



*Cloud Computing*

# **Introduction to Cloud Computing**

# *Agenda*

- What is Cloud Computing ?
  - Different perspectives
  - Properties and characteristics
  - Benefits from cloud computing
- Service and deployment models
  - Three service models
  - Four deployment models



Properties and characteristics

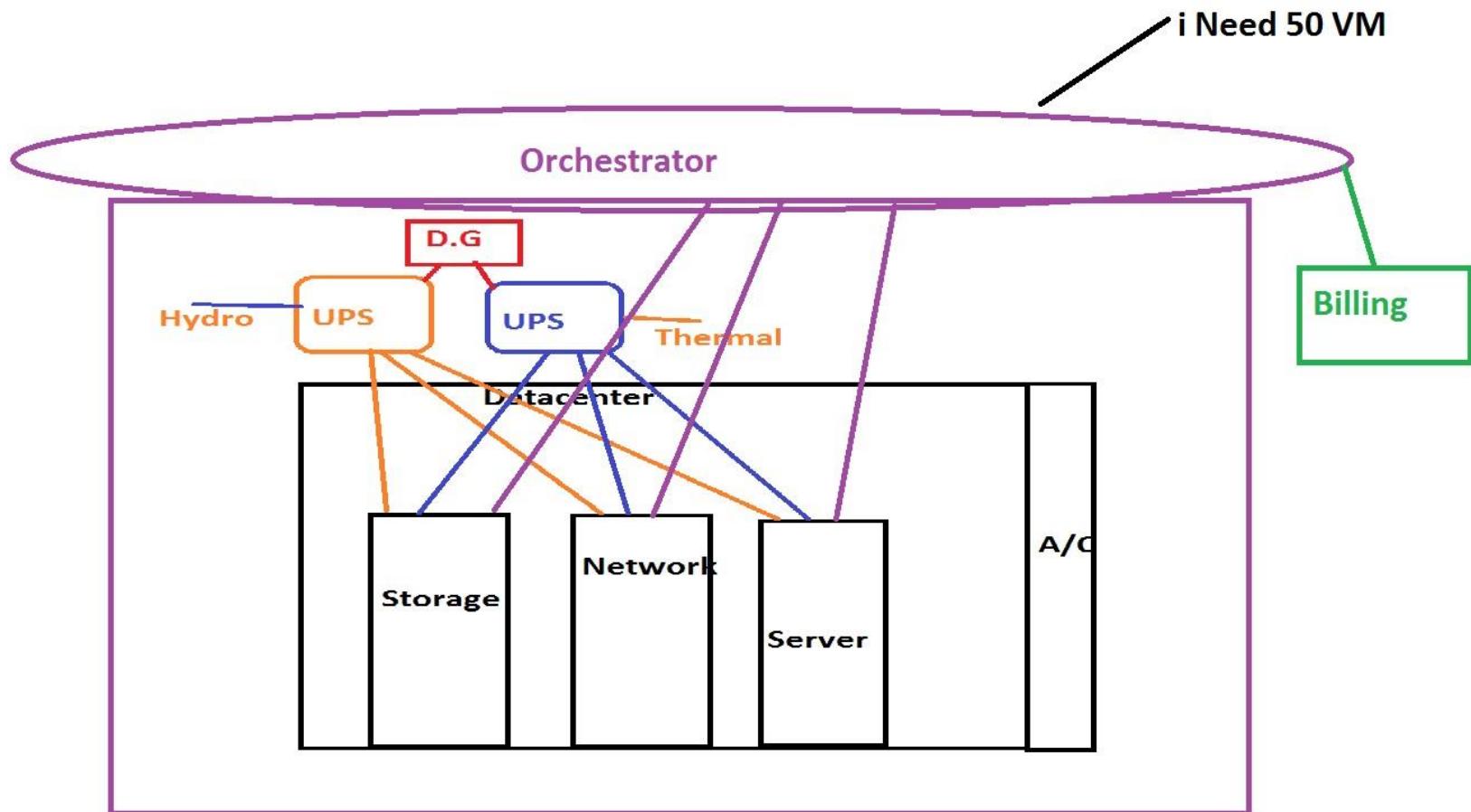
## ***WHAT IS CLOUD COMPUTING ?***

# *Cloud Disclaimers*

- Talk from Oracle CEO *Larry Ellison*
  - We've redefined Cloud Computing to include everything that we already do. I don't understand what we would do differently other than change the wording of some of our ads.
- Talk from *Rich Stallman*
  - It's stupidity. It's worse than stupidity: it's a marketing hype campaign. Somebody is saying this is inevitable – and whenever you hear somebody saying that, it's very likely to be a set of businesses campaigning to make it true.



# *Whats Behind the Cloud?*



# *In Our Humble Opinion*

- Cloud computing is a paradigm of computing, a new way of thinking about IT industry but not any specific technology.
  - Central ideas
    - **Utility Computing**
    - **SOA** - Service Oriented Architecture
    - **SLA** - Service Level Agreement
  - Properties and characteristics - **2AMPS**
    - High **availability** and **reliability**
    - High **accessibility** and **portability**
    - High **manageability** and **interoperability**
    - High **performance** and **optimization**
    - High **scalability** and **elasticity**
  - Enabling techniques
    - Hardware virtualization
    - Parallelized and distributed computing
    - Web service



# *Central Ideas*



**DON'T TELL ME DETAILS!!  
I DON'T CARE!!**

# Central Ideas

- Perspective from user :
  - Users do not care about how the works are done
    - Instead, they only concern about what they can get
  - Users do not care about what the provider actually did
    - Instead, they only concern about their quality of service
  - Users do not want to own the physical infrastructure
    - Instead, they only want to pay as many as they used
- What dose user really care ?
  - They only care about their “Service”



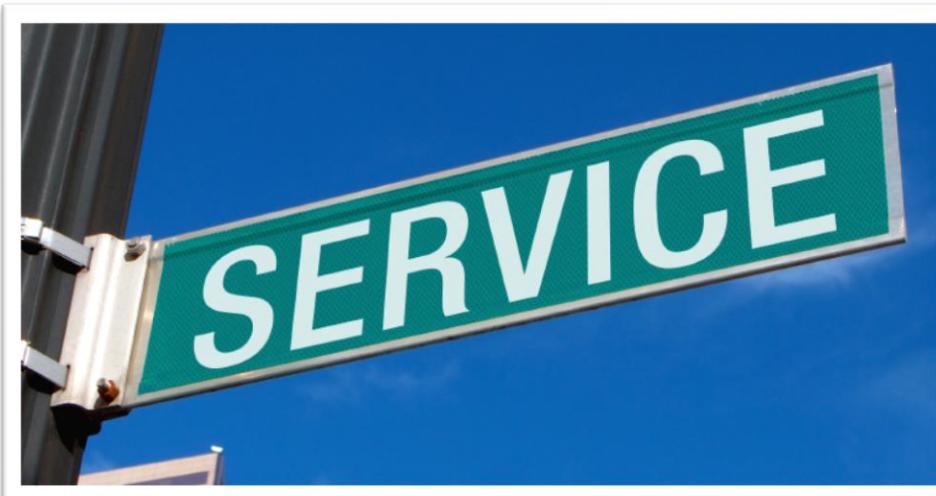
# Utility Computing

- One service provisioning model
  - Service provider makes computing resources and infrastructure management available to the customer as needed, and charges them for specific usage rather than a flat rate.
  - Like other types of on-demand computing , the utility model seeks to maximize the efficient use of resources and/or minimize associated costs.



# What Is Service?

- Service is what you connect together using Web Services.
- Service is the endpoint of a connection.
- Functionalities of service :
  - A service should be well-defined
  - A service should be self-contained
  - A service should not depend on the context or state of other services.

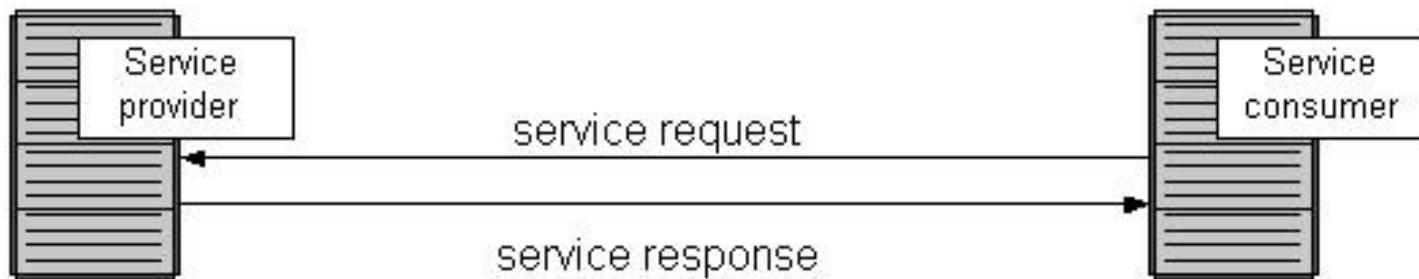


# What Is Web Service?

- Definition :
  - Web service is self-describing and stateless modules that perform discrete units of work and are available over the network
  - Web service providers offer APIs that enable developers to exploit functionality over the Internet, rather than delivering full-blown applications
- Web Services Description Language (WSDL) :
  - Expressed in XML which include both data type and messages
  - Four types of operations :
    - One-way - Messages sent without a reply required
    - Request & response - Sending and replying messages
    - Solicit response - A request for a response
    - Notification - Messages sent to multiple receivers

# Service Oriented Architecture

- Definition
  - Service Oriented Architecture (SOA) is essentially a collection of services which communicate with each other
  - Contain a flexible set of design principles used during the phases of systems development and integration
  - Provide a loosely-integrated suite of services that can be used within multiple business domains
- Approach
  - Usually implemented by Web Service model



# *Quality Of Service*

- Original definition
  - Quality of Service (QoS) is a set of technologies for managing network traffic in a cost effective manner to enhance user experiences for home and enterprise environments.
- Now QoS becomes to a broad term that is used following areas :
  - Customer care evaluations
  - Technological evaluations



# *Quality Of Service*

- Customer care evaluations
  - QoS is usually measured in terms of issues that have a direct impact on the experience of the customer
  - Only issues that produce a negative effect on the goods and services received by the customer come under scrutiny
- Technological evaluations
  - QoS has to do with the efficient operation of various systems
  - This can lead to adjusting procedures or adapting software programs and code to achieve the desired effect while making a more efficient use of available resources

# Service Level Agreement

- Definition
  - A service-level agreement (SLA) is a contract between a network service provider and a customer that specifies, usually in measurable terms (QoS), what services the network service provider will furnish
- Common content in contract
  - Performance guarantee metrics
    - Up-time and down-time ratio
    - System throughput
    - Response time
  - Problem management detail
  - Penalties for non-performance
  - Documented security capabilities



# *Properties and Characteristics*



**Availability  
Reliability**

- Fault tolerance
- System resilience
- System security

# ***Availability & Reliability***



**Data Never Loss  
Machine Never Fail**

**Availability  
Reliability**

- Fault tolerance
- System resilience
- System security

# ***Availability & Reliability***

- What is availability ?
  - The degree to which a system, subsystem, or equipment is in a specified operable and committable state at the start of a mission, when the mission is called for at an unknown time.
  - Cloud system usually require high availability
    - Ex. “Five Nines” system would statistically provide 99.999% availability
- What is reliability ?
  - The ability of a system or component to perform its required functions under stated conditions for a specified period of time.
- But how to achieve these properties ?
  - Fault tolerance system
  - Require system resilience
  - Reliable system security

**Availability  
Reliability**

- Fault tolerance
- System resilience
- System security

# **Fault Tolerance**

- What is fault tolerant system ?
  - Fault-tolerance is the property that enables a system to continue operating properly in the event of the failure of some of its components.
  - If its operating quality decreases at all, the decrease is proportional to the severity of the failure, as compared to a naively-designed system in which even a small failure can cause total breakdown.
- Four basic characteristics :
  - No single point of failure
  - Fault detection and isolation to the failing component
  - Fault containment to prevent propagation of the failure
  - Availability of reversion modes

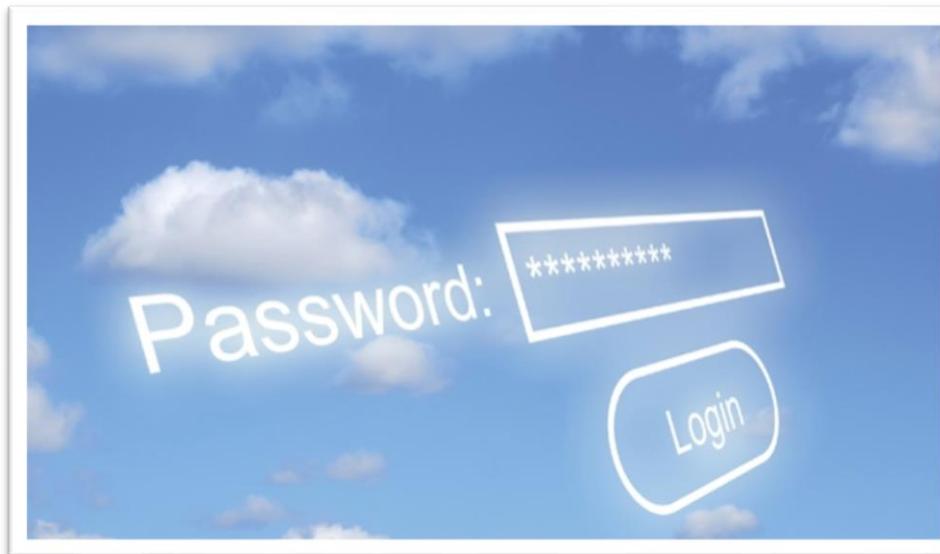
**Availability  
Reliability**

- Fault tolerance
- System resilience
- System security

# **System Security**

- **Security issue in Cloud Computing :**

- Cloud security is an evolving sub-domain of computer security, network security, and, more broadly, information security.
- It refers to a broad set of policies, technologies, and controls deployed to protect data, applications, and the associated infrastructure of cloud computing.



**Availability  
Reliability**

- Fault tolerance
- System resilience
- System security

# **System Security**

- Important security and privacy issues :
  - Data Protection
    - To be considered protected, data from one customer must be properly segregated from that of another.
  - Identity Management
    - Every enterprise will have its own identity management system to control access to information and computing resources.
  - Application Security
    - Cloud providers should ensure that applications available as a service via the cloud are secure.
  - Privacy
    - Providers ensure that all critical data are masked and that only authorized users have access to data in its entirety.

**Accessibility  
Portability**

- Uniform access
- Thin client

# ***Accessibility & Portability***



**Anyone !  
Anytime !  
Anywhere !**

**Accessibility  
Portability**

- Uniform access
- Thin client

# ***Accessibility & Portability***

- What is accessibility ?
  - Accessibility is a general term used to describe the degree to which a product, device, service, or environment is accessible by as many people as possible.
- What is service portability ?
  - Service portability is the ability to access services using any devices, anywhere, continuously with mobility support and dynamic adaptation to resource variations.
- But how to achieve these properties ?
  - Uniform access
  - Thin client

**Accessibility  
Portability**

- Uniform access
- Thin client

# ***Uniform Access***

- How do users access cloud services ?

- Cloud provider should provide their cloud service by means of widespread accessing media. In other word, users from different operating systems or other accessing platforms should be able to directly be served.
- Nowadays, web browser technique is one of the most widespread platform in almost any intelligent electronic devices. Cloud service take this into concern, and delivery their services with web-based interface through the Internet.



# Thin Client

- What is thin client ?
  - Thin client is a computer or a computer program which depends heavily on some other computer to fulfill its traditional computational roles. This stands in contrast to the traditional fat client, a computer designed to take on these roles by itself.
- Characteristics :
  - Cheap client hardware
    - While the cloud providers handle several client sessions at once, the clients can be made out of much cheaper hardware.
  - Diversity of end devices
    - End user can access cloud service via plenty of various electronic devices, which include mobile phones and smart TV.
  - Client simplicity
    - Client local system do not need complete operational functionalities.

Manageability  
Interoperability

- Control automation
- System monitoring
- Billing system

# *Manageability & Interoperability*



**I Want Full Control !!**

# Manageability & Interoperability

- What is manageability ?
  - Enterprise-wide administration of cloud computing systems. Systems manageability is strongly influenced by network management initiatives in telecommunications.
- What is interoperability ?
  - Interoperability is a property of a product or system, whose interfaces are completely understood, to work with other products or systems, present or future, without any restricted access or implementation.
- But how to achieve these properties ?
  - System control automation
  - System state monitoring

Manageability  
Interoperability

- Control automation
- System monitoring
- Billing system

# *Control Automation*

- Four functional areas :
  - Self-Configuration
    - Automatic configuration of components.
  - Self-Healing
    - Automatic discovery, and correction of faults.
  - Self-Optimization
    - Automatic monitoring and control of resources to ensure the optimal functioning with respect to the defined requirements.
  - Self-Protection
    - Proactive identification and protection from arbitrary attacks.

# System Monitoring

- What is system monitor ?
  - A System Monitor in systems engineering is a process within a distributed system for collecting and storing state data.
- What should be monitored in the Cloud ?
  - Physical and virtual hardware state
  - Resource performance metrics
  - Network access patterns
  - System logs
  - ... etc
- Anything more ?
  - Billing system



# Billing System

- Billing System in Cloud
  - Users pay as many as they used.
  - Cloud provider must first determine the list of service usage price.
  - Cloud provider have to record the resource or service usage of each user, and then charge users by these records.
- How can cloud provider know users' usage ?
  - Get those information by means of monitoring system.
  - Automatically calculate the total amount of money which user should pay. And automatically request money from use's banking account.



Performance  
Optimization

- Parallel processing
- Load balancing
- Job scheduling

# Performance & Optimization



High Performance  
Improvement

**Performance  
Optimization**

- Parallel processing
- Load balancing
- Job scheduling

# *Performance & Optimization*

- Performance guarantees ??
  - As the great computing power in cloud, application performance should be guaranteed.
  - Cloud providers make use of powerful infrastructure or other underlining resources to build up a highly performed and highly optimized environment, and then deliver the complete services to cloud users.
- But how to achieve this property ?
  - Parallel computing
  - Load balancing
  - Job scheduling

# Parallel Processing

- Parallel Processing
  - Parallel processing is a form of computation in which many calculations are carried out simultaneously, operating on the principle that large problems can often be divided into smaller ones, which are then solved concurrently.
- Parallelism in different levels :
  - Bit level parallelism
  - Instruction level parallelism
  - Data level parallelism
  - Task level parallelism

Performance  
Optimization

- Parallel processing
- Load balancing
- Job scheduling

# Parallel Processing

- Hardware approaches
  - Multi-core computer
  - Symmetric multi-processor
  - General purpose graphic processing unit
  - Vector processor
  - Distributed computing
    - Cluster computing
    - Grid computing
- Software approaches
  - Parallel programming language
  - Automatic parallelization



Performance  
Optimization

- Parallel processing
- Load balancing
- Job scheduling

# Load Balancing

- What is load balancing ?
  - Load balancing is a technique to distribute workload evenly across two or more computers, network links, CPUs, hard drives, or other resources, in order to get optimal resource utilization, maximize throughput, minimize response time, and avoid overload.
- Why should be load balanced ?
  - Improve resource utilization
  - Improve system performance
  - Improve energy efficiency

Unbalanced →



# Job Scheduling

- What is job scheduler ?
  - A job scheduler is a software application that is in charge of unattended background executions, commonly known for historical reasons as batch processing.
- What should be scheduled in Cloud ?
  - Computation intensive tasks
  - Dynamic growing and shrinking tasks
  - Tasks with complex processing dependency
- How to approach ?
  - Use pre-defined workflow
  - System automatic configuration

Scalability  
Elasticity

- Dynamic provision
- Multi-tenant design

# *Scalability & Elasticity*



**Give me the world  
without limitation!!**

## Scalability Elasticity

- Dynamic provision
- Multi-tenant design

# Scalability & Elasticity

- What is scalability ?
  - A desirable property of a system, a network, or a process, which indicates its ability to either handle growing amounts of work in a graceful manner or to be readily enlarged.
- What is elasticity ?
  - The ability to apply a quantifiable methodology that allows for the basis of an adaptive introspection within a real time infrastructure.
- But how to achieve these properties ?
  - Dynamic provisioning
  - Multi-tenant design

**Scalability**  
**Elasticity**

- Dynamic provision
- Multi-tenant design

# *Multi-tenant Design*

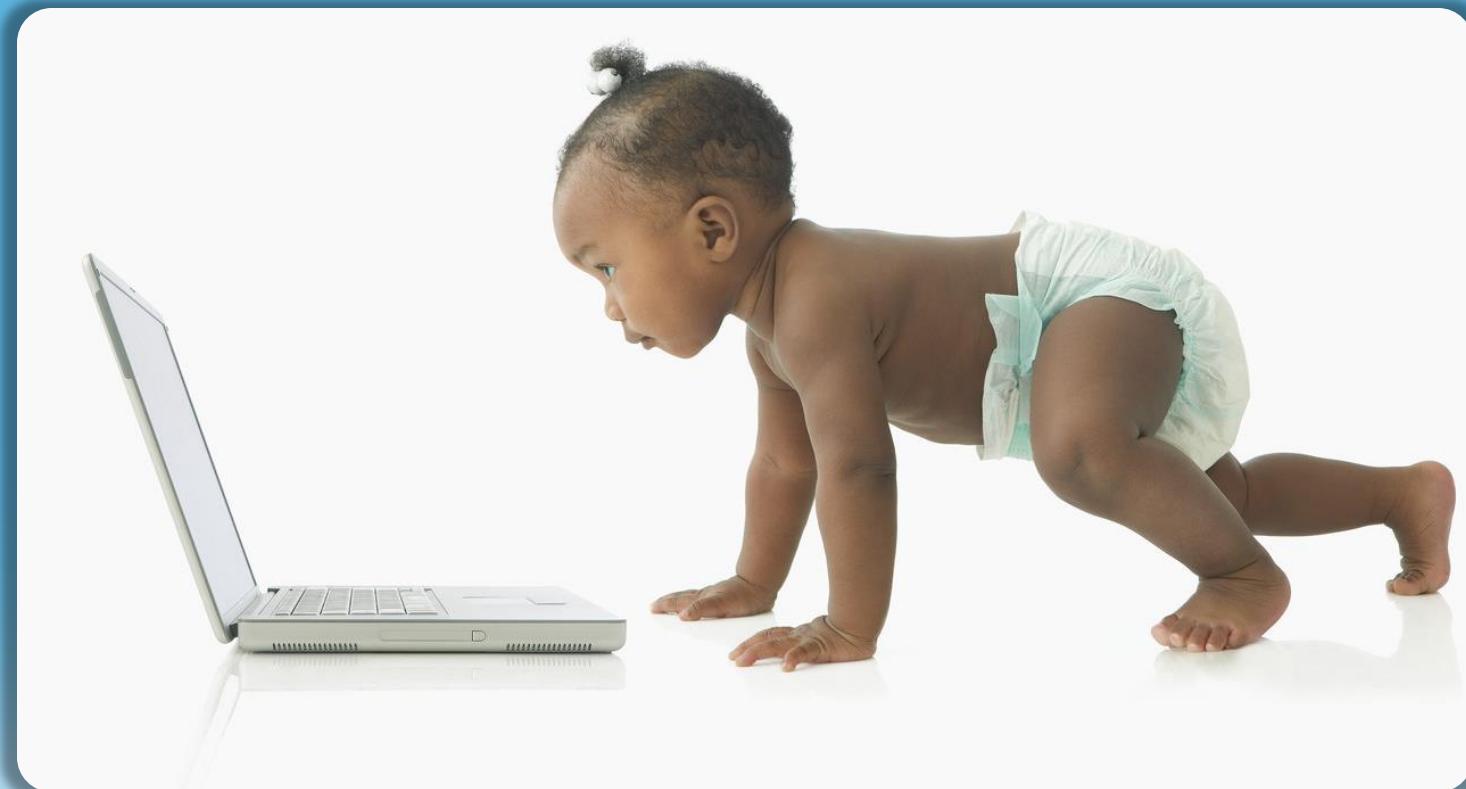
- What is multi-tenant design ?
  - Multi-tenant refers to a principle in software architecture where a single instance of the software runs on a server, serving multiple client organizations.
  - With a multi-tenant architecture, a software application is designed to virtually partition its data and configuration thus each client organization works with a customized virtual application instance.
- Client oriented requirements :
  - Customization
    - Multi-tenant applications are typically required to provide a high degree of customization to support each target organization's needs.
  - Quality of service
    - Multi-tenant applications are expected to provide adequate levels of security and robustness.

# *Benefits From Cloud*

- Cloud computing brings many benefits :
  - For the market and enterprises
    - Reduce initial investment
    - Reduce capital expenditure
    - Improve industrial specialization
    - Improve resource utilization
  - For the end user and individuals
    - Reduce local computing power
    - Reduce local storage power
    - Variety of thin client devices in daily life



# For End User and Individual



# *Reduce Local Computing Power*

- Traditional local computing power requirement :
  - One need to buy your own personal computer
  - Buy powerful processor if you need intensive computing
  - Buy large memory to meet application requirement
  - Install plenty of applications in need
- Some drawbacks :
  - One can hardly replicate the same system environment
  - One needs to regularly update or upgrade software and hardware
  - One needs to reinstall all applications if you reinstall the OS

# *Reduce Local Computing Power*

- Using Cloud Computing services :
  - One can utilize the remote computing power in the cloud
  - One needs only basic computing power to connect to internet
  - Application in the cloud will automatically upgrade
- Some benefits :
  - One can access his/her applications anywhere through the Internet
  - One can dynamically request for computing power on demand
  - Application may need not to be reinstalled even reinstall the OS

# *What is Virtualization??*

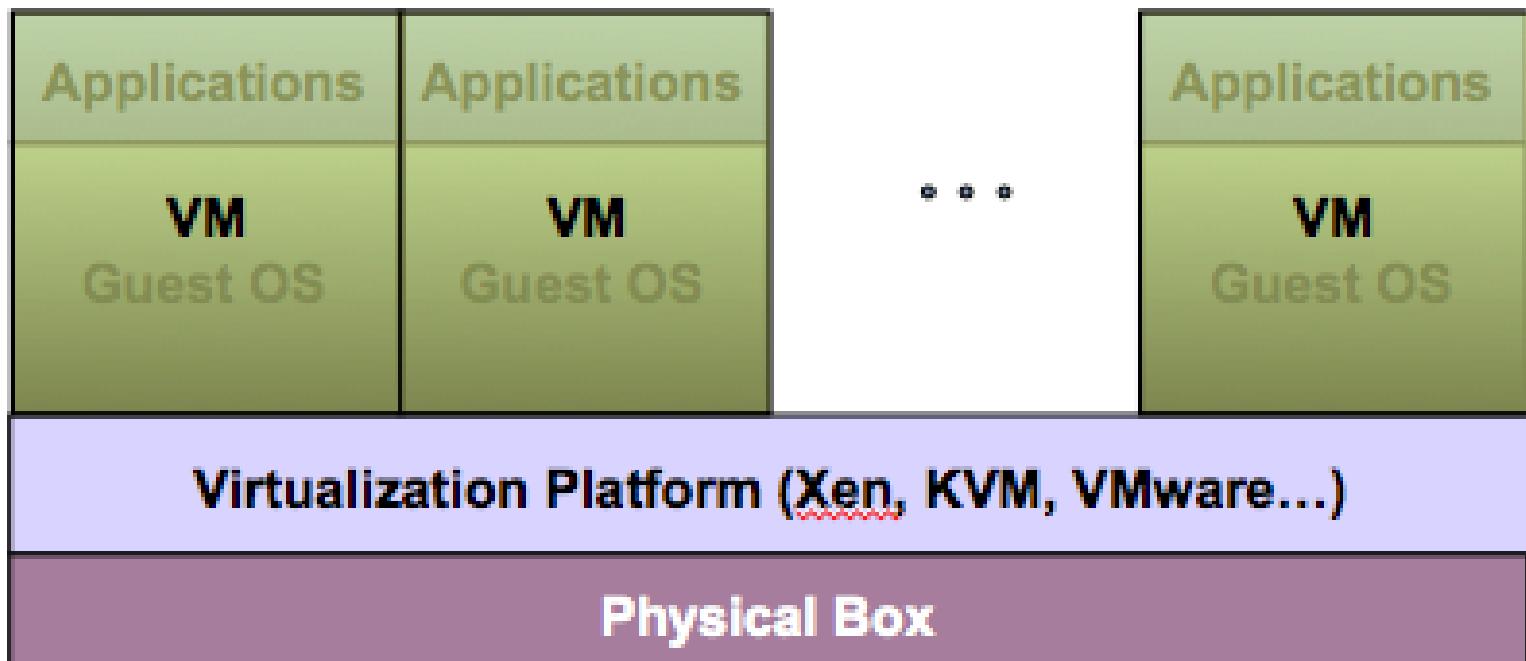
# *Definition*

- **Virtualization** is the ability to run multiple operating systems on a single physical system and share the underlying hardware resources\*
- It is the process by which one computer hosts the appearance of many computers.
- Virtualization is used to improve IT throughput and costs by using physical resources as a pool from which virtual resources can be allocated.

\*VMWare white paper, *Virtualization Overview*

# *Virtualization Architecture*

- A Virtual machine (VM) is an isolated runtime environment (guest OS and applications)
- Multiple virtual systems (VMs) can run on a single physical system



# *Hypervisor*

- A **hypervisor**, a.k.a. a virtual machine manager/monitor (VMM), or virtualization manager, is a program that allows multiple operating systems to share a single hardware host.
- Each guest operating system appears to have the host's processor, memory, and other resources all to itself. However, the hypervisor is actually controlling the host processor and resources, allocating what is needed to each operating system in turn and making sure that the guest operating systems (called virtual machines) cannot disrupt each other.

# *Benefits of Virtualization*

- Sharing of resources helps cost reduction
- Isolation: Virtual machines are isolated from each other as if they are physically separated
- Encapsulation: Virtual machines encapsulate a complete computing environment
- Hardware Independence: Virtual machines run independently of underlying hardware
- Portability: Virtual machines can be migrated between different hosts.

# *Virtualization in Cloud Computing*

Cloud computing takes virtualization one step further:

- You don't need to own the hardware
- Resources are rented as needed from a cloud
- Various providers allow creating virtual servers:
  - Choose the OS and software each instance will have
  - The chosen OS will run on a large server farm
  - Can instantiate more virtual servers or shut down existing ones within minutes
- You get billed only for what you used

# *Cloud Computing Agenda*

- What is Cloud Computing ?
  - Different perspectives
  - Properties and characteristics
  - Benefits from cloud computing
- Service and deployment models
  - Three service models
  - Four deployment models



Choose the service you need.

## ***SERVICE MODELS***

# A Simple Analogy

Say, you just moved to a city  
and you are looking for a  
place to live.



# What is your choice ?

Built a new house ?  
Buy an empty house ?  
Live in a hotel ?



# Let's built a new house !!

You can fully control everything you like your new house to have. But that is a hard work ...



# If you buy an empty house ?



You can customize some part of  
your house. But never change  
the original architecture.

# How about live in a hotel ?

Live in a hotel will be a good idea if the only thing you care is enjoy your life!! There is nothing you can do with the house except living in it.

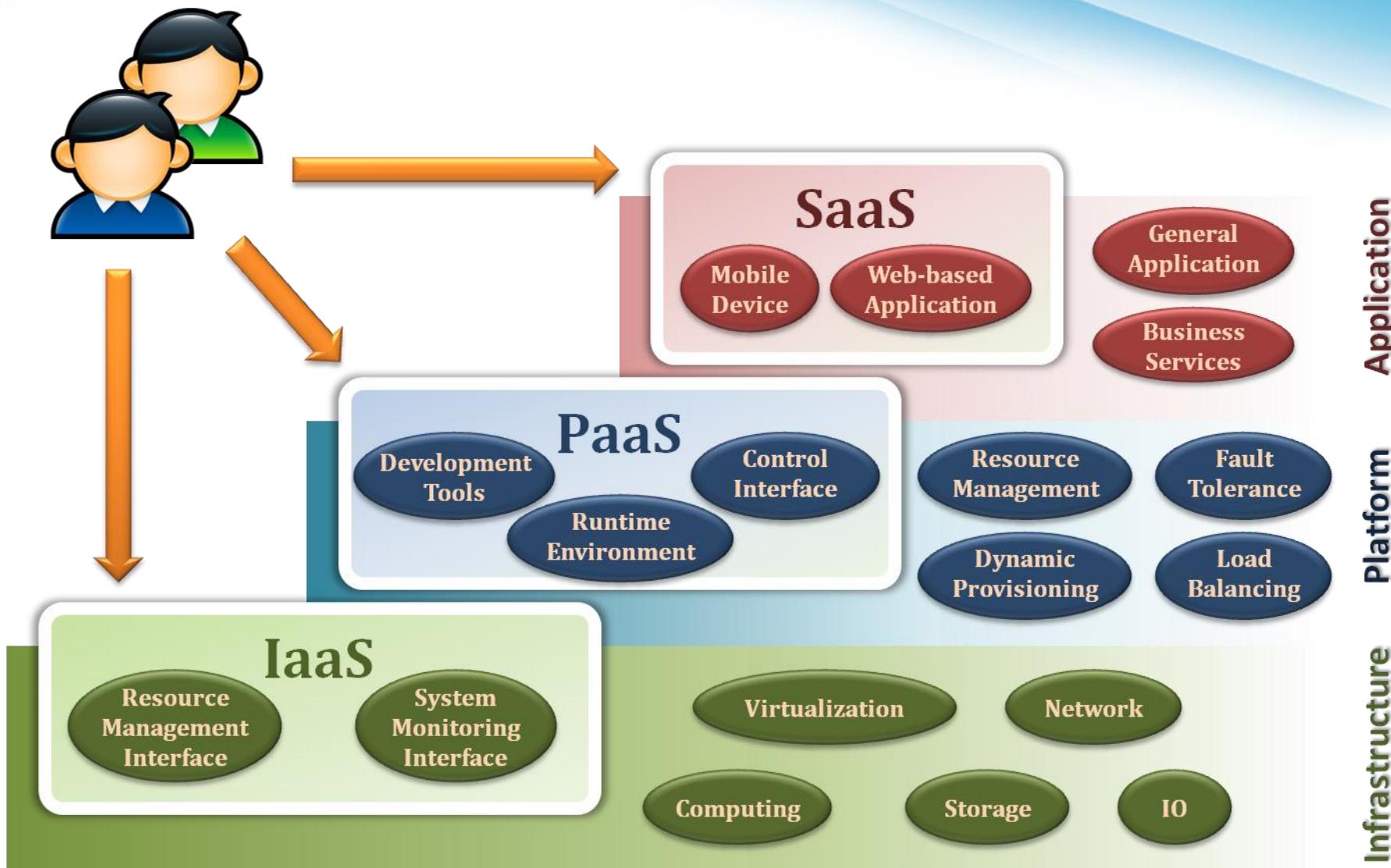


**Let's translate to  
Cloud Computing !!**

# *Service Models Overview*

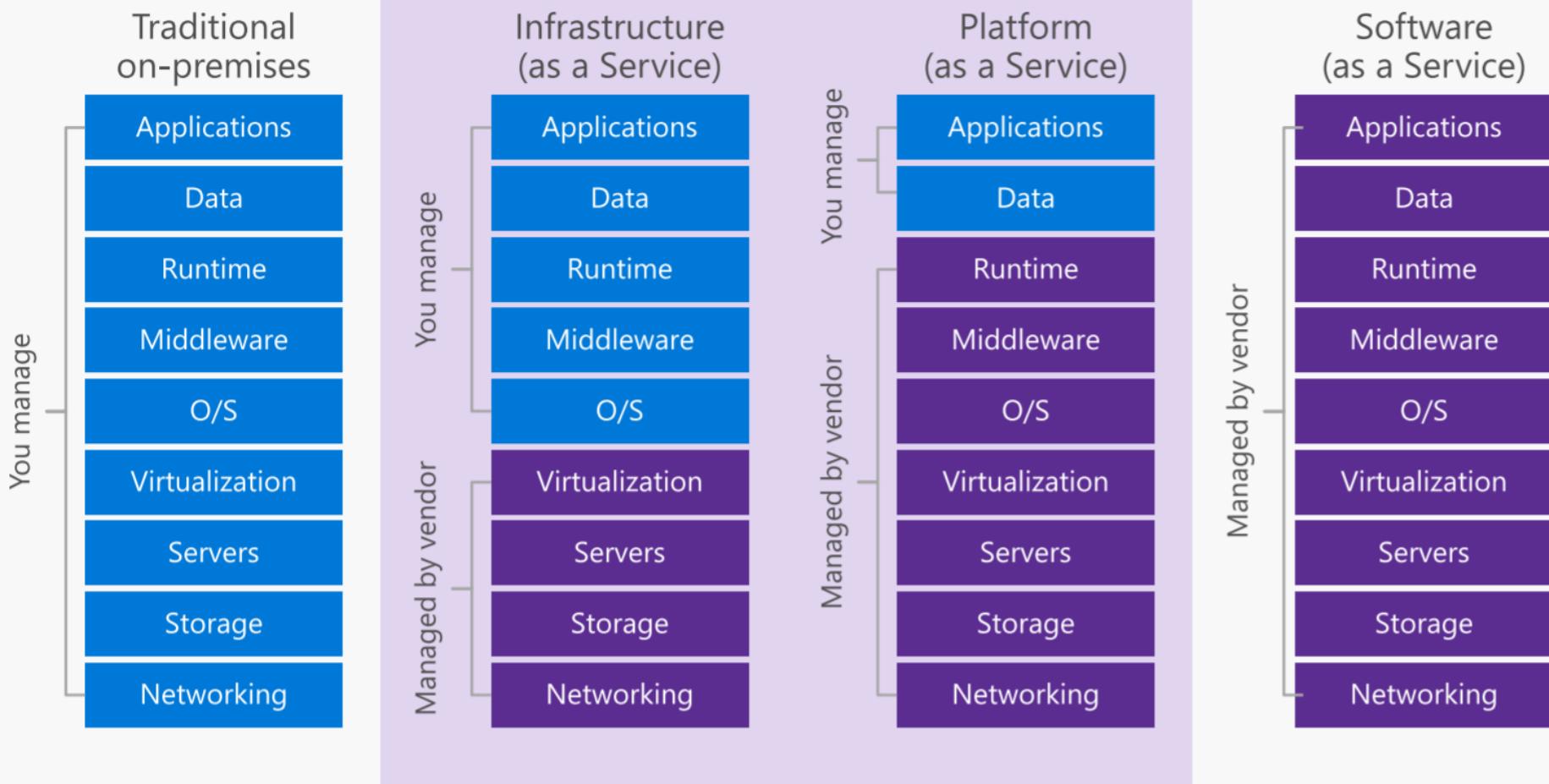
- What if you want to have an IT department ?
  - Similar to ***build a new house*** in previous analogy
    - You can rent some virtualized infrastructure and build up your own IT system among those resources, which may be fully controlled.
    - Technical speaking, use the ***Infrastructure as a Service (IaaS)*** solution.
  - Similar to ***buy an empty house*** in previous analogy
    - You can directly develop your IT system through one cloud platform, and do not care about any lower level resource management.
    - Technical speaking, use the ***Platform as a Service (PaaS)*** solution.
  - Similar to ***live in a hotel*** in previous analogy
    - You can directly use some existed IT system solutions, which were provided by some cloud application service provider, without knowing any detail technique about how these service was achieved.
    - Technical speaking, use the ***Software as a Service (SaaS)*** solution.

# Service Model Overview



# *Cloud Service Models*

# Cloud service models



Infrastructure as a Service

Platform as a Service

Software as a Service

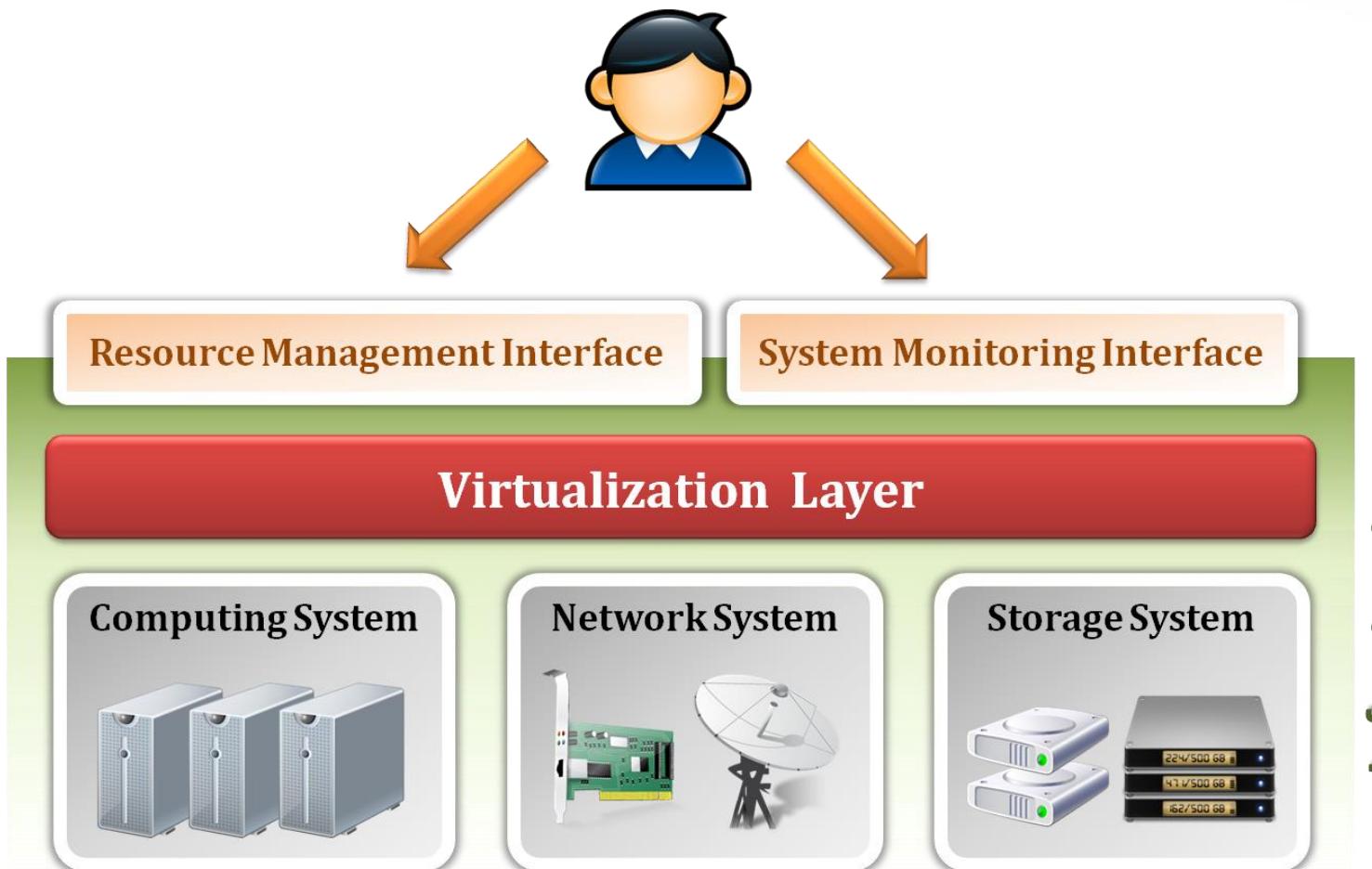
## ***SERVICE MODELS***

# *Infrastructure as a Service*

- Infrastructure as a Service - IaaS
  - The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications.
  - The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, deployed applications, and possibly limited control of select networking components .
- Examples :
  - Amazon EC2
  - Eucalyptus
  - OpenNebula
  - ... etc

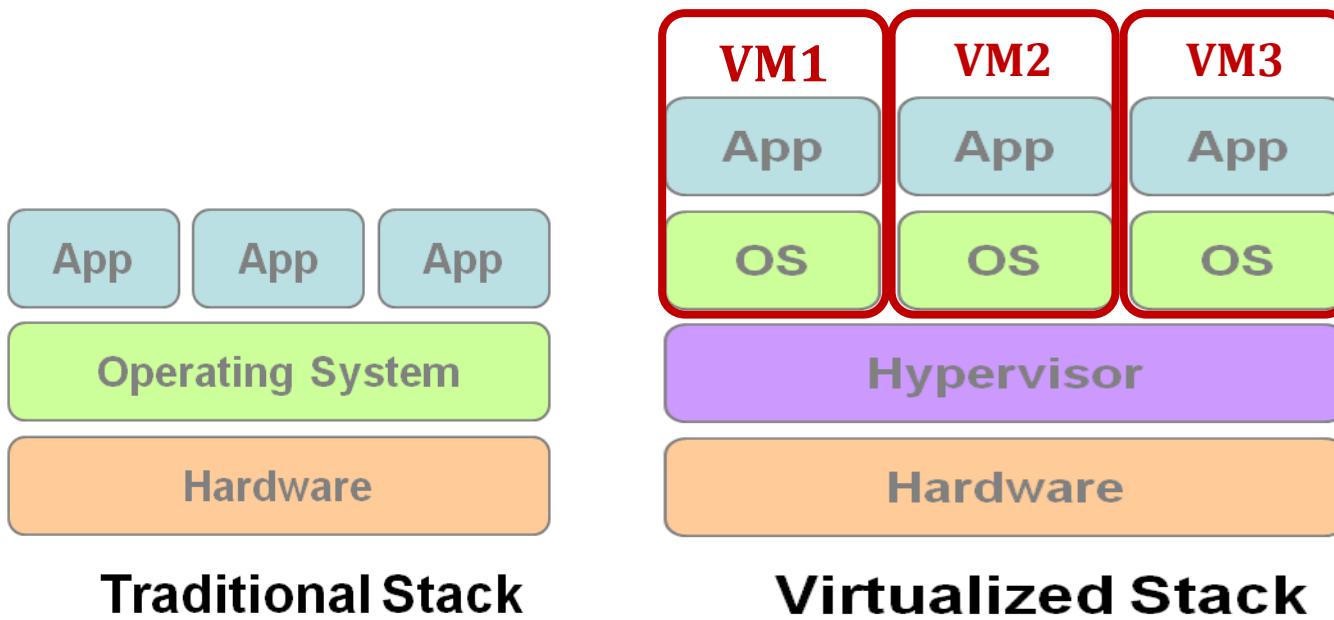
# *Infrastructure as a Service*

- System architecture :



# *Infrastructure as a Service*

- Enabling technique - ***Virtualization***
  - Virtualization is an abstraction of logical resources away from underlying physical resources.
    - Virtualization technique shift OS onto hypervisor.
    - Multiple OS share the physical hardware and provide different services.
    - Improve utilization, availability, security and convenience.



# *Infrastructure as a Service*

- Properties supported by virtualization technique :
  - Manageability and Interoperability
  - Availability and Reliability
  - Scalability and Elasticity



# *Infrastructure as a Service*

- Provide service –**Resource Management Interface**
  - Several types of virtualized resource :
    - ***Virtual Machine*** – As an IaaS provider, we should be able to provide the basic virtual machine operations, such as *creation, suspension, resumption* and *termination*, ...etc.
    - ***Virtual Storage*** – As an IaaS provider, we should be able to provide the basic virtual storage operations, such as *space allocation, space release, data writing* and *data reading*, ...etc.
    - ***Virtual Network*** – As an IaaS provider, we should be able to provide the basic virtual network operations, such as *IP address allocation, domain name register, connection establishment* and *bandwidth provision*, ...etc.

# *Infrastructure as a Service*

- Provide service – **System Monitoring Interface**
  - Several types of monitoring metrics :
    - **Virtual Machine** – As an IaaS provider, we should be able to monitor some system states of each virtual machine, such as *CPU loading, memory utilization, IO loading* and *internal network loading*, ...etc.
    - **Virtual Storage** – As an IaaS provider, we should be able to monitor some storage states of each virtual storage, such as *virtual space utilization, data duplication* and *storage device access bandwidth*, ...etc.
    - **Virtual Network** – As an IaaS provider, we should be able to monitor some network states of each virtual network, such as *virtual network bandwidth, network connectivity* and *network load balancing*, ...etc.

# *IaaS - Summary*

- IaaS is the deployment platform that abstract the infrastructure.
- IaaS enabling technique
  - Virtualization
    - Server Virtualization
    - Storage Virtualization
    - Network Virtualization
- IaaS provided services
  - Resource Management Interface
  - System Monitoring Interface

Infrastructure as a Service

**Platform as a Service**

Software as a Service

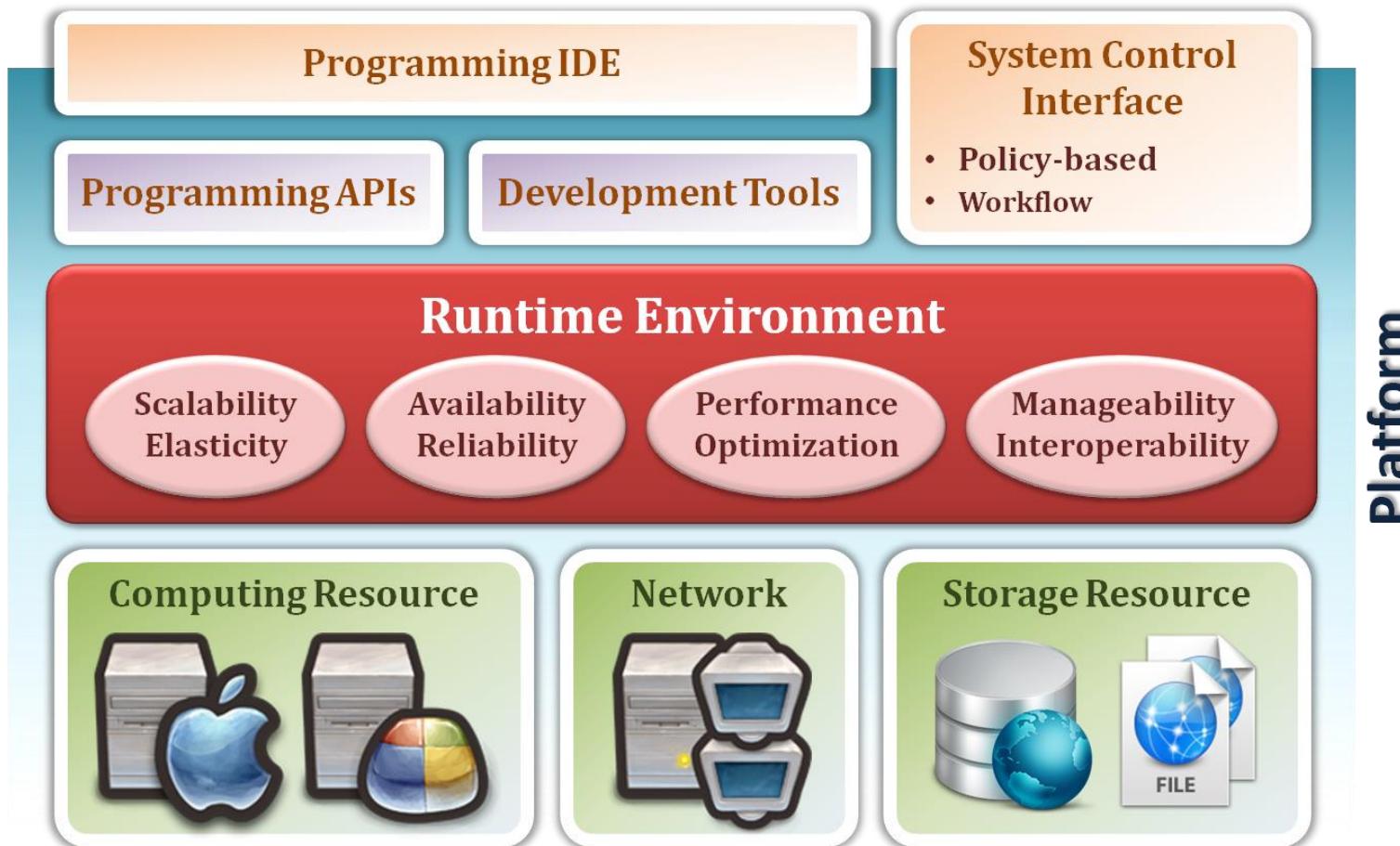
## ***SERVICE MODELS***

# *Platform as a Service*

- Platform as a Service - PaaS
  - The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages and tools supported by the provider.
  - The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly application hosting environment configurations.
- Examples :
  - Microsoft Windows Azure
  - Google App Engine
  - Hadoop
  - ... etc

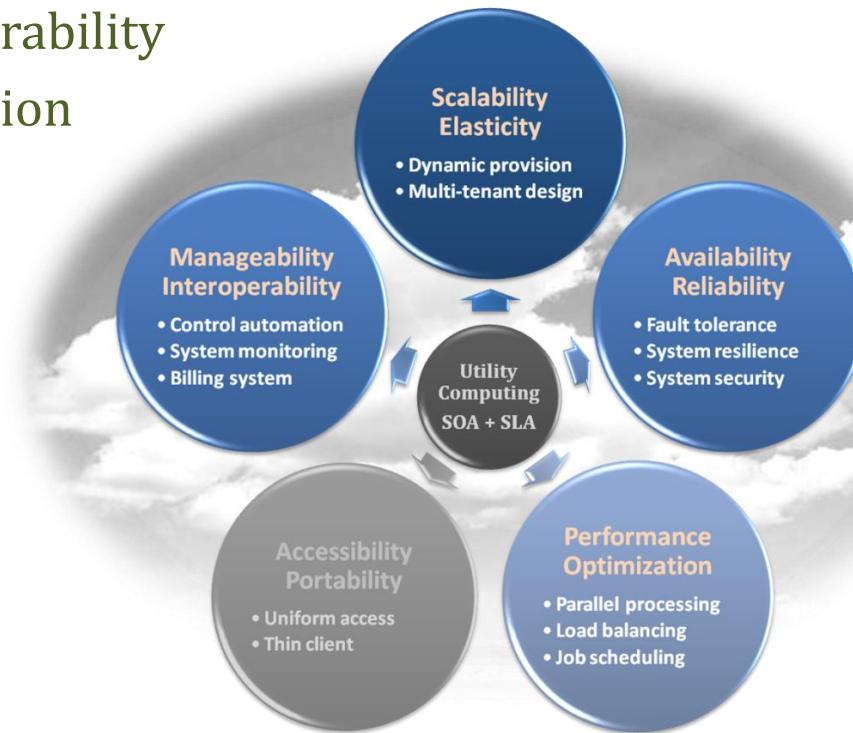
# *Platform as a Service*

- System architecture :



# *Platform as a Service*

- Enabling technique – **Runtime Environment Design**
  - Runtime environment refers to collection of software services available. Usually implemented by a collection of program libraries.
- Common properties in Runtime Environment :
  - Manageability and Interoperability
  - Performance and Optimization
  - Availability and Reliability
  - Scalability and Elasticity



# *PaaS - Summary*

- PaaS is the development platform that abstract the infrastructure, OS, and middleware to drive developer productivity.
- PaaS enabling technique
  - Runtime Environment
- PaaS provide services
  - Programming IDE
    - Programming APIs
    - Development tools
  - System Control Interface
    - Policy based approach
    - Workflow based approach

Infrastructure as a Service

Platform as a Service

**Software as a Service**

## ***SERVICE MODELS***

# *Software as a Service*

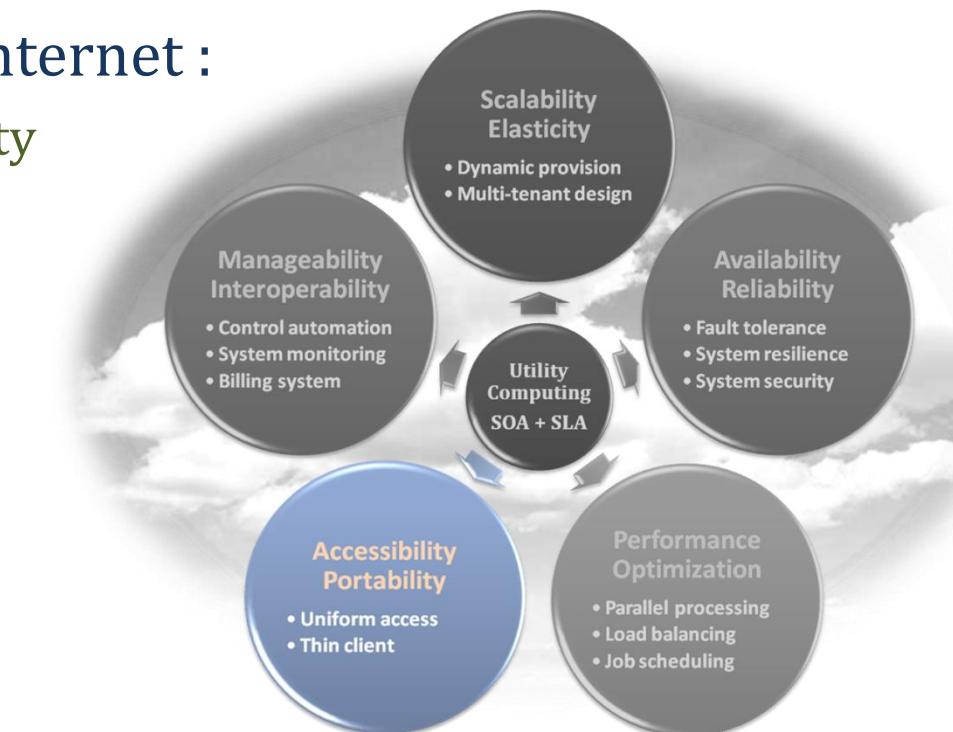
- Software as a Service - SaaS
  - The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based email).
  - The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.
- Examples :
  - Google Apps (e.g., Gmail, Google Docs, Google sites, ...etc)
  - SalesForce.com
  - EyeOS
  - ... etc

# *Software as a Service*



# *Software as a Service*

- Enabling Technique – **Web Service**
  - Web 2.0 is the trend of using the full potential of the web
    - Viewing the Internet as a computing platform
    - Running interactive applications through a web browser
    - Leveraging interconnectivity and mobility of devices
    - Enhanced effectiveness with greater human participation
- Properties provided by Internet :
  - Accessibility and Portability



# *Software as a Service*

- Provide service – **Web-based Applications**
  - Conventional applications should translate their access interface onto web-based platform.
  - Applications in different domains
    - **General Applications** – Applications which are designed for general propose, such as *office suit*, *multimedia* and *instant message*, ...etc.
    - **Business Applications** – Application which are designed for business propose, such as *ERP*, *CRM* and *market trading system*, ...etc.
    - **Scientific Applications** – Application which are designed for scientific propose, such as *aerospace simulation* and *biochemistry simulation*, ...etc.
    - **Government Applications** – Applications which are designed for government propose, such as *national medical system* and *public transportation system service*, ...etc.

# *SaaS - Summary*

- **SaaS is the finished applications that you rent and customize.**
- SaaS enabling technique
  - Web Service
- SaaS provide services
  - Web-based Applications
    - General applications
    - Business applications
    - Scientific applications
    - Government applications
  - Web Portal

How to deploy a cloud system ?

## ***DEPLOYMENT MODELS***



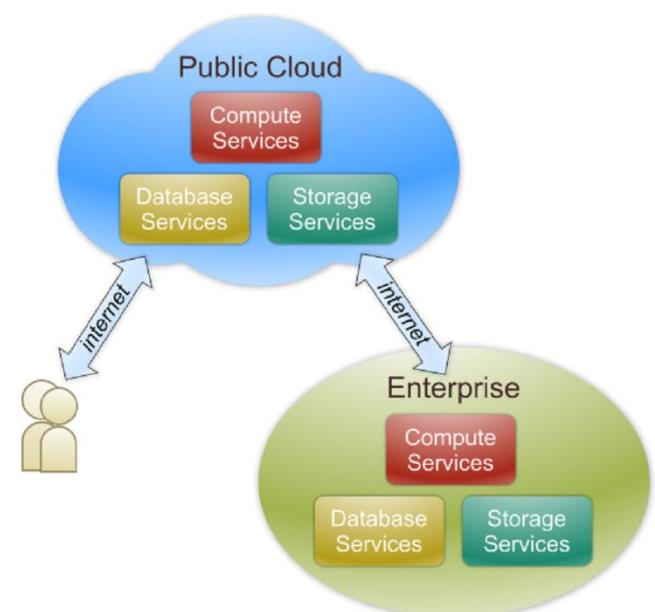
# *Deployment Model*

- There are four primary cloud deployment models :
  - Public Cloud
  - Private Cloud
  - Community Cloud
  - Hybrid Cloud
- Each can exhibit the previously discussed characteristics; their differences lie primarily in the scope and access of published cloud services, as they are made available to service consumers.

# Public Cloud

- Public cloud definition

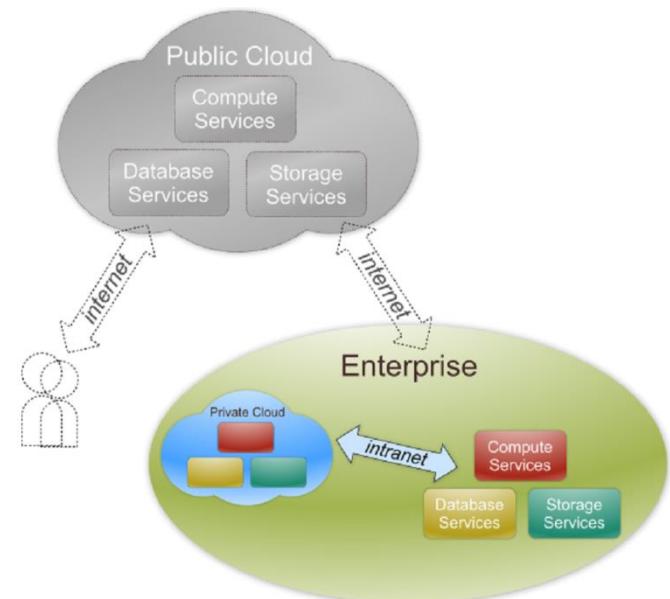
- The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.
- Also known as external cloud or multi-tenant cloud, this model essentially represents a cloud environment that is openly accessible.
- Basic characteristics :
  - Homogeneous infrastructure
  - Common policies
  - Shared resources and multi-tenant
  - Leased or rented infrastructure
  - Economies of scale



# *Private Cloud*

- Private cloud definition

- The cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on premise or off premise.
- Also referred to as internal cloud or on-premise cloud, a private cloud intentionally limits access to its resources to service consumers that belong to the same organization that owns the cloud.
- Basic characteristics :
  - Heterogeneous infrastructure
  - Customized and tailored policies
  - Dedicated resources
  - In-house infrastructure
  - End-to-end control



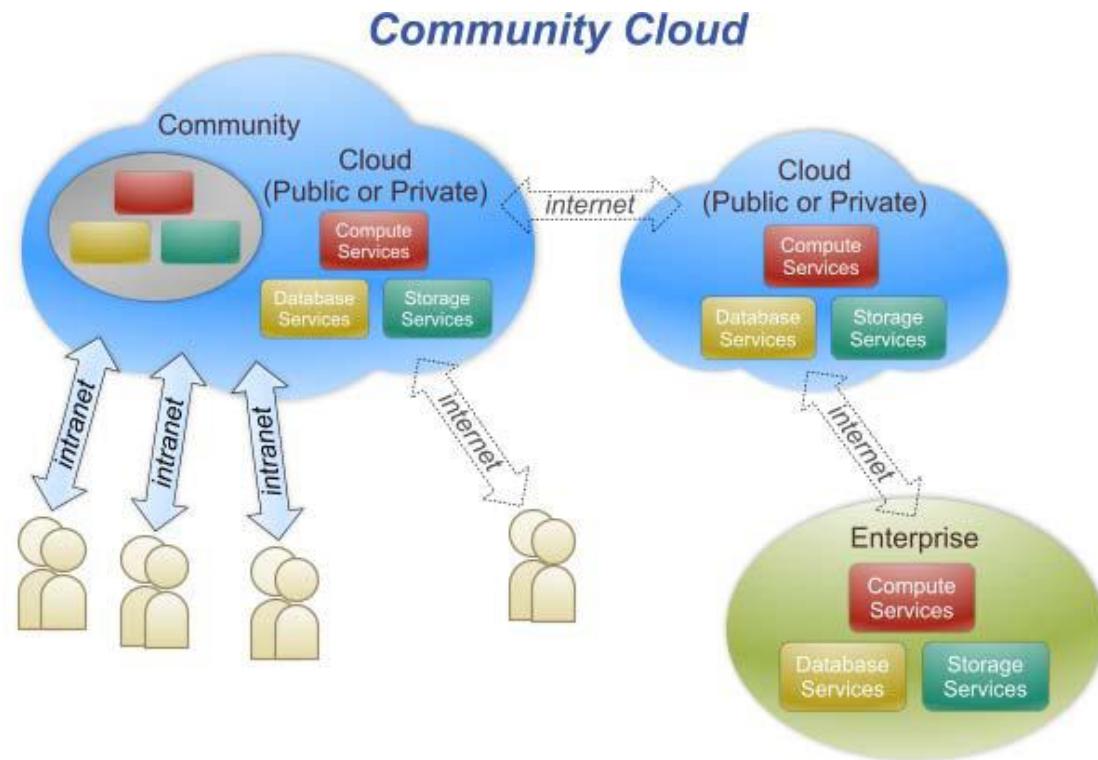
# *Public vs. Private*

- Comparison :

	<b>Public Cloud</b>	<b>Private Cloud</b>
<b>Infrastructure</b>	<i>Homogeneous</i>	<i>Heterogeneous</i>
<b>Policy Model</b>	<i>Common defined</i>	<i>Customized &amp; Tailored</i>
<b>Resource Model</b>	<i>Shared &amp; Multi-tenant</i>	<i>Dedicated</i>
<b>Cost Model</b>	<i>Operational expenditure</i>	<i>Capital expenditure</i>
<b>Economy Model</b>	<i>Large economy of scale</i>	<i>End-to-end control</i>

# *Community Cloud*

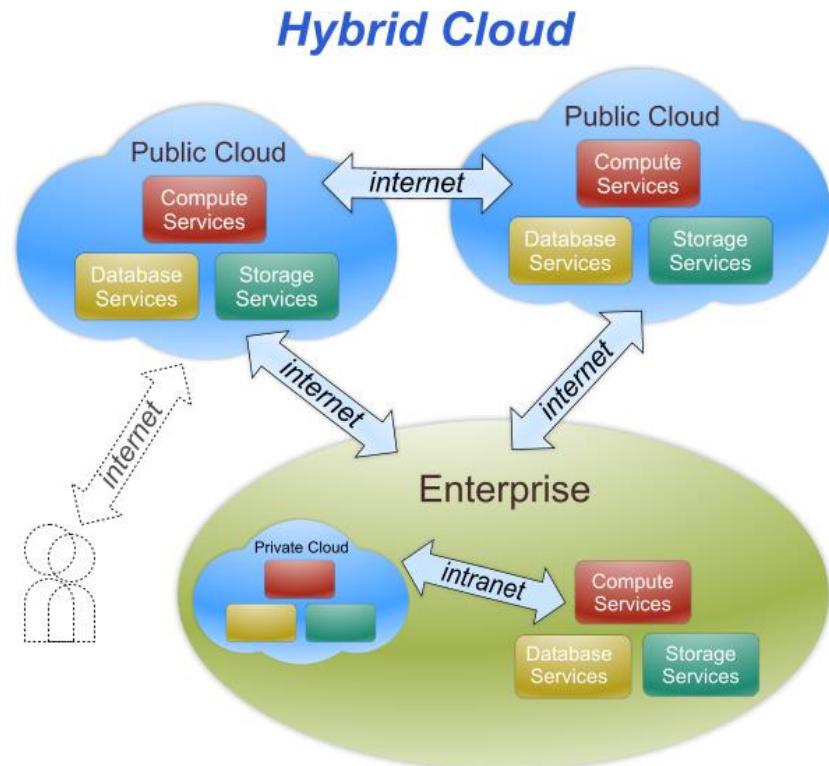
- Community cloud definition
  - The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations).



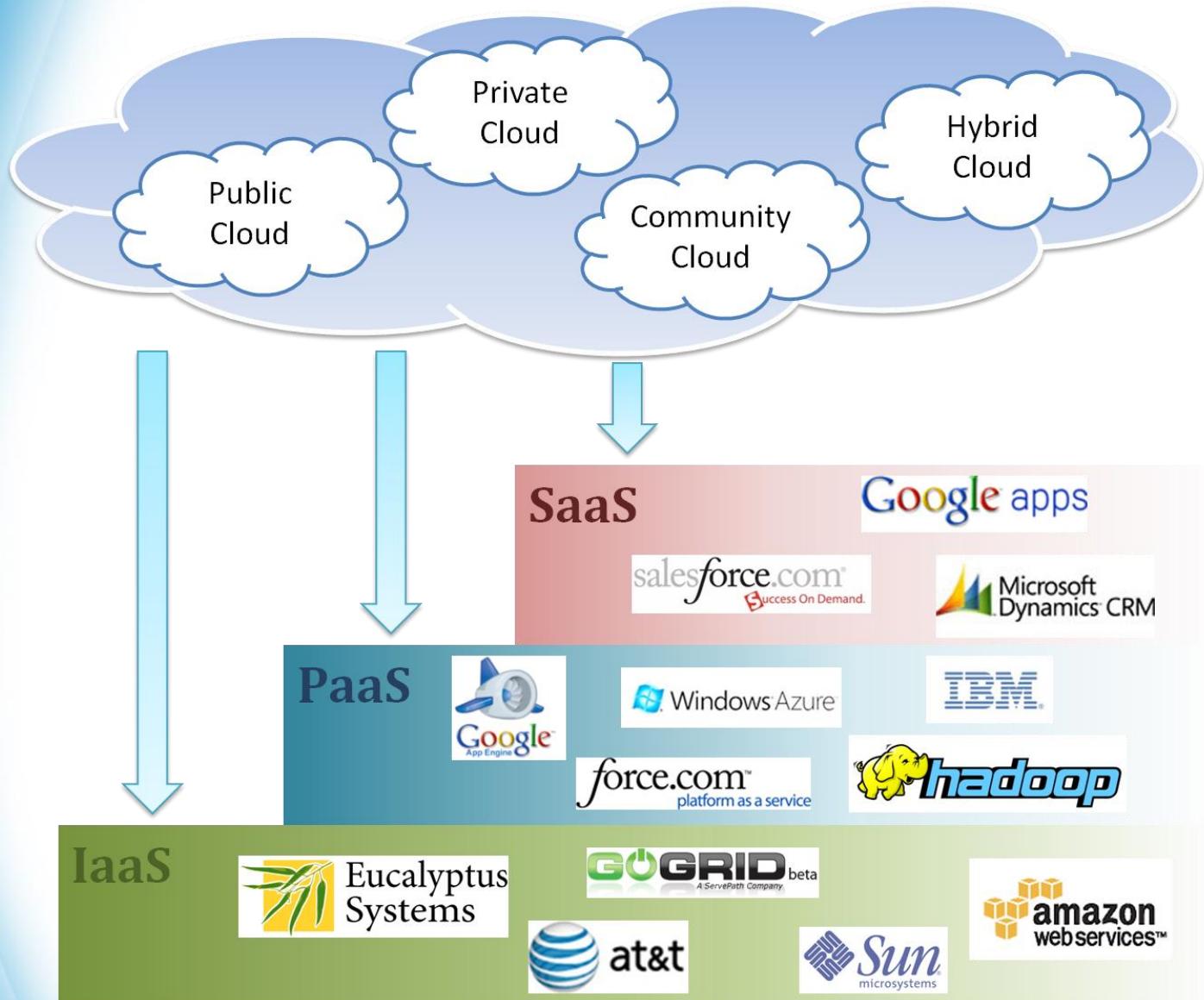
# *Hybrid Cloud*

- Hybrid cloud definition

- The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load-balancing between clouds).



# *Cloud Ecosystem*



# *Summary*

- What is cloud computing in your mind
  - Clear or Cloudy?
- Cloud computing is a new paradigm shift of computing
- Cloud computing can provide high quality of properties and characteristics based on essentially central ideas
- Service models and deployment models provide services that can be used to
  - Rent fundamental computing resources
  - Deploy and develop customer-created applications on clouds
  - Access provider's applications over network (wired or wireless)

# References

- NIST (National Institute of Standards and Technology).  
<http://csrc.nist.gov/groups/SNS/cloud-computing/>
- M. Armbrust et. al., "Above the Clouds: A Berkeley View of Cloud Computing," Technical Report No. UCB/EECS-2009-28, University of California at Berkeley, 2009.
- R. Buyya et. al., "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation Computer Systems, 2009.
- Cloud Computing Use Cases.  
<http://groups.google.com/group/cloud-computing-use-cases>
- Cloud Computing Explained.  
<http://www.andyharjanto.com/2009/11/wanted-cloud-computing-explained-in.html>
- From Wikipedia, the free encyclopedia
- All resources of the materials and pictures were partially retrieved from the Internet.