explains, "guarantees 99.95% availability of the service within a Region over a trailing 365 period." Since it has been the EBS and RDS services rather than EC2 itself that has failed (and all the failures have been restricted to Availability Zones within a single Region), the SLA has not been breached, legally speaking.
2. Get technical staff involved to validate SLAs against common outage scenarios – Another set of outages come along to delight those who follow

Microsoft's journey to cloud nirvana. On August 7 it was a power outage that affected their Dublin datacenter and affected service to European users of Business Productivity Online Services (BPOS).

3. Have contingency plans in place to support worse case scenarios – When we re-designed for the cloud this Amazon failure was exactly the sort of issue that we wanted to be resilient to. Our architecture avoids using EBS as our main data storage service, and the SimpleDB, S3 and Cassandra services that we do depend upon were not affected by the outage.

Bottom line, the SLA is your contract with the service provider and sets expectations for the relationship. It needs to be written to protect your cloud service(s) according to the level of risk you are prepared to accept. The goal is to have an SLA which both the cloud consumer and provider can understand and agree to, including an exit strategy. The SLA should be looked at as the document that establishes the partnership between the parties and is used to mitigate any problems.

Hopefully you can now agree that a SLA is required for a Cloud service and is for the benefit of both the consumer and the provider. In the long run, it will save both parties money and drive satisfaction for not only the parties directly involved, but more importantly, the end-users.

In should be noted, at the recent December member meeting, the Cloud Standards Customer Council launched a new Working Group to focus on the development of a Service Level Agreement Cookbook.

What is hypervisor?

The hypervisor is a key to enable virtualization. In its simpler form, the hypervisor is specialized firmware or software, or both, installed on single hardware that would allow you to host several virtual machines. It allows physical hardware to be shared across several virtual machines. A computer on which hypervisor runs one or more virtual machines is called a host machine. The virtual machine is called a guest machine. Basically, the hypervisor allows the physical host machine to run various guest machines. This helps in achieving maximum benefits from computing resources such as memory, network bandwidth, and CPU cycles.

Advantages of Hypervisors

- Though virtual machines operate on the same physical hardware, they are separated from each other. This also depicts that if one virtual machine undergoes a crash, error, or a malware attack, it doesn't affect the other virtual machines.
- Another benefit is that virtual machines are very mobile as they don't depend on the underlying hardware. Since they are not linked to physical hardware, switching between local or remote virtualized servers gets a lot easier as compared to traditional applications.

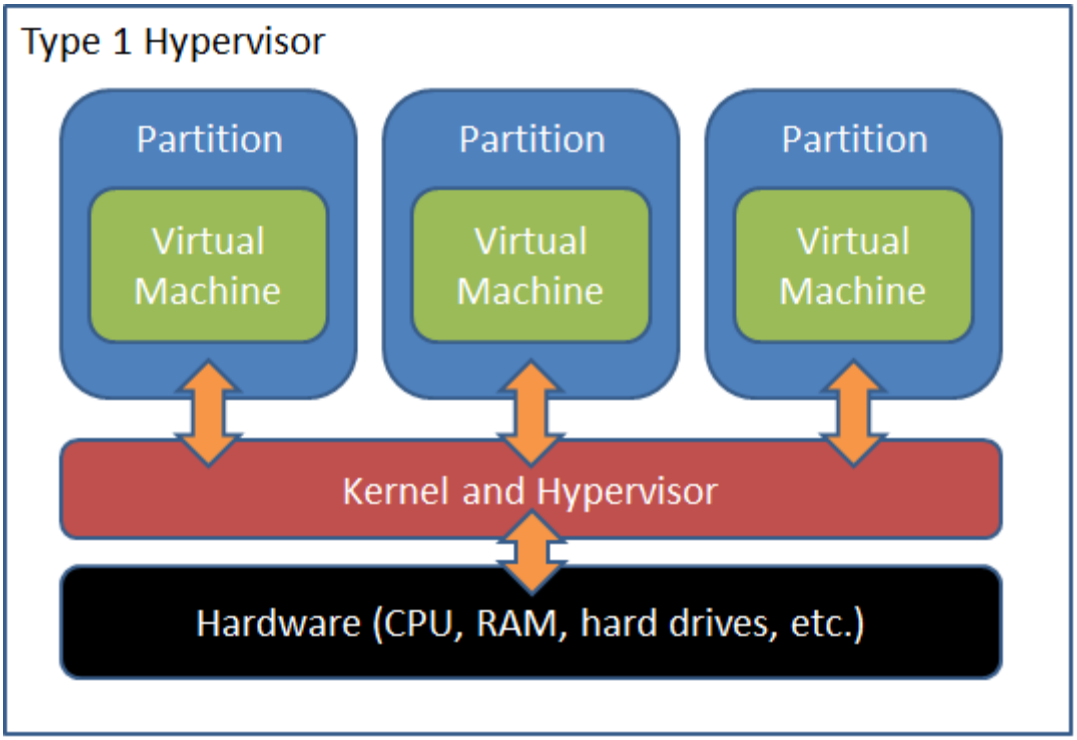
Types of Hypervisor In Cloud Computing

There are two main types of hypervisor in cloud computing.

Type I Hypervisor

A type I hypervisor operates directly on the host's hardware to monitor hardware and guest virtual machines, and it's referred to as the bare metal. Usually, they don't require the installation of software ahead of time. Instead, you can install right onto the hardware. This type of hypervisor tends to be powerful and requires a great deal of expertise to function it well. In addition, Type I hypervisor are more complex and have certain hardware requirements to run adequately. Due to this, it is mostly chosen by IT operations and data center computing.

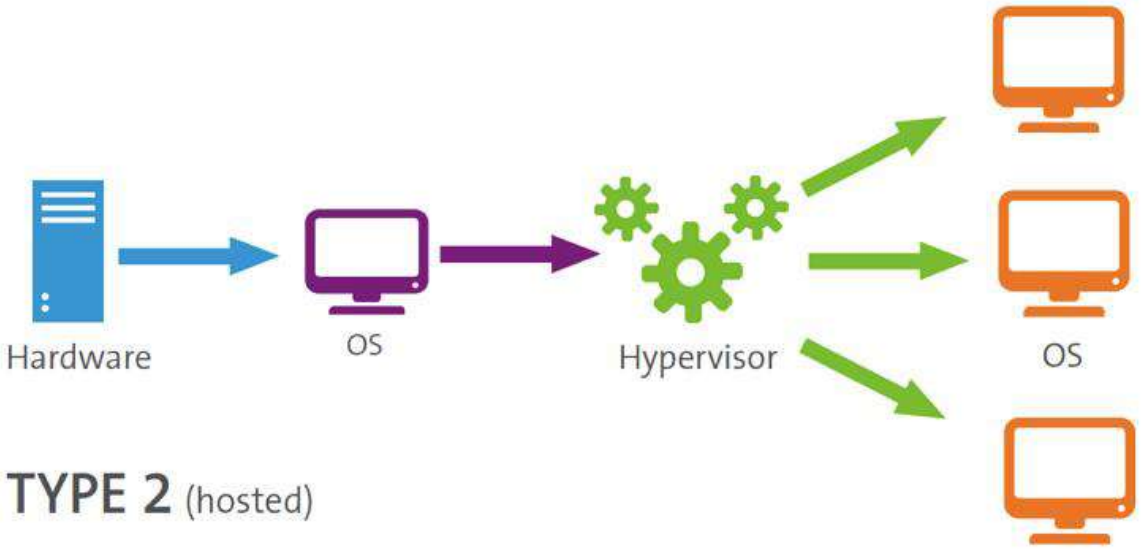
Examples of type I hypervisors include Xen, Oracle VM Server for SPARC, Oracle VM Server for x86, Microsoft Hyper-V and VMware’s ESX/ESXi.



Type II Hypervisor

It's also called a hosted hypervisor because it is usually installed onto an existing operating system. They are not much capable to run more complex virtual tasks. People use it for basic development, testing, and emulation. If there is any security flaw found inside the host OS, it can potentially compromise all of virtual machines running. This is why type II hypervisors cannot be used for data center computing. They are designed for end-user systems where security is a lesser concern. For instance, developers could use type II Hypervisor to launch virtual machines in order to test software product before their release.

A few examples are Virtual box, VMware workstation, fusion.



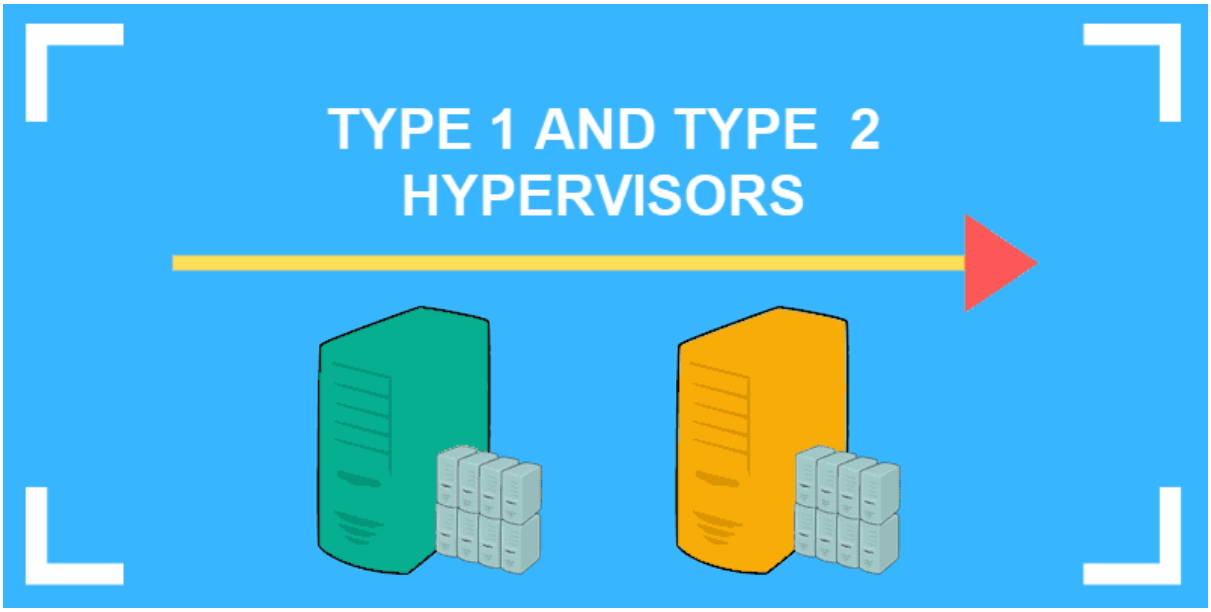
Conclusion

When you achieve virtualization, it brings a consolidation of multiple resources. This tends to reduce costs and improves manageability. In addition to it, a hypervisor can manage increased workloads. In a situation when a specific hardware node gets overheated, you can easily switch those virtual machines onto some other physical nodes. Virtualization also delivers other benefits of security, debugging and support.

What makes virtualization possible are hypervisors.

Server virtualization allows different operating systems running separate applications on one server while still using the same physical resources. These virtual machines make it possible for a system and network administrators to have a dedicated machine for every service they need to run.

Not only does this reduce the number of physical servers required, but it also saves time when trying to pinpoint issues.



What are Hypervisors?

A hypervisor is a crucial piece of software that makes virtualization possible. It abstracts guest machines and the operating system they run on, from the actual hardware.

Hypervisors create a virtualization layer that separates CPU / Processors, RAM and other physical resources from the virtual machines you create.

The machine we install a hypervisor on is called a ***host machine***, versus ***guest virtual machines*** that run on top of them.

Hypervisors emulate available resources so that guest machines can use them. No matter what operating system you boot up with a virtual machine, it will think that actual physical hardware is at its disposal.

From a VM's standpoint, there is no difference between the physical and virtualized environment. Guest machines do not know that the hypervisor created them in a virtual environment. Or that they are sharing available computing power. VMs run simultaneously with the hardware that powers them, and so they are entirely dependent on its stable operation.

- **Type 1 Hypervisor** (also called bare metal or native)
- **Type 2 Hypervisor** (also known as hosted hypervisors)

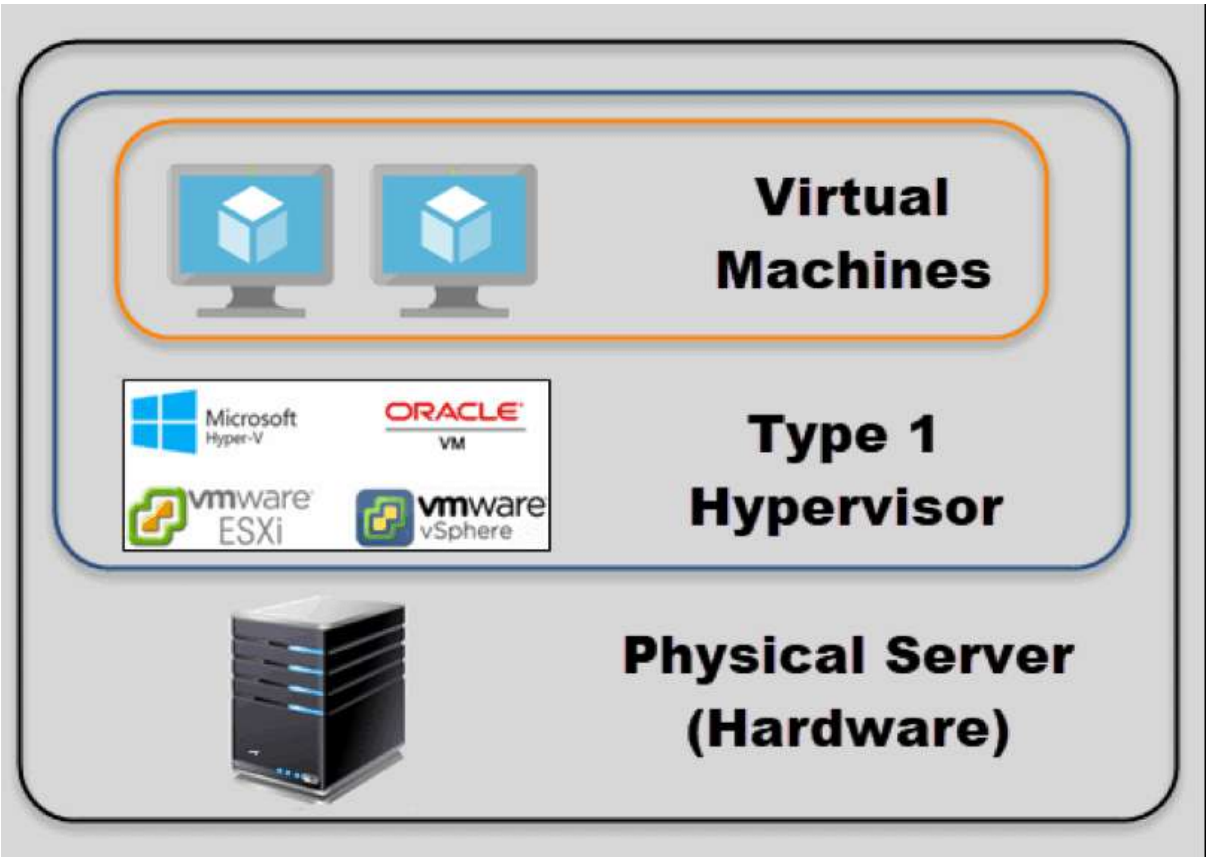
Type 1 Hypervisor

A bare-metal hypervisor (Type 1) is a layer of software we install directly on top of a physical server and its underlying hardware.

There is no software or any operating system in between, hence the name *bare-metal hypervisor*. A Type 1 hypervisor is proven in providing excellent performance and stability since it does not run inside Windows or any other operating system.

Type 1 hypervisors are an OS themselves, a very basic one on top of which you can run virtual machines. The physical machine the hypervisor is running on serves virtualization purposes only. You cannot use it for anything else.

Type 1 hypervisors are mainly found in enterprise environments.



Hypervisor Type 1 Performance

Given that type 1 hypervisors are relatively simple, they do not offer many functionalities.

Once you boot up a physical server with a bare-metal hypervisor installed, it displays a command prompt-like screen. If you connect a monitor to the server, what you get to see are some of the hardware and network details. This consists of the CPU type, the amount of memory, the IP address and the MAC address.

Below is an example of a VMware type 1 hypervisor's screen after the server boots up.

Another type 1 hypervisor may look quite different but they also only allow for simple server configuration. This consists of changing the date and time, IP address, password, etc. In order to create virtual instances, you need a management console set up on another machine. Using the console, you can connect to the hypervisor on the server, and manage your virtual environment.

A management console can be web-based or a separate software package you install on the machine for which you want remote management. Depending on what functionalities you need, the license cost for management consoles varies substantially.

One action you can perform includes moving virtual machines between physical servers, manually or automatically. This move is based on resource needs of a VM at a given moment and happens without any impact to the end-users. It's the same process if a piece of hardware or a whole server fails. Properly configured management software moves virtual machines to a working server as soon as an issue arises. The detection and restoration procedure takes place automatically and seamlessly.

One of the best features of type 1 hypervisors is that they allow for over-allocation of physical resources.

With type 1 hypervisors, you can assign more resources to your virtual machines than you have available. For example, if you have 128GB of RAM on your server and eight virtual machines, you can assign 24GB of RAM to each of them. This totals to 192GB of RAM, but VMs themselves will not actually consume all 24GB from the physical server. The VMs think they have 24GB when in reality they only use the amount of RAM they need to perform particular tasks.

The hypervisor allocates only the amount of necessary resources for an instance to be fully functional. This is one of the reasons all modern enterprise data centers, such as phoenixNAP, use type 1 hypervisors.

Type 1 Vendors

There are many different hypervisor vendors available. Most provide trial periods to test out their services before you buy them.

The licensing costs can be high if you want all the bells and whistles they have on offer.

These are the most common **type 1 hypervisors**:

VMware vSphere with ESX/ESXi

VMware is an industry-leading vendor of virtualization technology, and many large data centers run on their products. It may not be the most cost-effective solution for smaller IT environments. If you do not need all the advanced features VMware vSphere offers, there is a free version of this hypervisor and multiple commercial editions.

KVM (Kernel-Based Virtual Machine)

KVM is built into Linux as an added functionality. It lets you convert the Linux kernel into a hypervisor. It is sometimes confused with a type 2 hypervisor (see definition below). It has direct access to hardware along with virtual machines it

hosts. KVM is an open-source hypervisor that contains all the features of Linux with the addition of many other functionalities. This makes it one of the top choices for enterprise environments. Some of the highlights include live migration, scheduling and resource control, alongside higher prioritization.

Microsoft Hyper-V

Despite VMware's hypervisor being higher on the ladder with its numerous advanced features, Microsoft's Hyper-V has become a worthy opponent. Microsoft also offers a free edition of their hypervisor, but if you want a GUI and additional functionalities, you will have to go for one of the commercial versions. Hyper-V may not offer as many features as VMware vSphere package, but you still get live migration, replication of virtual machines, dynamic memory and many other features.

Oracle VM

This hypervisor has open-source Xen at its core and is free. Advanced features are only available in paid versions. Even though Oracle VM is essentially a stable product, it is not as robust as vSphere, KVM or Hyper-V.

Citrix Hypervisor (formerly known as Xen Server)

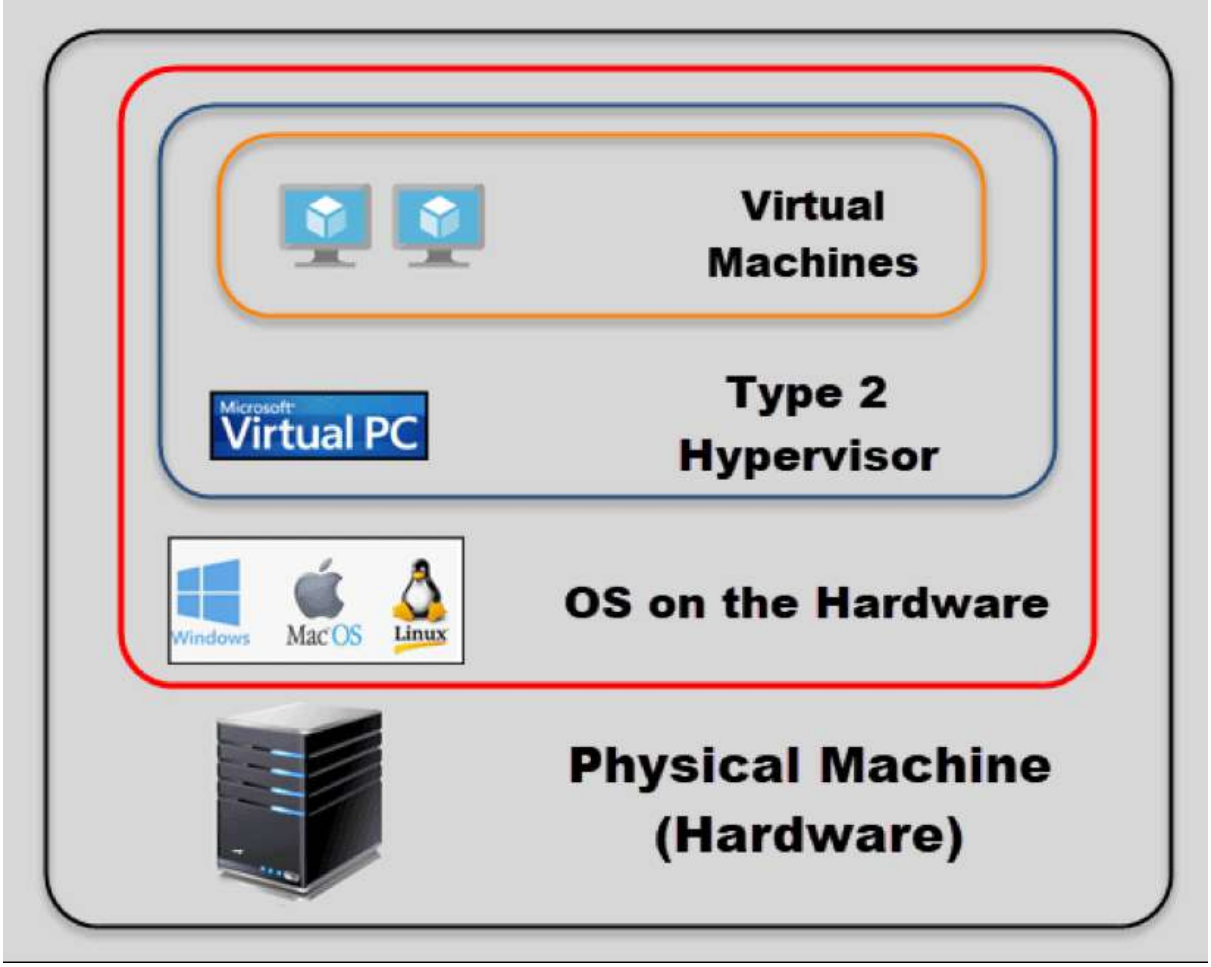
This Server virtualization platform by Citrix is best suited for enterprise environments. It can handle all types of workloads and provides features for the most demanding tasks. Citrix is proud of its proprietary features, such as Intel and NVIDIA enhanced virtualized graphics and workload security with Direct Inspect APIs.

Type 2 Hypervisor

This type of hypervisor runs inside of an operating system of a physical host machine.

This is why we call type 2 hypervisors – ***hosted hypervisors*****.** As opposed to type 1 hypervisors that run directly on the hardware, hosted hypervisors have one software layer underneath. In this case we have:

- A physical machine.
- An operating system installed on the hardware (Windows, Linux, macOS).
- A type 2 hypervisor software within that operating system.
- The actual instances of guest virtual machines.



Type 2 hypervisors are typically found in environments with a small number of servers.

What makes them convenient is that you do not need a management console on another machine to set up and manage virtual machines. You can do all of this on the server where you install the hypervisor. They are not any different from the other applications you have in your operating system.

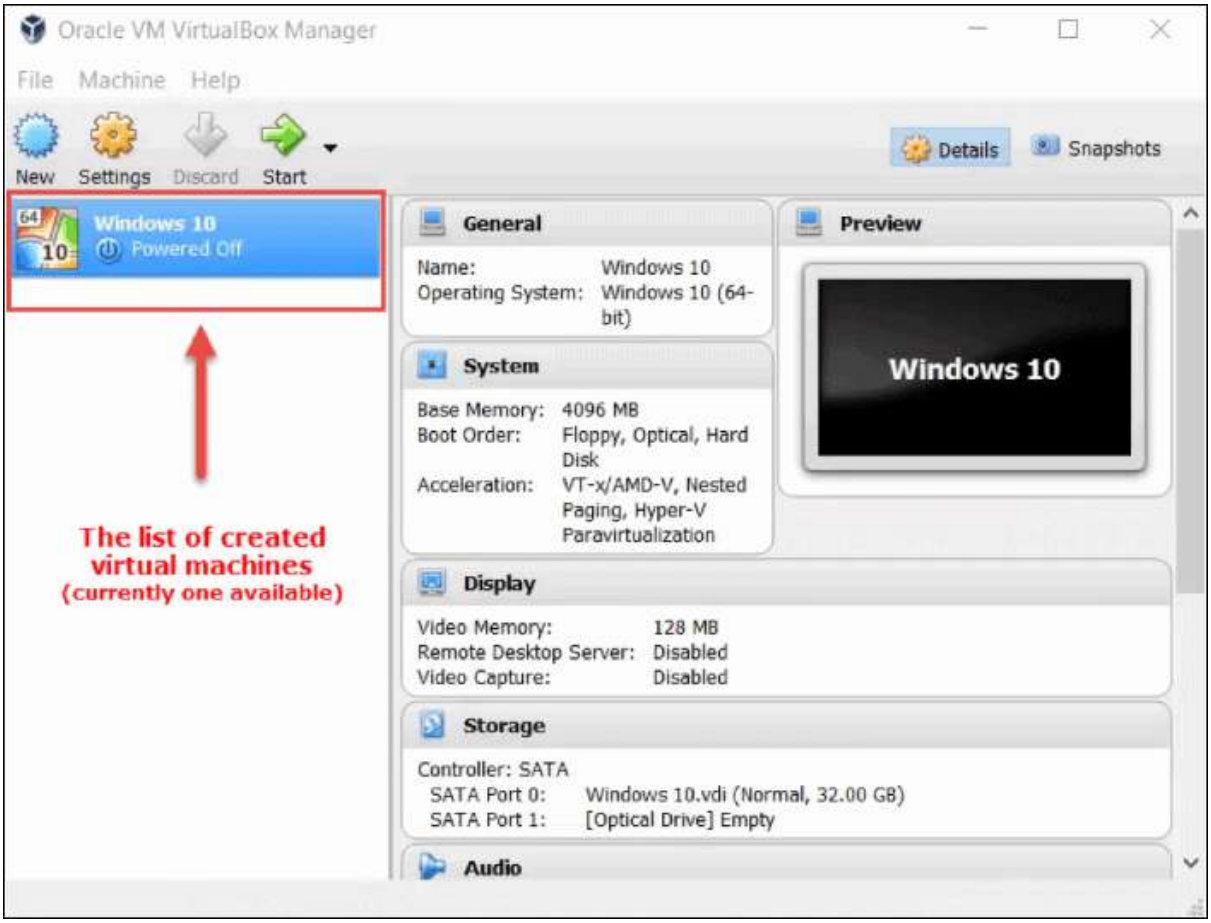
When you launch a virtual machine, you get another window to perform all tasks.

Hypervisor Type 2 Performance

Hosted hypervisors essentially also act as management consoles for virtual machines, you can perform any task using the built-in functionalities.

There is no need to install separate software on another machine to create and maintain your virtual environment. You simply install and run a type 2 hypervisor as you would any other application within your OS. With it, you can create snapshots or clone your virtual machines, import or export appliances, etc.

Here is one example of a type 2 hypervisor interface (VirtualBox by Oracle):



You do need to be careful when allocating actual resources with this type of hypervisor.

Bare-metal hypervisors can dynamically allocate available resources depending on the current needs of a particular VM. A type 2 hypervisor occupies whatever you allocate to a virtual machine.

When you assign 8GB of RAM to a VM, that amount will be taken up even if the VM is using only a fraction of it. If the host machine has 32GB of RAM and you create three VMs with 8GB each, you are left with 8GB of RAM to keep the physical machine running. Creating another VM with 8GB of ram would bring down your system. This is critical to keep in mind, so as to avoid over-allocating resources and crashing the host machine.

Type 2 hypervisors are convenient for testing new software and research projects.

It is possible to use one physical machine to run multiple instances with different operating systems to test how an application behaves in each environment or to create a specific network environment. You only need to make sure that there are

enough physical resources to keep both the host and the virtual machines running.

Type 2 Vendors

As is the case with bare-metal hypervisors, you can choose between numerous vendors and products. Conveniently, many type 2 hypervisors are free in their basic versions and provide sufficient functionalities.

Some even provide advanced features and performance boosts when you install add-on packages, free of charge. We will mention a few of the most used hosted hypervisors:

Oracle VM VirtualBox

A free but stable product with enough features for personal use and most use cases for smaller businesses. VirtualBox is not resource demanding, and it has proven to be a good solution for both desktop and server virtualization. It provides support for guest multiprocessing with up to 32 vCPUs per virtual machine, PXE Network boot, snapshot trees, and much more.

VMware Workstation Pro/VMware Fusion

VMware Workstation Pro is a type 2 hypervisor for Windows OS. It is full of advanced features and has seamless integration with vSphere. This allows you to move your apps between desktop and cloud environments.

It does come with a price tag, as there is no free version. If you want to take a glimpse into VMware hosted hypervisors free of charge, you can try VMware Workstation Player. This is the basic version of the hypervisor suitable for small sandbox environments.

For MacOS users, VMware has developed Fusion that is similar to their Workstation product. It comes with somewhat fewer features, but also carries a smaller price tag.

Windows Virtual PC

It only supports Windows 7 as a host machine and Windows OS on guest machines. This includes multiple versions of Windows 7 and Vista, as well as XP SP3. Virtual PC is completely free.

Parallels Desktop

A competitor to VMware Fusion. It is primarily intended for MacOS users and offers plenty of features depending on the version you purchase. Some of the features are network conditioning, integration with Chef/Ohai/Docker/Vagrant, support for up to 128GB per VM, etc.

Type 1 vs. Type 2 Hypervisor

Choosing the right type of hypervisor strictly depends on your individual needs.

The first thing you need to keep in mind is the size of the virtual environment you intend to run.

For personal use and smaller deployments, you can go for one of the type 2 hypervisors. If budget is not an issue, VMware will provide every feature you need. Otherwise, Oracle VM VirtualBox is a hypervisor that will provide most of the functionalities generally needed.

Enterprise Environments

Even though type 1 hypervisors are the way to go, you do need to take into consideration many factors before making a decision.

The critical factor is usually the licensing cost. This is where you need to pay extra attention since licensing may be per server, per CPU or sometimes even per core. In the current market, there is a battle going on between VMware vSphere and Microsoft Hyper-V. While Hyper-V was falling behind a few years ago, it has now become a valid choice, even for larger deployments. The same argument can be made for KVM.

Many vendors offer multiple products and layers of licenses to accommodate any organization. You may want to create a list of the requirements. Such as, how many VMs you need, maximum allowed resources per VM, nodes per cluster, specific functionalities, and then check which of these products best fits your needs. Note: trial periods can be very useful when testing for which hypervisor to choose.

In Closing

This article has explained what **a hypervisor is and the types of hypervisors (type 1 and type 2)** you can use.

Hypervisor vendors offer packages that contain multiple products with different licensing agreements. You will need to research the options thoroughly before making a final decision. Even though you can migrate between the hypervisors, this can be a tedious and expensive process. It’s best to get this decision right from the get go.

What is a hypervisor?

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

Why use a hypervisor?

Hypervisors make it possible to use more of a system's available resources and provide greater IT mobility since the guest VMs are independent of the host hardware. This means they can be easily moved between different servers. Because multiple virtual machines can run off of one physical server with a hypervisor, a hypervisor reduces:

- Space
- Energy
- Maintenance requirements

Types of hypervisors

There are two main hypervisor types, referred to as "Type 1" (or "bare metal") and "Type 2" (or "hosted"). A **type 1 hypervisor** acts like a lightweight operating system and runs directly on the host's hardware, while a **type 2 hypervisor** runs as a software layer on an operating system, like other computer programs.

The most commonly deployed type of hypervisor is the type 1 or bare-metal hypervisor, where virtualization software is installed directly on the hardware where the operating system is normally installed. Because bare-metal hypervisors are isolated from the attack-prone operating system, they are extremely secure. In addition, they generally perform better and more efficiently than hosted hypervisors. For these reasons, most enterprise companies choose bare-metal hypervisors for data center computing needs.

While bare-metal hypervisors run directly on the computing hardware, hosted hypervisors run on top of the operating system (OS) of the host machine. Although hosted hypervisors run within the OS, additional (and different) operating systems can be installed on top of the hypervisor. The downside of hosted hypervisors is that latency is higher than bare-metal hypervisors. This is because communication between the hardware and the hypervisor must pass through the extra layer of the OS. Hosted hypervisors are sometimes known as client hypervisors because they are most often used with end users and software testing, where higher latency is less of a concern.

Hardware acceleration technology can create and manage virtual resources faster by boosting processing speed for both bare-metal and hosted hypervisors. A type of hardware accelerator known as a **virtual Dedicated Graphics Accelerator** (vDGA) takes care of sending and refreshing high-end 3-D graphics. This frees up the main system for other tasks and greatly increases the display speed of images. For industries such as oil and gas exploration, where there is a need to quickly visualize complex data, this technology can be very useful.

Both types of hypervisors can run multiple virtual servers for multiple tenants on one physical machine. Public cloud service providers lease server space on the different virtual servers to different companies. One server might host several virtual servers that are all running workloads for different companies. This type of resource sharing can result in a "noisy neighbor" effect, when one of the tenants runs a large workload that interferes with the server performance for other tenants. It also poses more of a security risk than using a dedicated bare-metal server.

A bare-metal server that a single company has full control over will always provide higher performance than a virtual server that is sharing a physical server's bandwidth, memory and processing power with other virtual servers. The hardware for bare-metal servers can also be optimized to increase performance, which is not the case with shared public servers. Businesses that need to comply with regulations that require physical separation of resources will need to use their own bare-metal servers that do not share resources with other tenants.

What is a cloud hypervisor?

As cloud computing becomes pervasive, the hypervisor has emerged as an invaluable tool for running virtual machines and driving innovation in a cloud environment. Since a hypervisor is a software layer that enables one host computer to simultaneously support multiple VMs, hypervisors are a key element of the technology that makes cloud computing possible. Hypervisors make cloud-based applications available to users across a virtual environment while still enabling IT to maintain control over a cloud environment's infrastructure, applications and sensitive data.

Digital transformation and rising customer expectations are driving greater reliance on innovative applications. In response, many enterprises are migrating their virtual machines to the cloud. However, having to rewrite every existing application for the cloud can consume precious IT resources and lead to

infrastructure silos. Fortunately, as an integral part of a virtualization platform, a hypervisor can help migrate applications to the cloud quickly. As a result, enterprises can reap the cloud's many benefits, including reduced hardware expenditures, increased accessibility and greater scalability, for a faster return on investment.

How does a hypervisor work?

Hypervisors support the creation and management of virtual machines (VMs) by abstracting a computer's software from its hardware. Hypervisors make virtualization possible by translating requests between the physical and virtual resources. Bare-metal hypervisors are sometimes embedded into the firmware at the same level as the motherboard basic input/output system (BIOS) to enable the operating system on a computer to access and use virtualization software.

Benefits of hypervisors

There are several benefits to using a hypervisor that hosts multiple virtual machines:

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

Container vs hypervisor

Containers and hypervisors are both involved in making applications faster and more efficient, but they achieve this in different ways.

Hypervisors:

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

Containers:

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

Hypervisors and containers are used for different purposes. Hypervisors are used to create and run virtual machines (VMs), which each have their own complete operating systems, securely isolated from the others. In contrast to VMs, containers package up just an app and its related services. This makes them more lightweight and portable than VMs, so they are often used for fast and flexible application development and movement.