

# La Virtualizzazione nel Dispiegamento di Sistemi Complessi

Prof. Sergio Di Martino



# La virtualizzazione

- La virtualizzazione è una tecnica per aumentare l'efficacia e l'efficienza di Centri Elaborazione Dati complessi
- Tecnica risalente agli anni '60, ma ritornata in auge una decina di anni fa, grazie all'evoluzione tecnologica delle CPU

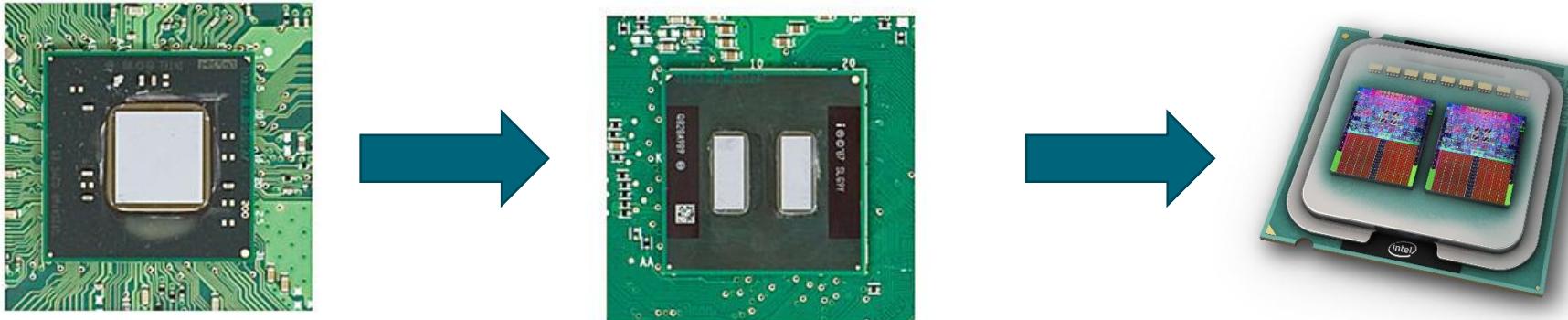
# L'evoluzione delle CPU

# Evoluzione delle CPU

- I microprocessori hanno da sempre avuto un rapidissimo incremento nelle prestazioni e una diminuzione dei costi.
- La strategia per migliorare le prestazioni è storicamente stata quella di aumentare la "velocità" di un processore.
  - 1978: Intel 8086, 4MHz
  - 1985: Intel 386, 25 MHz (~ 6x)
  - 1995: Intel Pentium Pro, 200 MHz (~ 8x)
  - 2005: Intel Pentium P4 3,8 GHz (3800 MHz) (~ 19x)
- Questa strategia ha subito una battuta d'arresto nel 2005
  - 2020: Intel i9-10980XE 3,0 GHz

# Le CPU multicore

- Raggiunti i limiti fisici del silicio, la soluzione è stata aumentare il numero di unità di processo (multi core), allo scopo di aumentare la potenza di calcolo.
  - 2020: Intel i9-10980XE 3,0 GHz → 18 unita di calcolo
  - 2020: AMD Ryzen™ Threadripper™ 3970X 3,7 GHz → 32 unita di calcolo



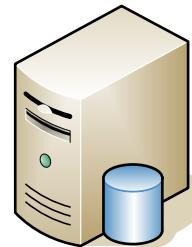
# Problemi dei Multi-core

- Tradizionalmente i programmi sono stati scritti per essere eseguiti su un computer con una singola CPU.
- La stragrande maggioranza delle applicazioni sono costituite da programmi sequenziali.
- Come sfruttare più unità in parallelo?

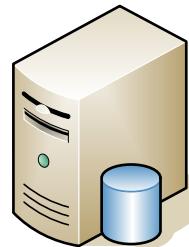
# **Infrastruttura tradizionale di un CED**

# Infrastruttura Tradizionale di CED

- Singoli server che offrono singoli servizi e/o applicazioni (*one workload, one box*)
- Nuovo Servizio e/o Applicazione → Nuovo Server, con proprio Storage



Server Application



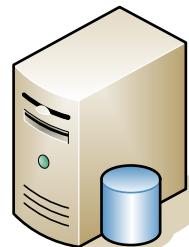
Server di Dominio



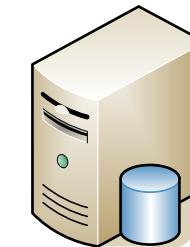
Server Database



Server di Posta



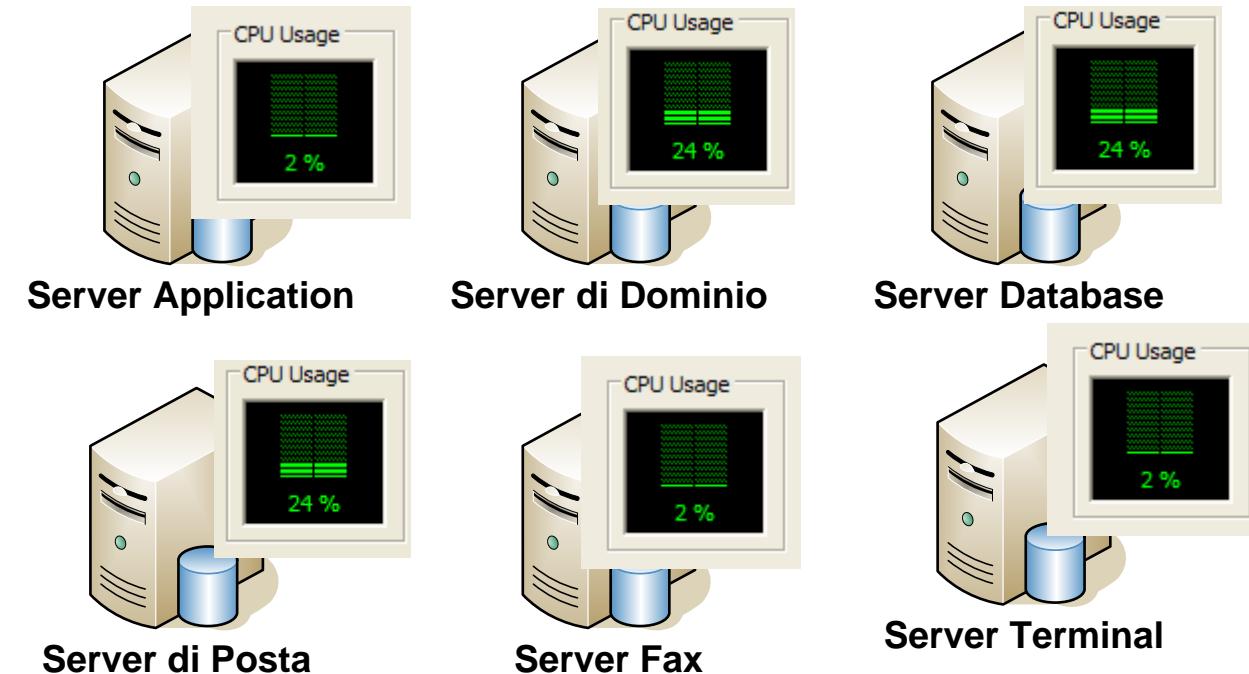
Server Fax



Server Terminal

# Infrastruttura Tradizionale di CED

- Tutti i server, consumano (tanta) corrente, emettono calore, occupano spazio, vanno gestiti fisicamente...

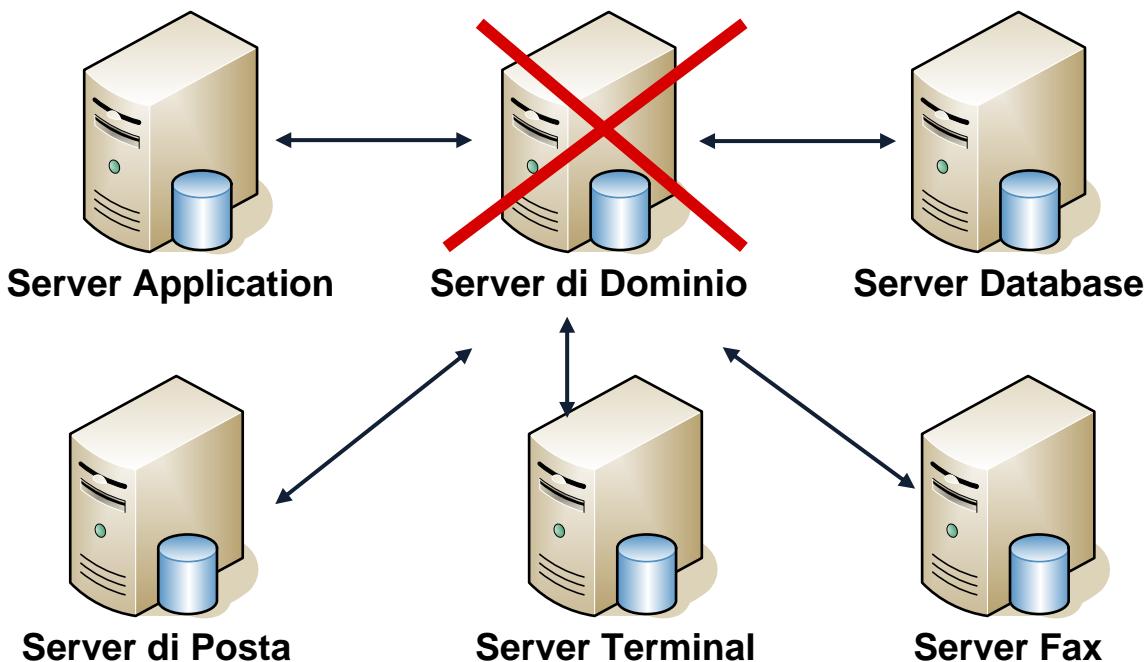


# Infrastruttura Tradizionale di CED: Conseguenze

- Sottoutilizzo delle risorse
  - La maggior parte dei server sfrutta in media tra il 5% ed il 15% della propria capacità computazionale.
  - In caso di servizi critici, necessità di ridondanza (computer di riserva normalmente non usati).
- Aumento dei costi di installazione e gestione
  - Rilascio aggiornamenti su scala nazionale molto costosi
  - Back-up da effettuare su ogni macchina
  - ...

# Infrastruttura Tradizionale di CED : Conseguenze

- I servizi sono intrinsecamente interconnessi.
  - Es: Il server di dominio, che gestisce le autenticazioni degli utenti, viene interrogato da tutti gli altri servizi/server
  - Cosa succede se si rompe un server?



# Fail di un server nello scenario tradizionale

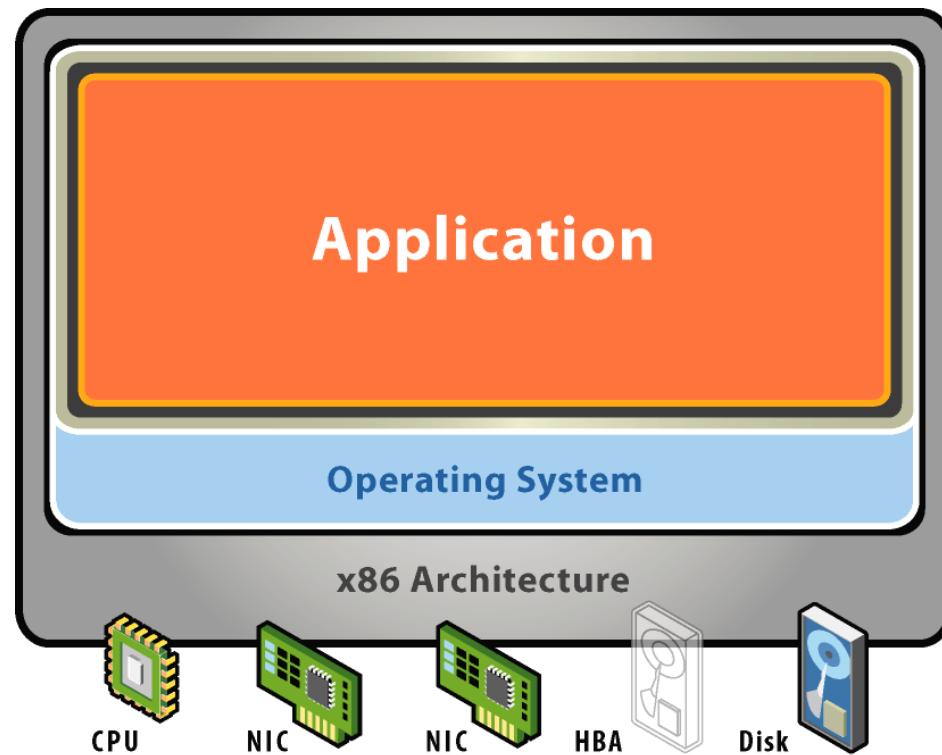
- In caso di FAIL di un server si possono bloccare i servizi di altri Server in comunicazione con esso.
- Ciò comporta:
  - Interruzione dell'operatività collegata a quel servizio
  - Nei casi più gravi interruzione dell'intera operatività aziendale
  - Aumento dei costi (costi di fermo e di ripristino)
  - Tempi di fermo non sempre prevedibili

# Le Macchine virtuali

Concetti Generali

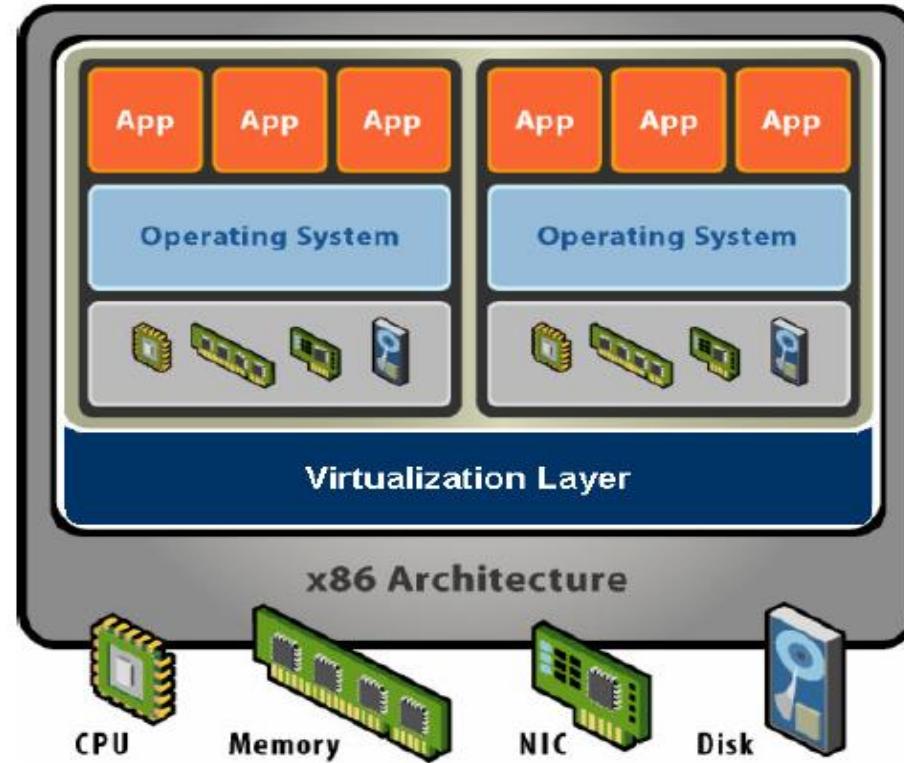
# Un Server Fisico

- Un Server fisico consiste di:
  - Una serie di risorse hardware
  - Un sistema operativo
  - Uno o più software applicativi
- **Idea:** se la CPU del Server ha più unità di calcolo parallele, possiamo "simulare" l'esistenza di più computer, indipendenti dal "vero" hw sottostante?



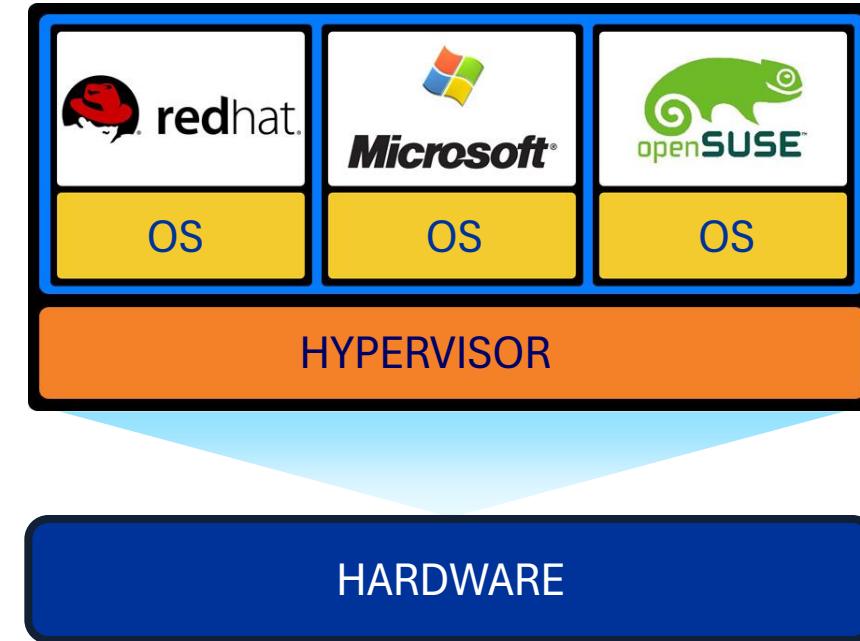
# Una Macchina Virtuale

- La virtualizzazione consiste nella creazione di una versione virtuale di una risorsa normalmente fornita fisicamente e appartenente a un sistema.
  - Migliore sfruttamento delle risorse e delle capacità di calcolo



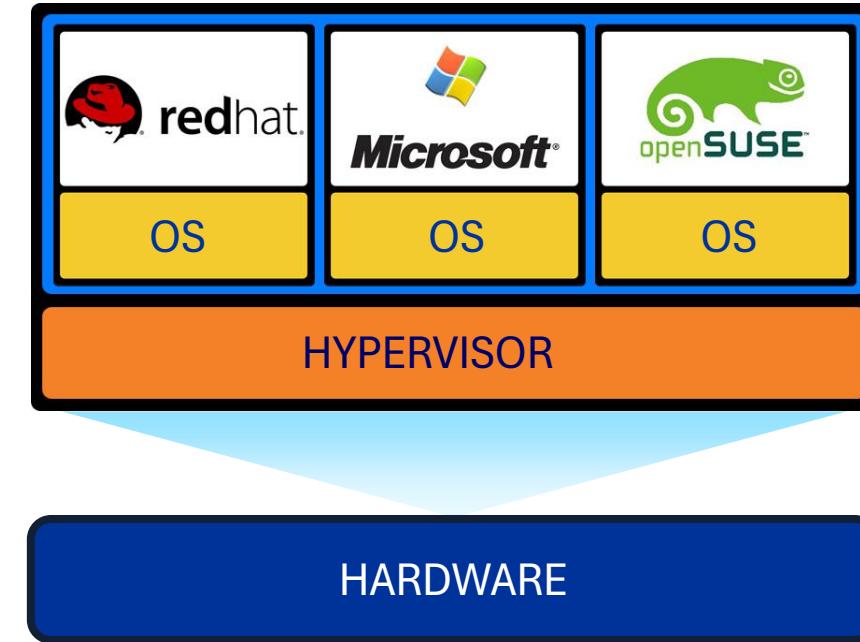
# Come si ottiene la Virtualizzazione?

- Si introduce un nuovo strato che parte prima del sistema operativo, chiamato **Hypervisor**.
- L'Hypervisor simula l'esistenza di più computer "virtuali" su uno fisico



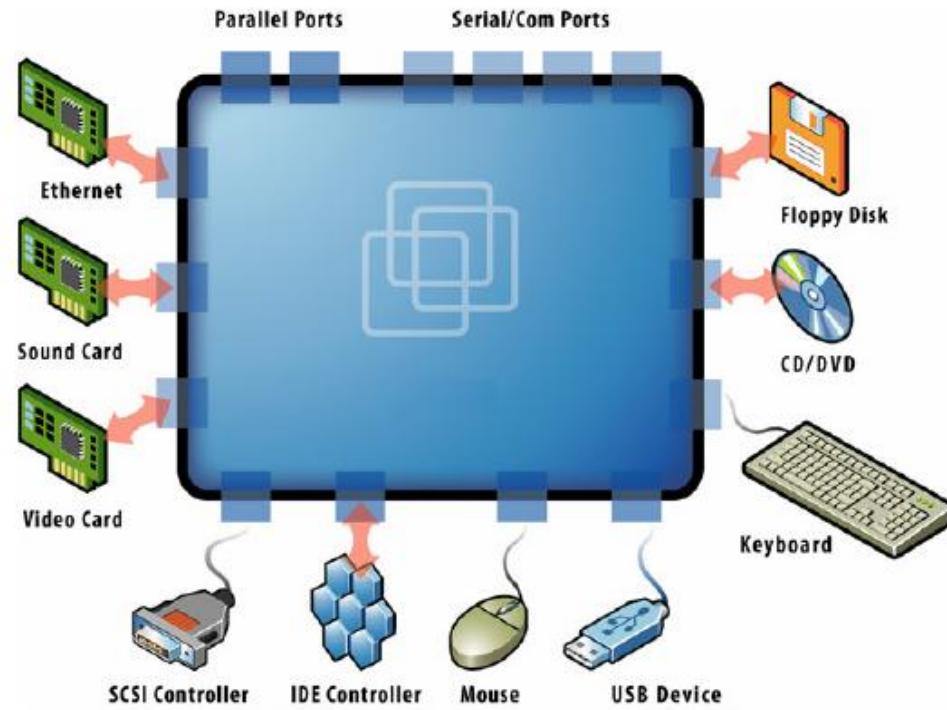
# Come si ottiene la Virtualizzazione?

- L'Hypervisor offre le risorse fisiche ai computer virtualizzati, e si occupa di gestirle in modo concorrente a favore delle macchine virtuali.
- In ogni computer virtualizzato, si installa un normalissimo sistema operativo, che non ha modo di sapere se sta funzionando su un computer "reale" o virtuale.



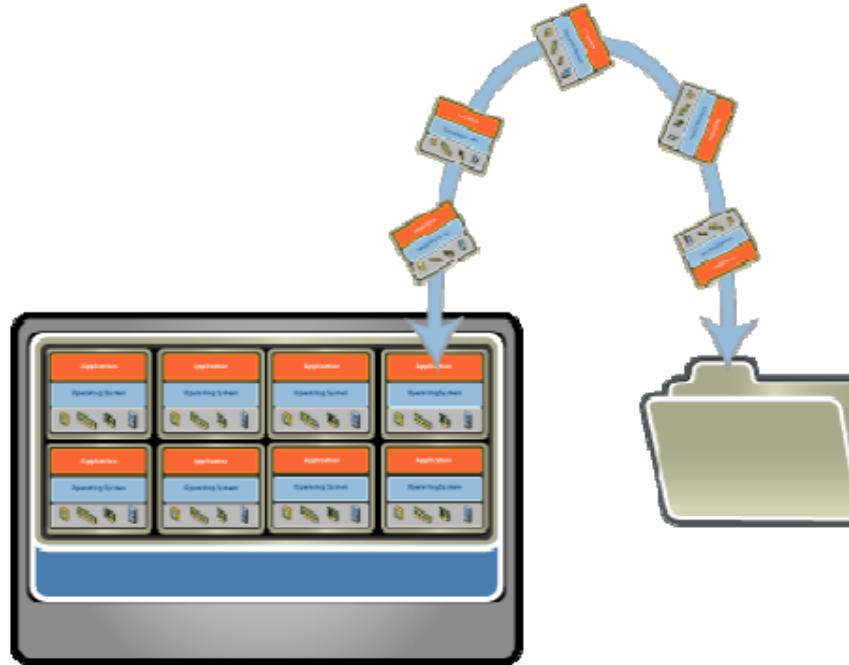
# Indipendenza dall'Hardware

- L'Hypervisor simula una configurazione standard di un server, nascondendo il vero hardware fisico
- Conseguenza: Le Virtual Machines possono funzionare su server fisici diversi senza bisogno di essere configurate



# Memorizzazione di una VM

- Una virtual machine è **un file**, contenente:
  - Sistema Operativo, Applicativi, Dati
  - Rappresentazione della Memoria del Server
- Un file memorizza lo stato di un server virtuale e ne permette la copia su un numero arbitrario di server fisici
- Esistono strumenti commerciali che trasformano il contenuto di un server fisico in una VM



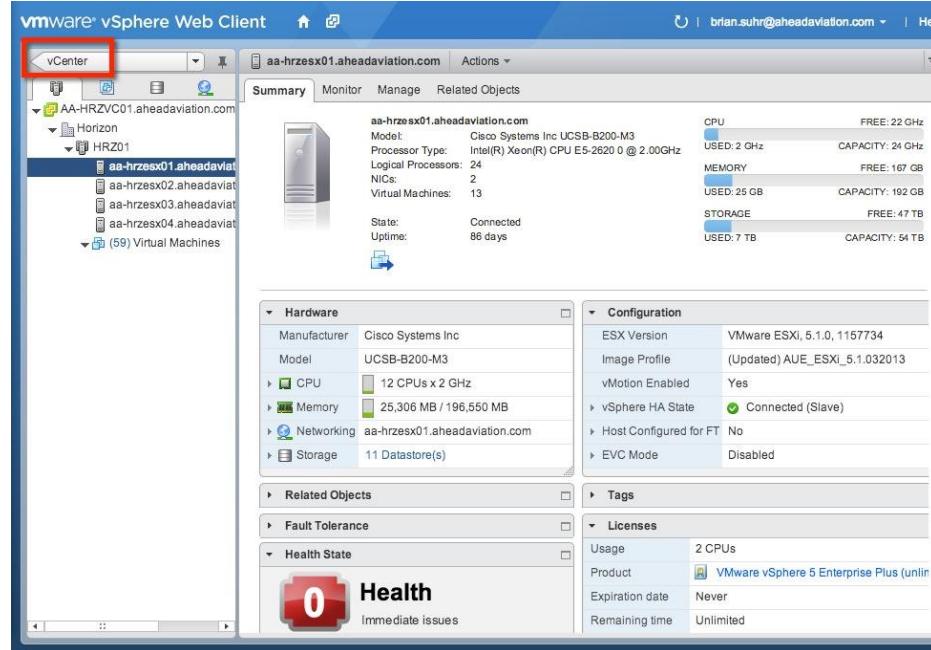
# Isolamento di una VM

- Ogni VM è totalmente isolata dalle altre VM in esecuzione sullo stesso server fisico
- In caso di attacco o di crash di una VM, le altre non sono intaccate



# Gestione di VM

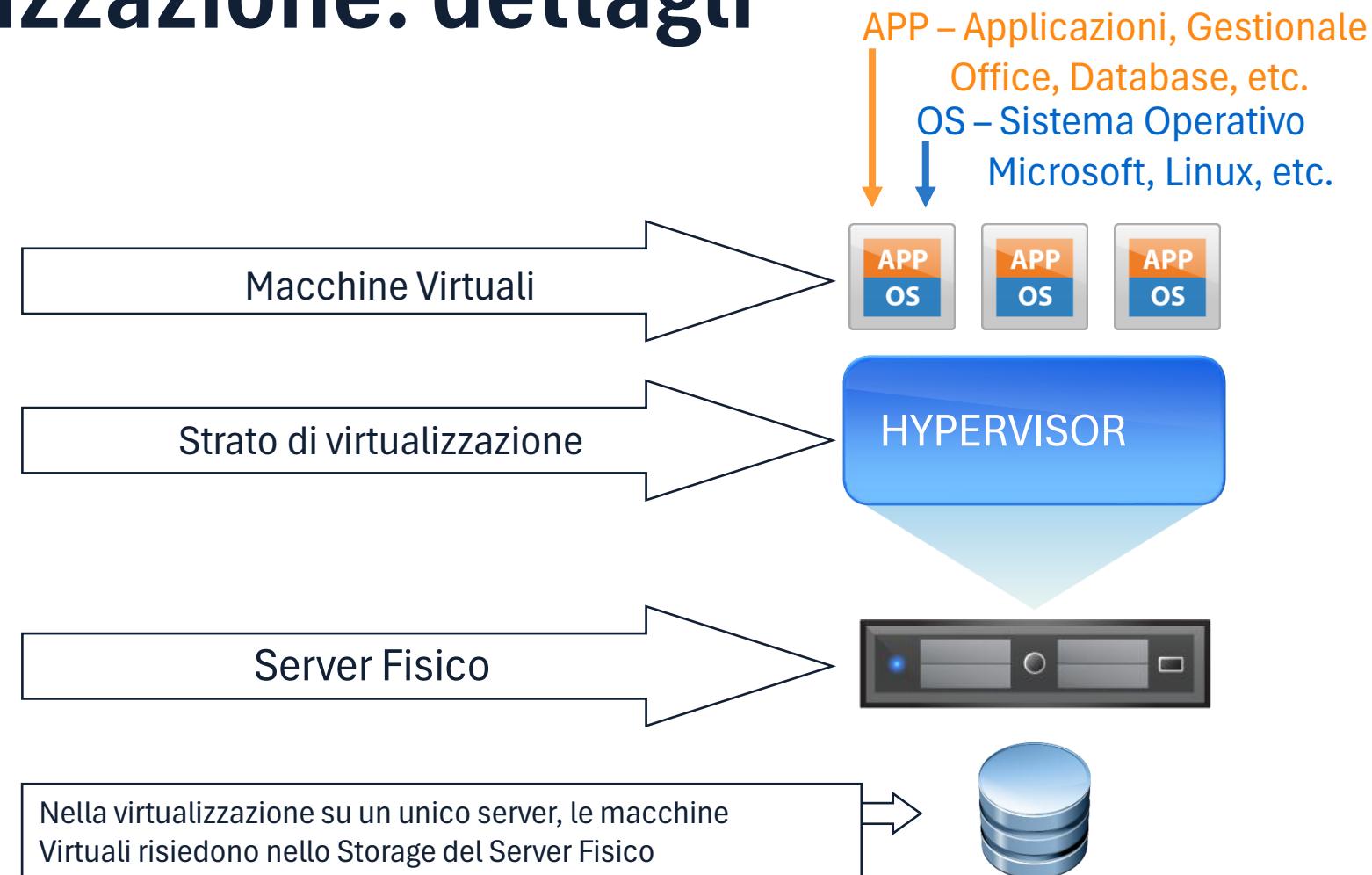
- Tutte le VM di un CED possono essere gestite attraverso un'unica finestra.
- La gestione dell'intero CED risulta centralizzata, con notevole semplificazione della gestione



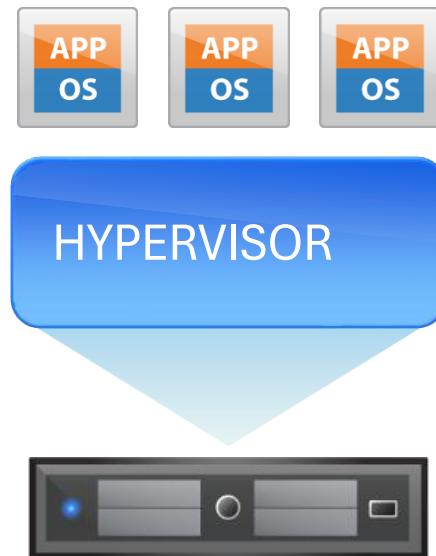
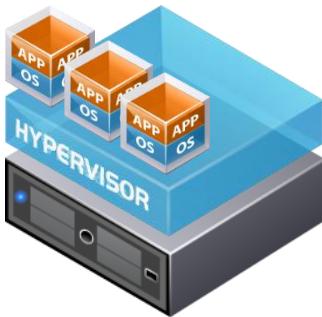
# Le Macchine virtuali

Dettagli Operativi

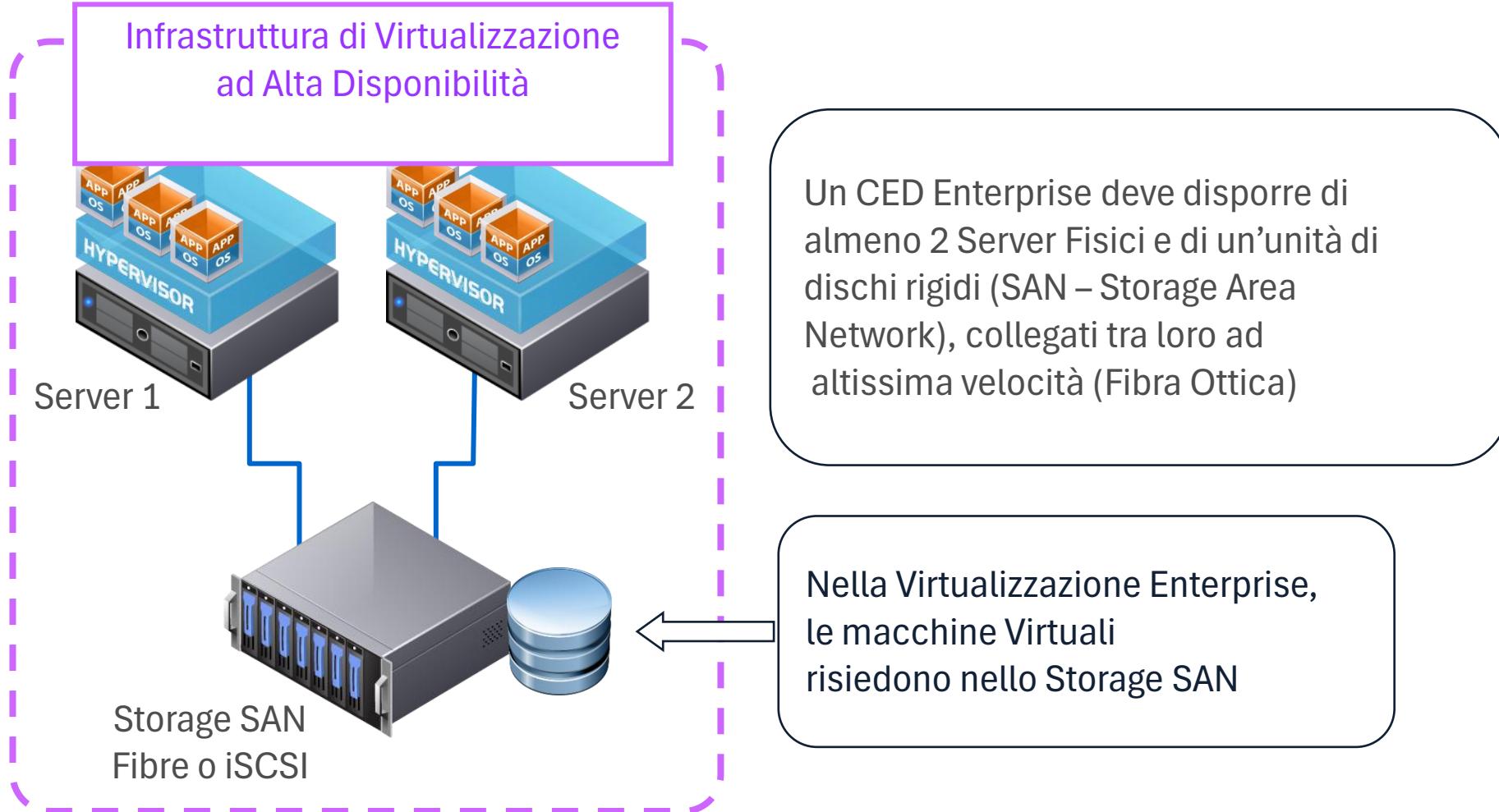
# Virtualizzazione: dettagli



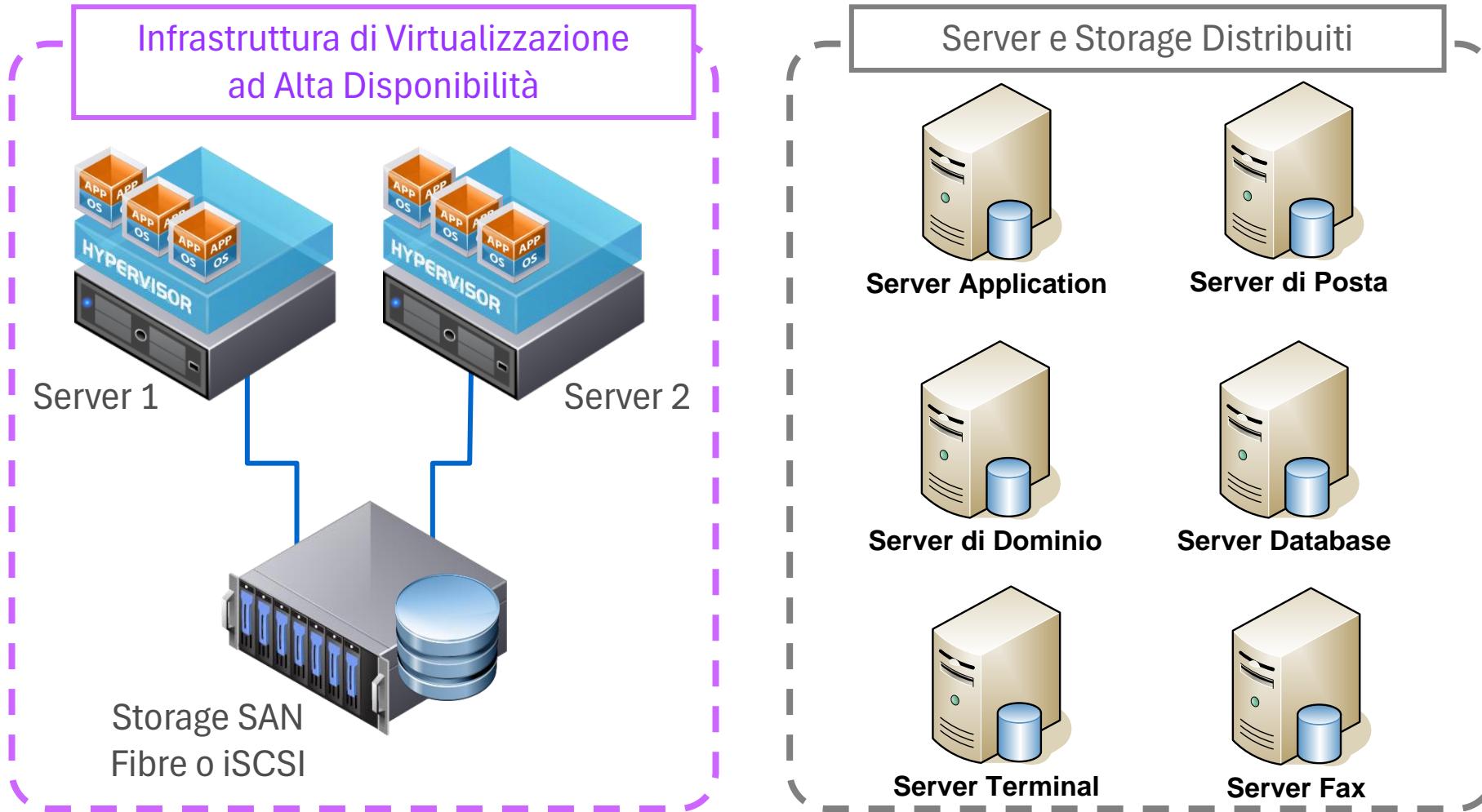
# Virtualizzazione Enterprise ad Alta Disponibilità



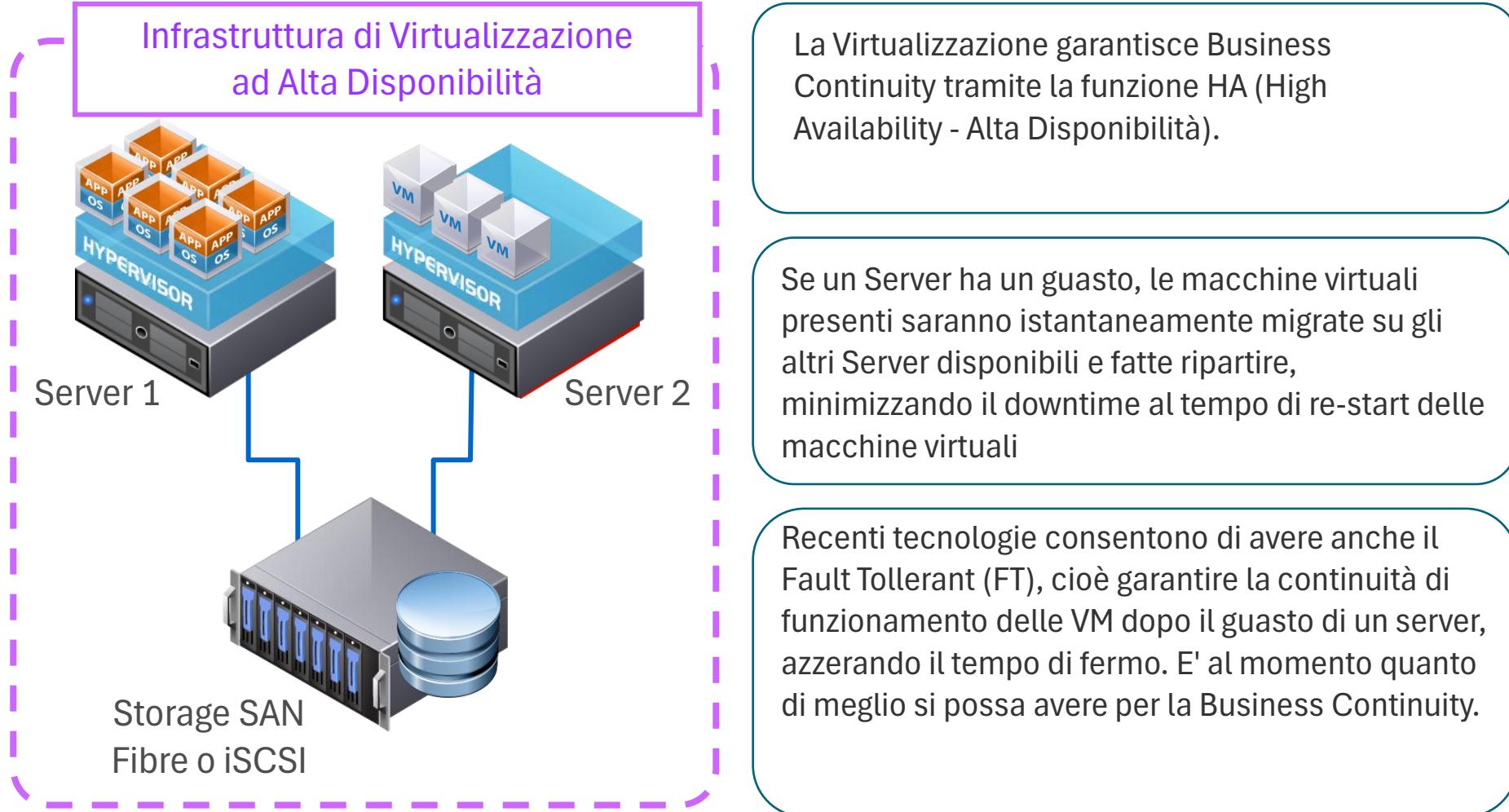
# Virtualizzazione Enterprise ad Alta Disponibilità



# Migrazione da Fisico a Virtuale

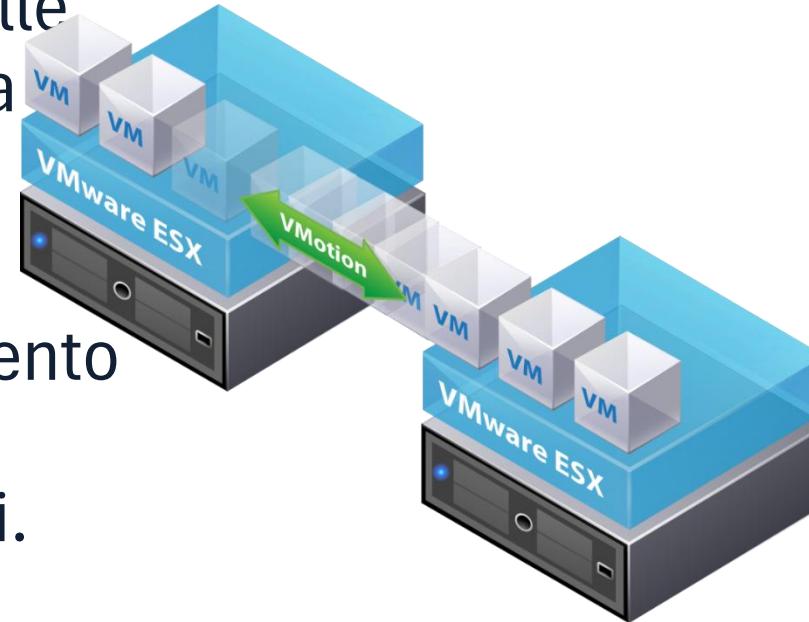


# Come Funziona un sistema ad Alta Disponibilità



# Virtualizzazione: La Migrazione Live delle VM

- La migrazione "a caldo" delle VM fra i vari Server Fisici è una delle tante funzionalità del sistema di virtualizzazione.
- Consente operazioni di manutenzione o di bilanciamento del carico mantenendo l'operatività dei server virtuali.

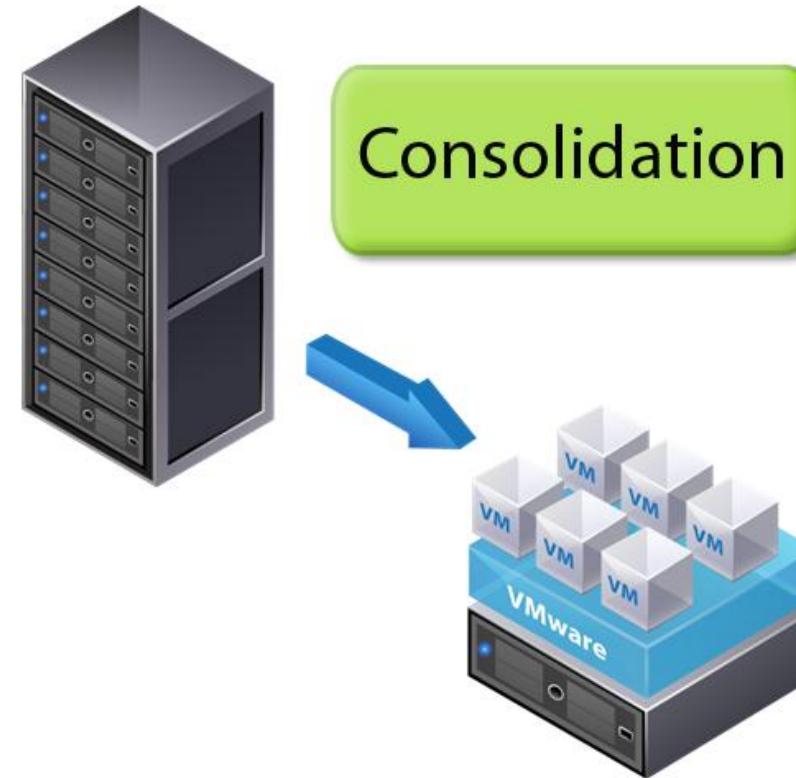


# Le Macchine virtuali

## Benefici

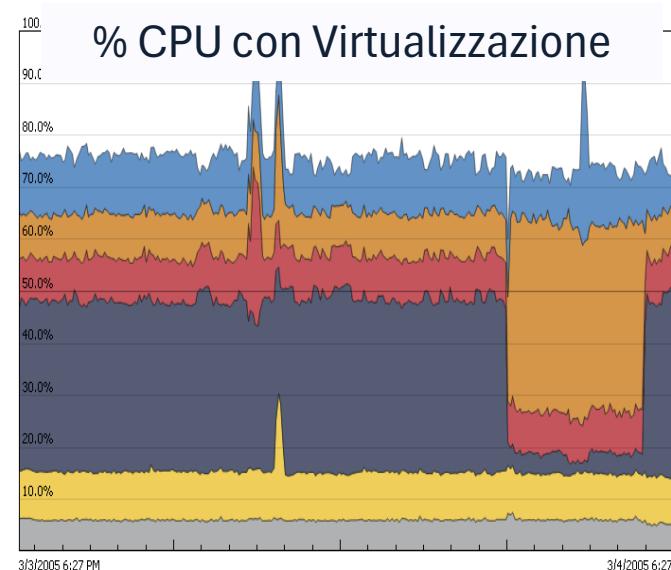
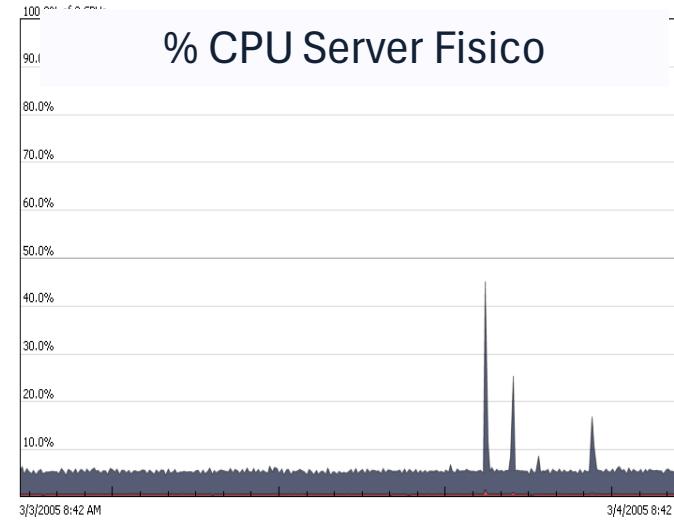
# Consolidamento di un CED

- Con l'introduzione della virtualizzazione si parla di "**consolidamento**" di un CED.
- Sfruttando al meglio le risorse di computazione parallela offerte dai nuovi processori, è possibile ridurre significativamente il numero di server necessari per erogare i servizi informatici richiesti da un ente



# Benefici introdotti dalla Virtualizzazione: Ottimizzazione

- Ottimizzazione delle risorse HW - La virtualizzazione permette di eseguire distinti sistemi operativi contemporaneamente in una singola macchina
- Ottimizzazione della occupazione dello spazio disco - ad una macchina virtuale viene assegnato lo spazio disco strettamente necessario.
  - Il resto del volume resta disponibile per nuove macchine virtuali



# Benefici introdotti dalla Virtualizzazione: Semplificazione

- Backup e disaster recovery
  - L'intero server (SO, applicazioni, dati, dispositivi e stato) non è altro che un file, più facile creare copie di backup per il disaster recovery
- Indipendenza dall'hardware
- Rilasci semplificati
  - Possibilità di clonare “LIVE” le macchine virtuali per test e aggiornamenti, senza compromettere l'integrità di quella originale.

# **Benefici introdotti dalla Virtualizzazione:**

## **Costi**

- Riduzione dei costi di IT
  - Riduzione del numero dei server necessari, con conseguente risparmio sui costi energetici, di condizionamento e dimensione degli spazi fisici richiesti dal CED.
- Riduzione dei costi di personale IT
  - Gestione centralizzata: Incremento di produttività del reparto IT, dovuto alla gestione centralizzata delle macchine attraverso un'unica interfaccia di amministrazione

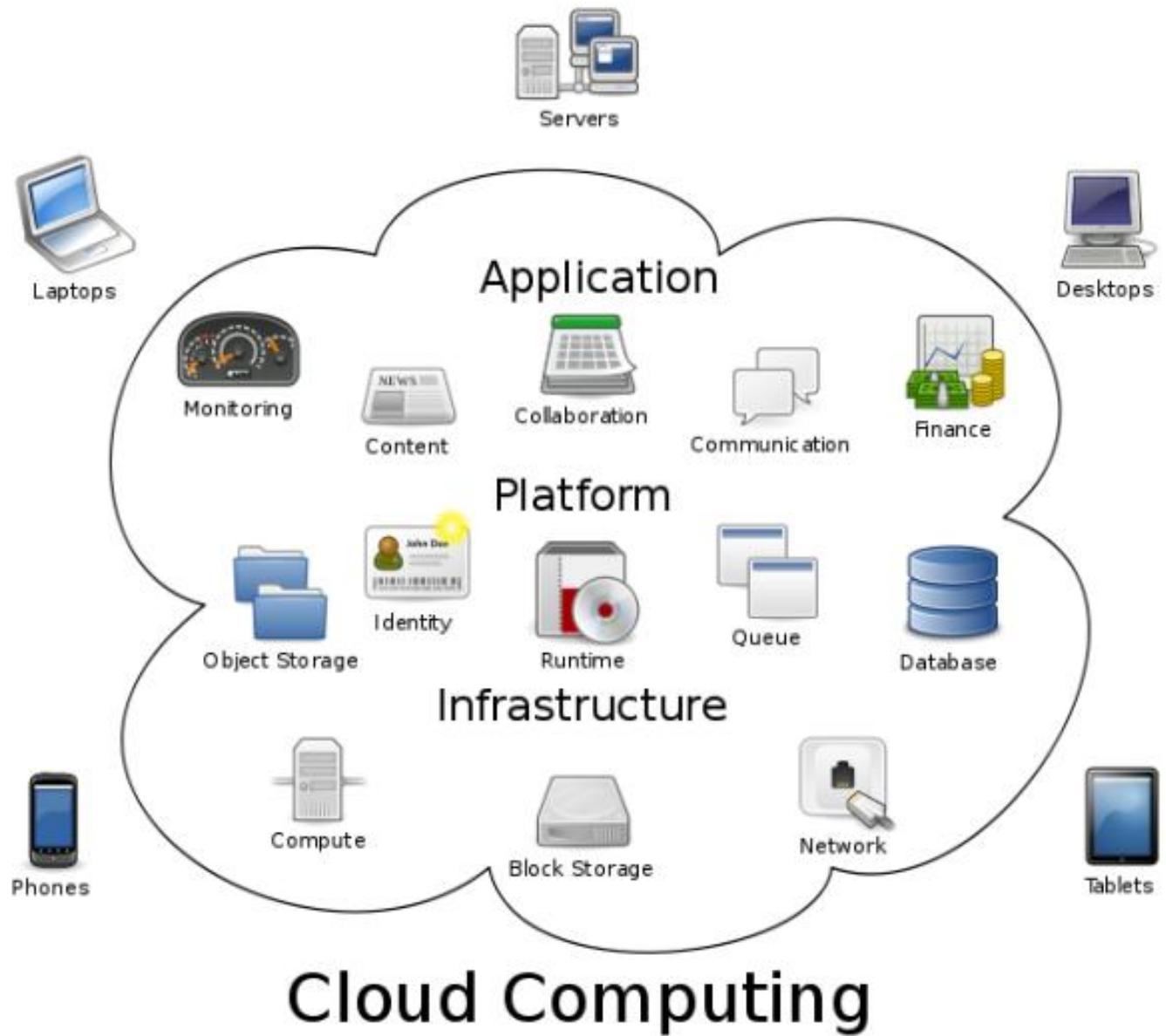
# Esempio in un CED della PA di Napoli

- Circa 800 macchine virtuali in esecuzione in parallelo, con bilanciamento del carico.
- Circa 30 server fisici.
- 2 Storage Area Network, con circa 500 TB di spazio.
- Gestione di 800 macchine da 2 console di amministrazione.

# Cloud Computing

# Cloud Computing

- IT resources provided as a service
  - Compute, storage, databases, queues
- Clouds leverage economies of scale of commodity hardware
  - Cheap storage, high bandwidth networks & multicore processors
  - Geographically distributed data centers
- Offerings from Microsoft, Amazon, Google, ...



# Cloud Scalability & Elasticity

# Scalability & Elasticity

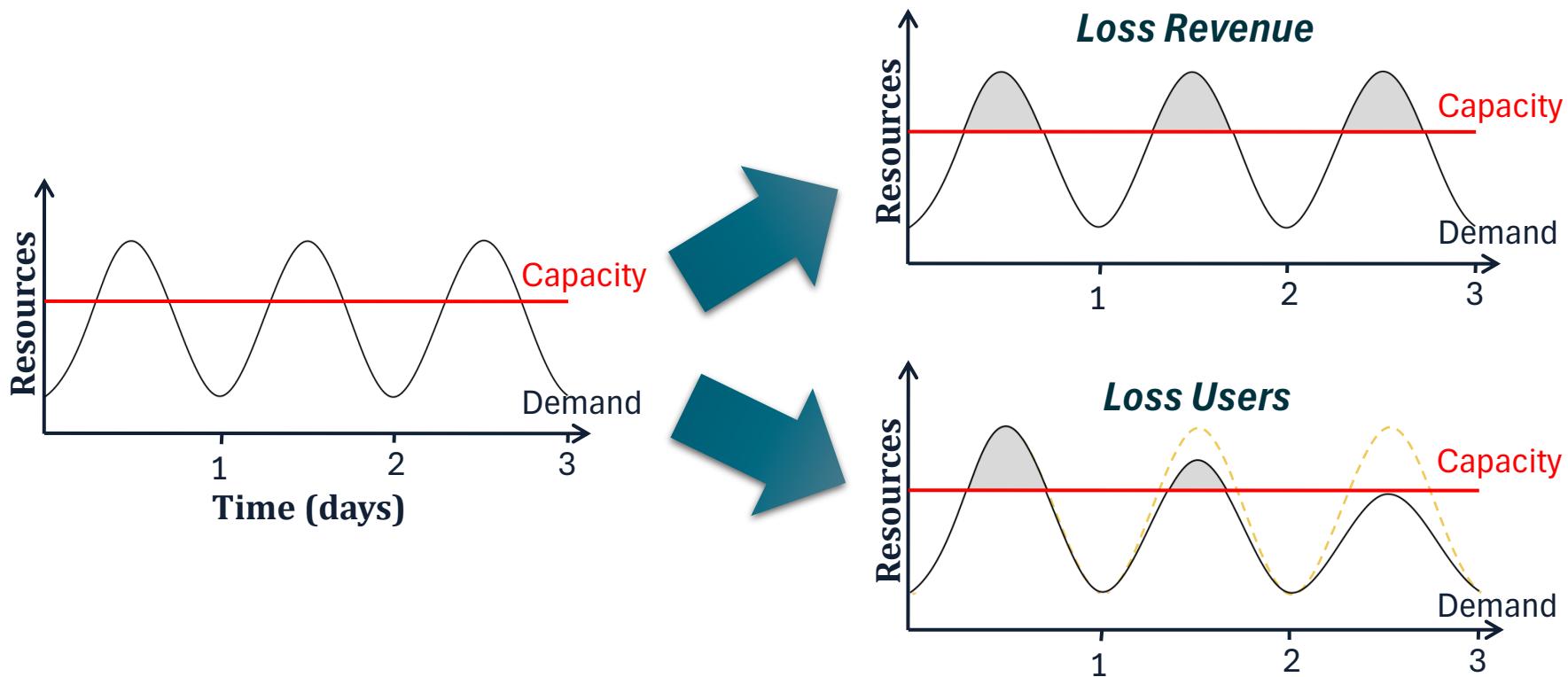
- What is scalability ?
  - A desirable property of a system, a network, or a process, which indicates its ability to either handle growing amounts of work in a graceful manner or to be readily enlarged.
- What is elasticity ?
  - The ability to grow or shrink infrastructure resources dynamically as needed to adapt to workload changes in an autonomic manner, maximizing the use of resources.
- But how to achieve these properties ?
  - Dynamic provisioning
  - Multi-tenant design

# Dynamic Provisioning

- Dynamic Provisioning is a simplified way to explain a complex networked server computing environment where server computing instances are provisioned or deployed from a administrative console or client application by the server administrator, network administrator, or any other enabled user.

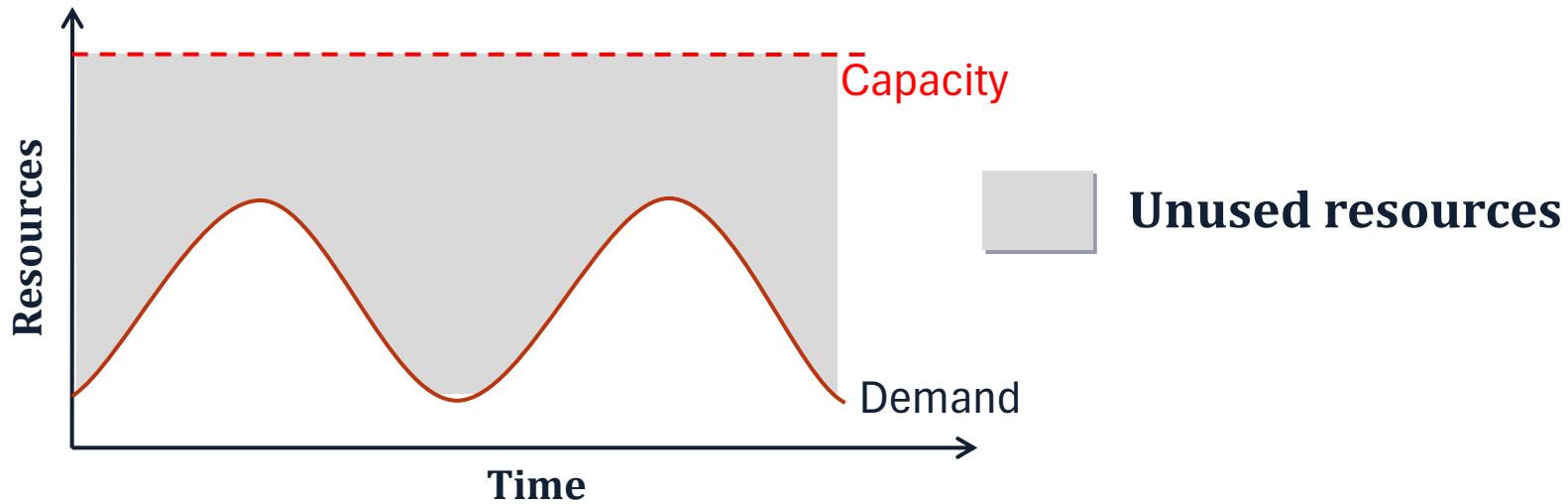
# Dynamic Provisioning

- In traditional computing model, two common problems :
  - Underestimate system utilization which result in under provision



# Dynamic Provisioning

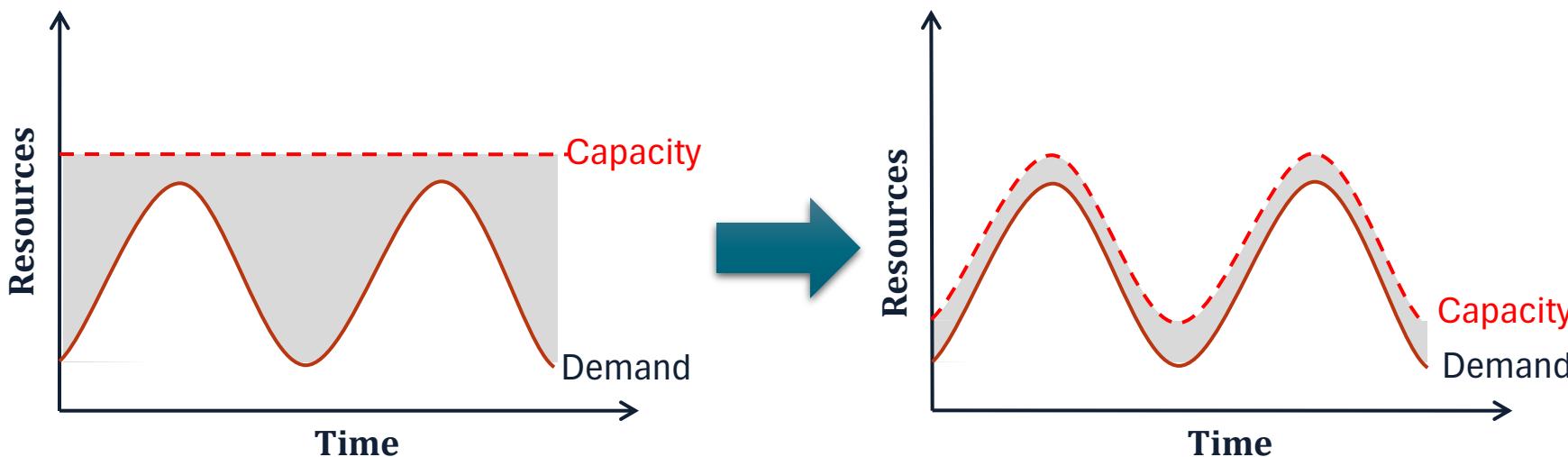
- Overestimate system utilization which result in low utilization



- How to solve this problem ??
  - Dynamically provision resources

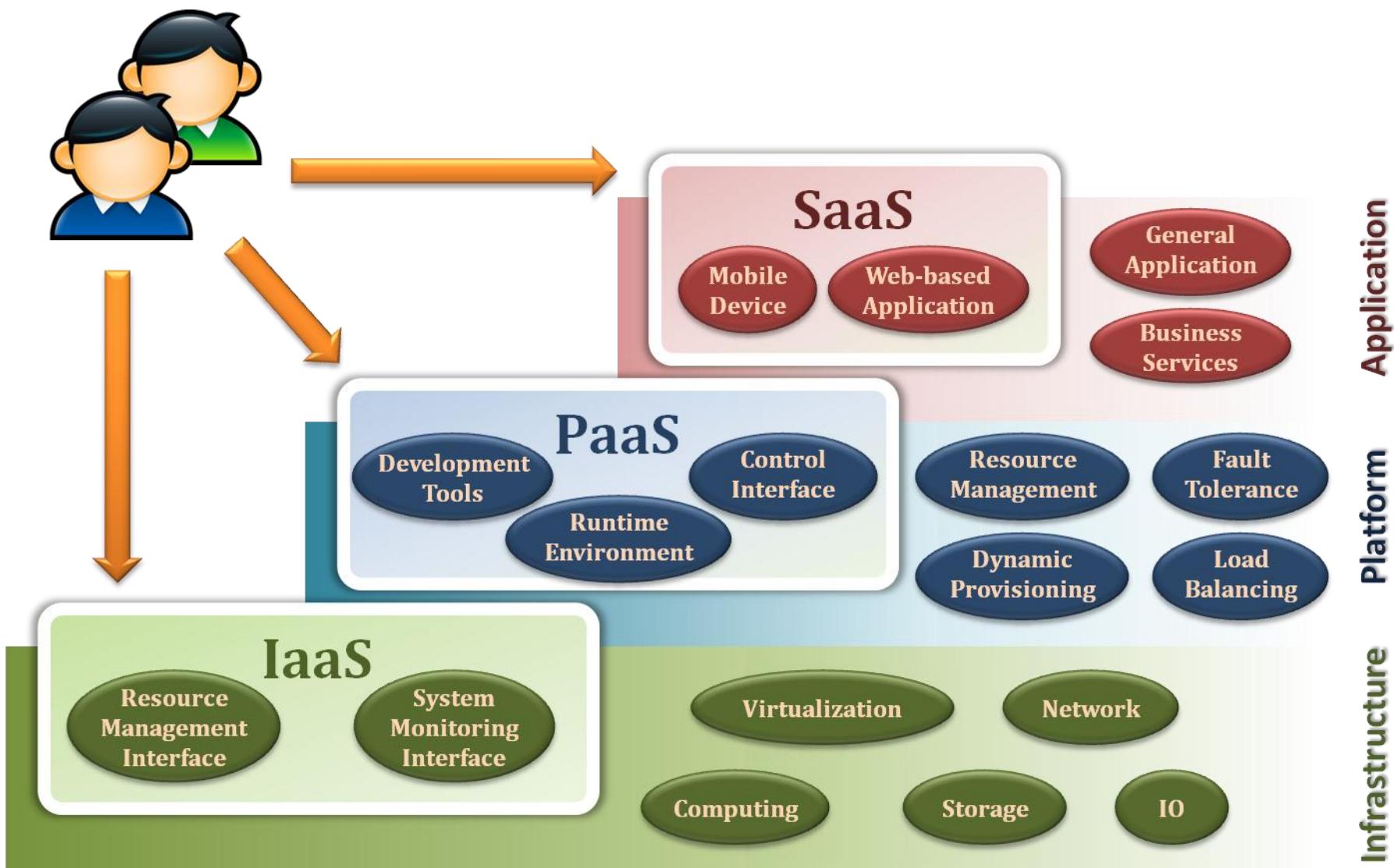
# Dynamic Provisioning

- Cloud resources should be provisioned dynamically
  - Meet seasonal demand variations
  - Meet demand variations between different industries
  - Meet burst demand for some extraordinary events



# **Service Models**

# Service Model Overview

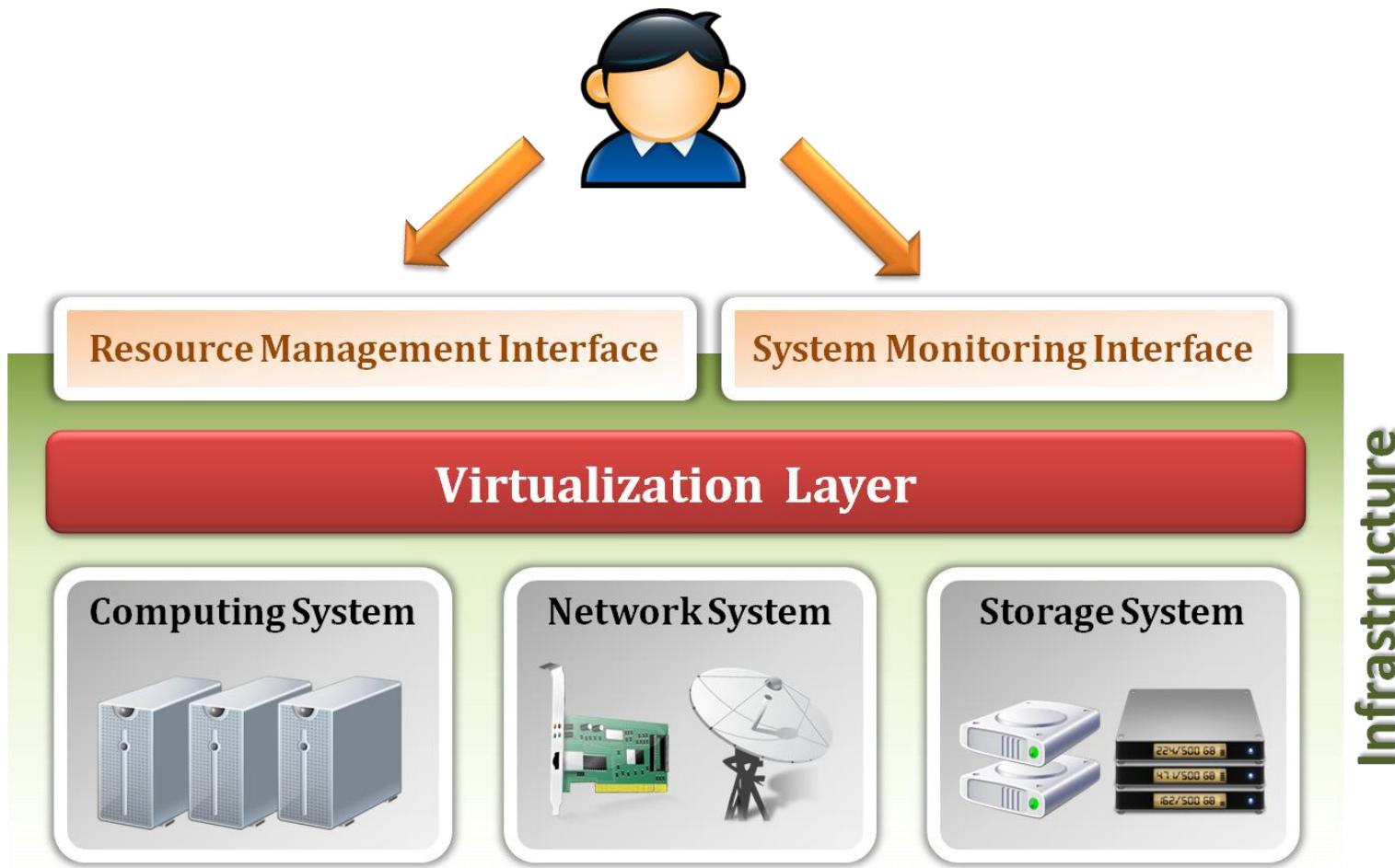


# Infrastructure as a Service - IaaS

- 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

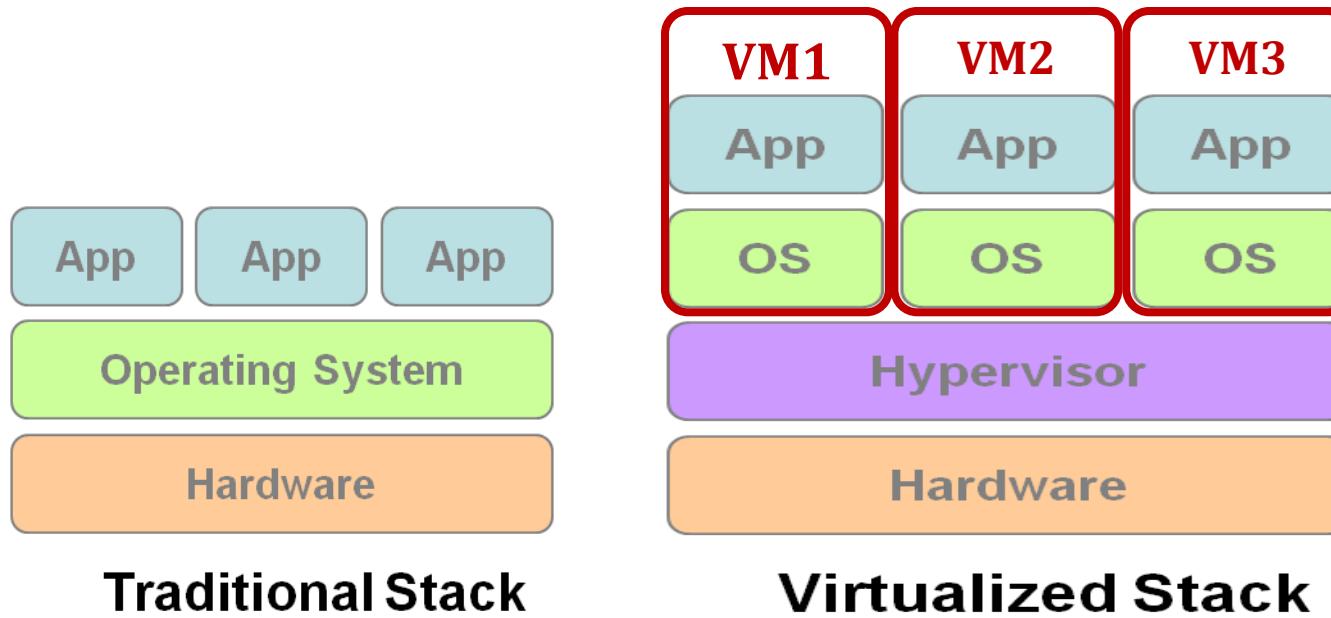
# Infrastructure as a Service

- 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.

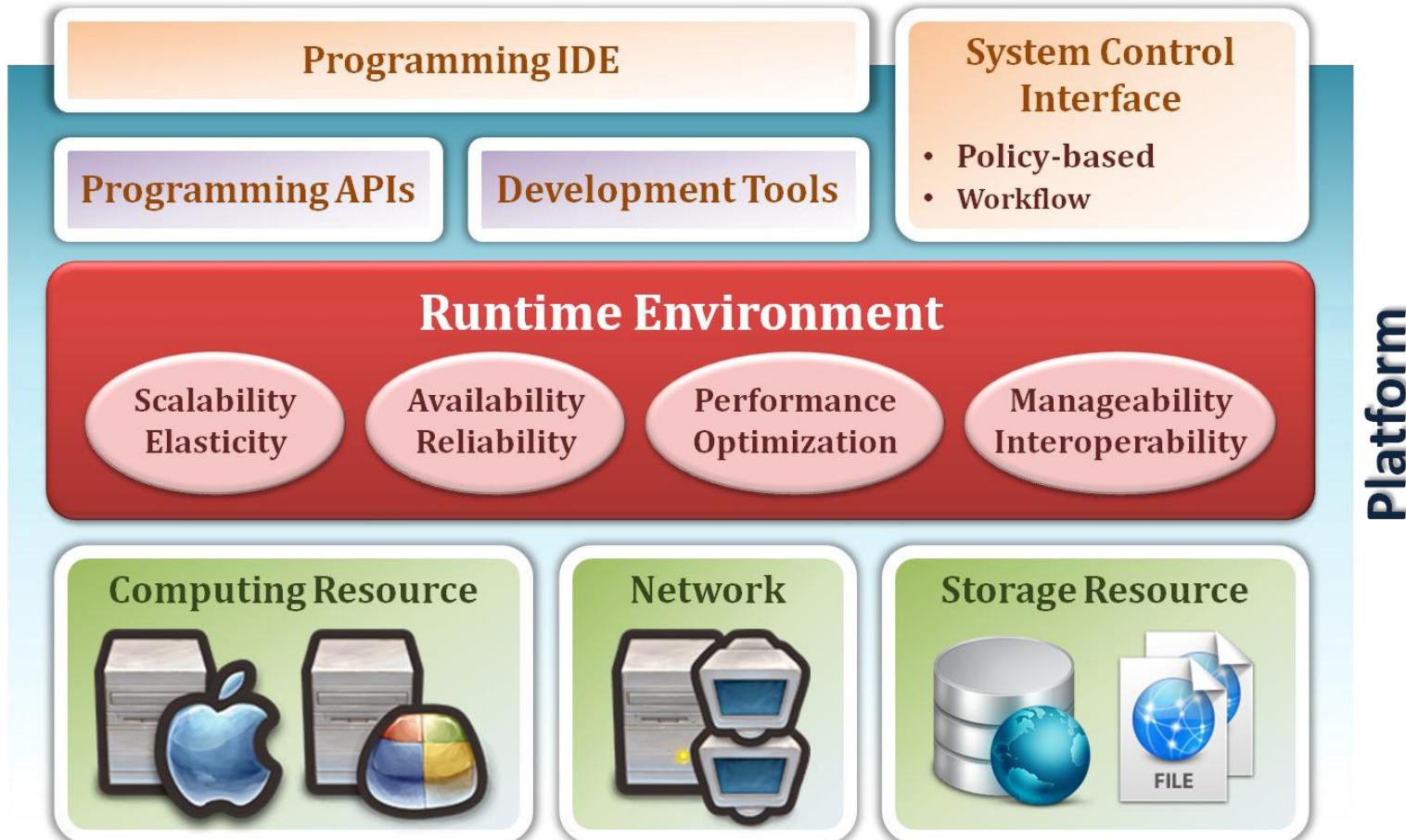


# Platform as a Service - PaaS

- 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



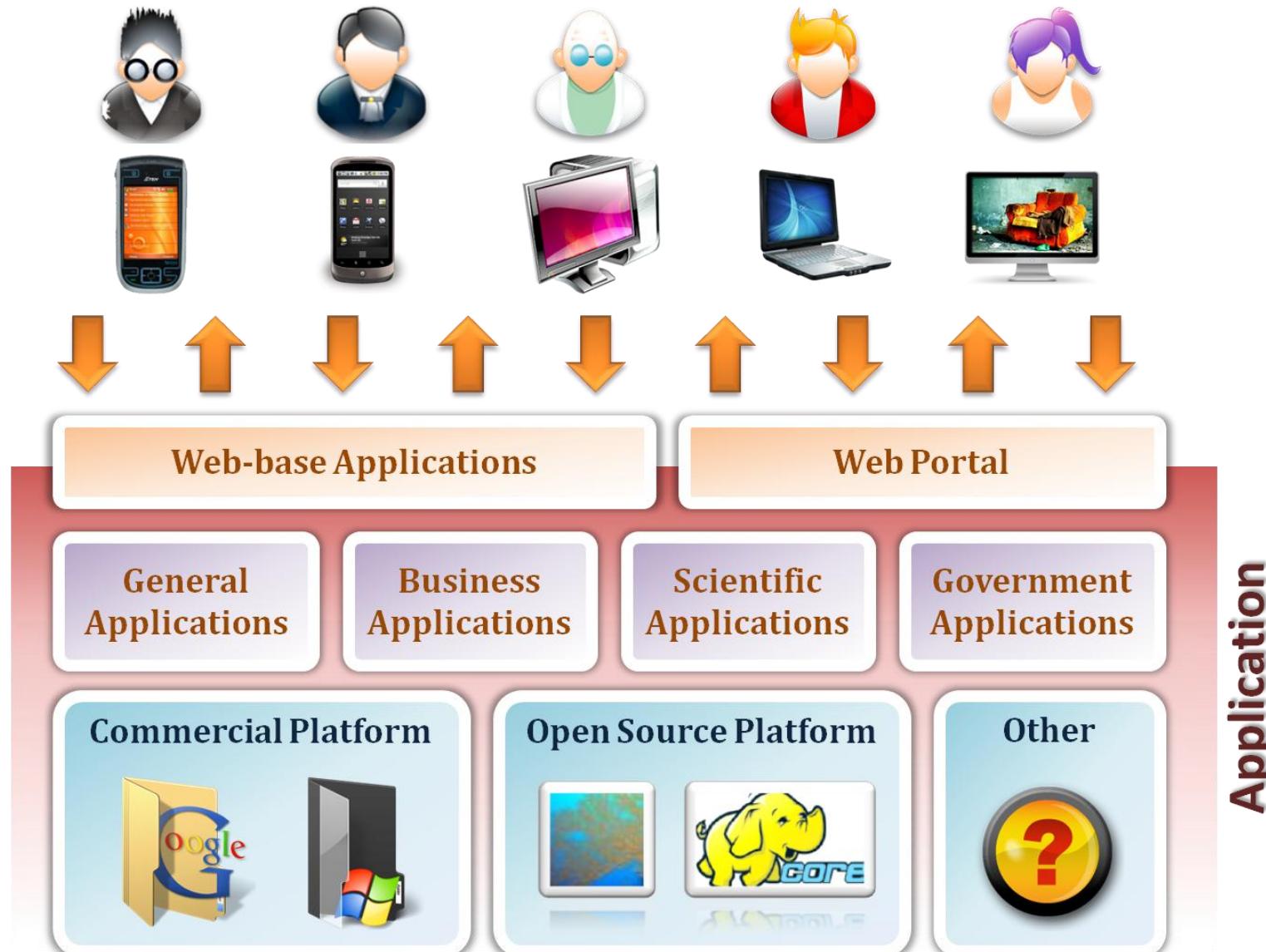
# Platform as a Service

- Provide service – **Programming IDE**
  - Users make use of programming IDE to develop their service among PaaS.
    - This IDE should integrate the full functionalities which supported from the underling runtime environment.
    - This IDE should also provide some development tools, such as profiler, debugger and testing environment.
  - The programming APIs supported from runtime environment may be various between different cloud providers, but there are still some common operating functions.
    - Computation, storage and communication resource operation

# 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

- 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.

# **Deployment models**

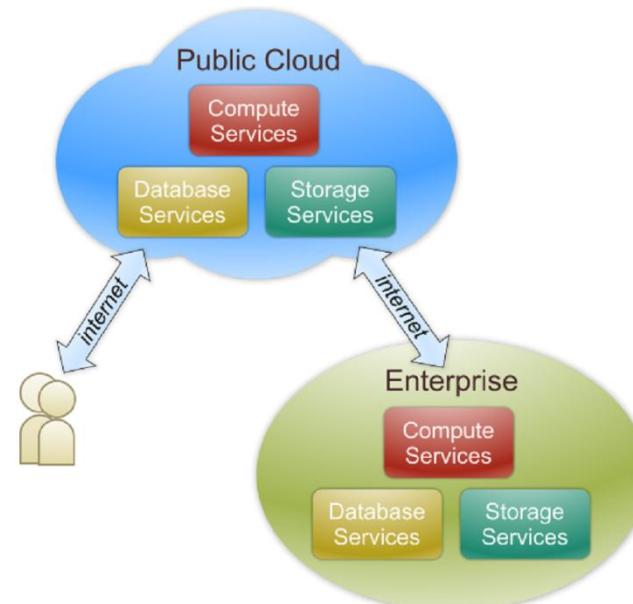
How to deploy a cloud system ?

# 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.

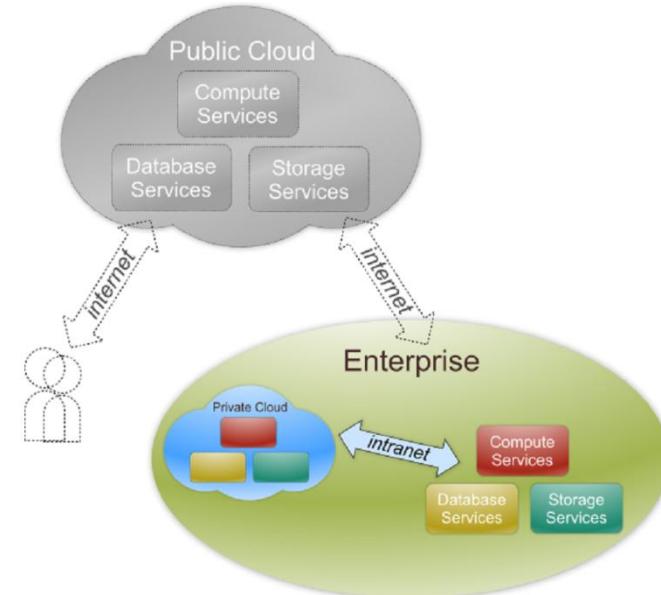
# Public Cloud

- 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

- 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



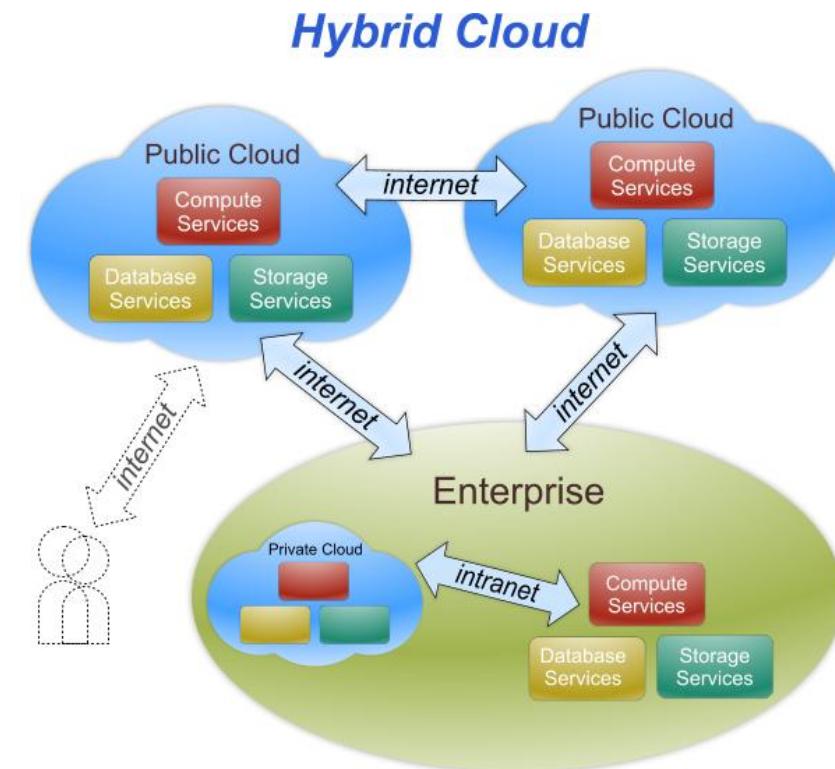
# Public vs. Private

- Comparison :

	Public Cloud	Private Cloud
<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>

# Hybrid Cloud

- 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

