



Design Patterns in Cloud Computing

Department of Computer Engineering at university of Guilan

25 Dec 2016





Overview

- Cloud Definition
- Characteristic
- Deployment models
- Type of Clouds
- IaaS
- PaaS
- SaaS
- TCO / ROI
- Cloud Migration Plan
- Design Patterns in Cloud Computing

Cloud Computing Definitions

- Forrester
 - ✓ A Standardized IT capability delivered via the **Internet**
 - ✓ Metered Services
 - ✓ Self Service (Private Cloud)
- NIST (National Institute of Standard and Technology)
 - ✓ A model for enabling
 - Ubiquitous and Convenient
 - On demand
- Larry Ellison
 - ✓ Just a computer connected to a network !
 - ✓ Changed his mind in 2015 ...

The Characteristics of Cloud Computing



On-demand
self-service



Ubiquitous
network
access



Location
transparent
resource
pooling

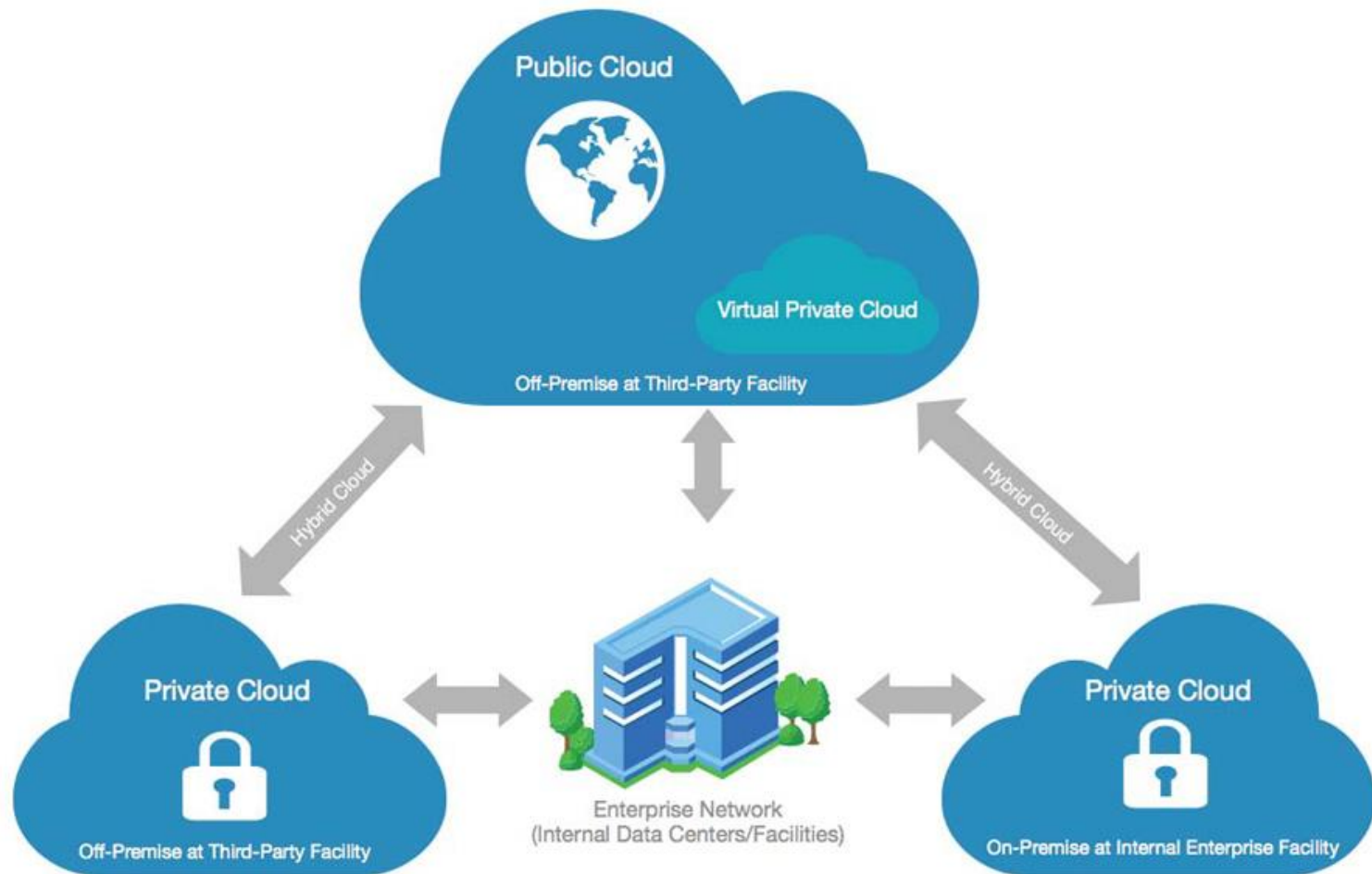


Rapid
elasticity



Measured
service with
pay per use

Cloud Deployment Models



Types of clouds

Applications

Middleware/OS

Servers

IaaS
host



Applications

Middleware/OS

Servers

PaaS
build



Applications

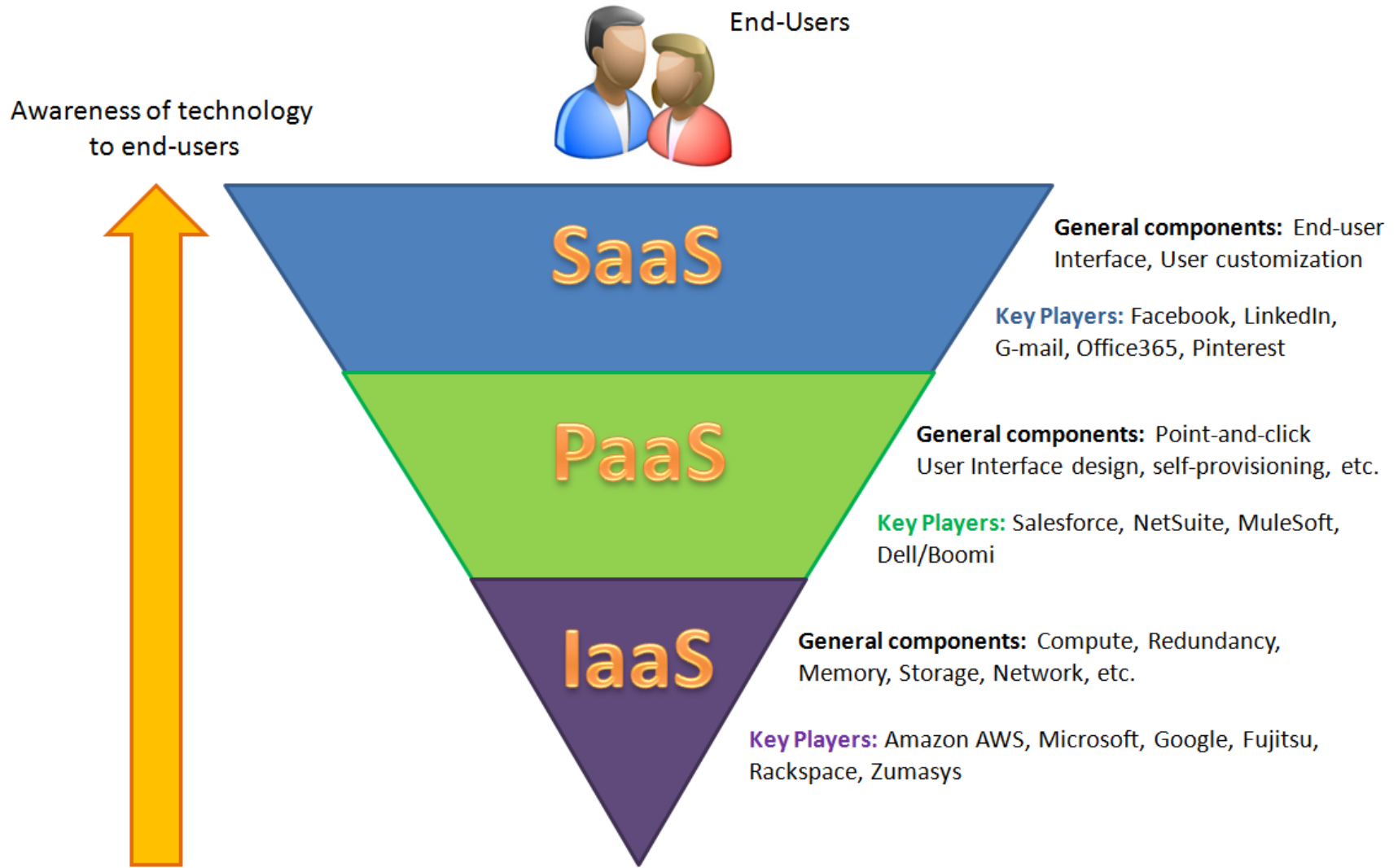
Middleware/OS

Servers

SaaS
Consume



Example of Type of Clouds



IaaS (Infrastructure as a Service)

Provides the following:

1. Servers-Compute , Machine
 2. Storage
 3. Network
 4. Operating System *
 5. And all the rest which are dealing with IaaS
- *The user instead of purchasing servers / software or even Data Centre Space and network equipment , rent those resources as a fully outsourced services on **demand-model**.*
 - *IaaS can be obtained as Public , Private and combination of both*
 - *It could be **Multi-tenancy**.*

IaaS (Infrastructure as a Service) Cont'd

❑ *When IaaS would be appropriate?*

- *While demand is very **volatile** (**Spike And Troughs**)
e.g: Digikala on Yalda Night*
- For new enterprise
 - ✓ **Shoestring budget**
 - ✓ Low investment
- Enterprise **grows rapidly** (Sudden and immediate success)
 - ✓ Normally scaling hardware would be problematic
e.g: Animoto (cloud base video creation)
- For Specific line of business
 - ✓ **Trial**

IaaS (Infrastructure as a Service) Cont'd

❑ *When IaaS would be inappropriate?*

- *Regulatory compliance in outsourcing*
- *Data Privacy*
 - *Personal Information*
 - *Medical Information*
- *Highest level of performance are needed*
 - *Network latency*

e.g: Sanjesh Servers are located in Iran and provided by Afranet

PaaS (Platform as a Service)

❑ *PaaS is a computing platform that abstract :*

- ✓ *Infrastructure*
- ✓ *OS and Middle-ware*

❑ *It helps Developer Productivity*

❑ *PaaS delivers more than infrastructure*

✓ *Solution stack*

- *Machine learning (speech API)*
- *Networking(Cloud CDN)*
- *Management Tools (Monitoring , Logging)*
- *Identity & Security*
- *Storage and Database (NOSQL , RDB)*
- *Etc*

PaaS- Solution Stack Examples

- ❑ *Is a set of software **subsystems** or components needed to create a complete platform*
- ❑ *No **additional software** is needed to support applications*
- ❑ *Applications are said to "**run on**" or "**run on top of**"*
- ❑ *Some common named stacks:*

- ✓ LAMP
- ✓ LAPP
- ✓ LNMP
- ✓ LEAP
- ✓ Ganeti
- ✓ XAMPP
- ✓ MEAN

PaaS (Platform as a Service) Cont'd

- *PaaS is **Magic Box***
- *Request anything **on demand***
- *Automatically build an initial environment*
- *Provide an ability to control fault tolerance **e.g : cage***
- *Supporting , **recovery** nightmare*
- *Supporting , **security property** to limit malicious behavior*
- *Most important :*
 - ✓ *Do not care about how it works*
 - ✓ *Pay as you go*

PaaS (Platform as a Service) Cont'd

- Provide complete *development , testing and deployment platform (environment , milestone)*
- Reduces the *complexity* of development and testing by placing the developer in a limited environment
- Supported by most *cloud providers* , including AWS , Google and Microsoft Azure or even integrated with your PaaS build Application

SaaS (Software as a Service)

- *Is often the path to saving the most money giving what enterprise are paying today for enterprise software.*
- *Removes the need for organizations to install and run hardware*
- *Eliminate the waves of hardware and software acquisition*
- *Eliminate server maintenance*
- *Supports both desktop and mobile computing*
- *Continuous automatic updates*

SaaS Examples

- ❑ There are more than **2000** software as a service
 - ✓ *Salesforce.com (Largest CRM system)*
 - ✓ *Google Apps for works (Automation)*
 - ✓ *Microsoft Office 365*

SaaS Examples, General Market

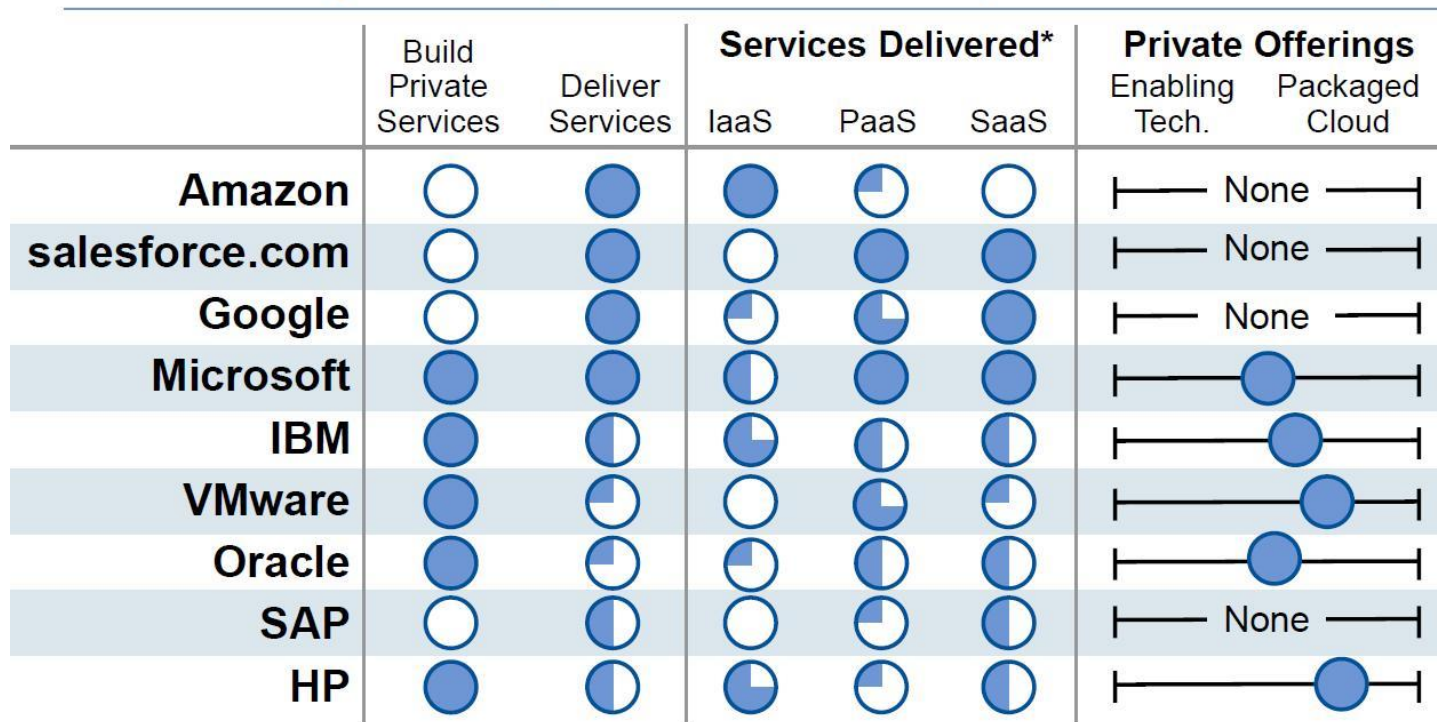
SaaS For Enterprises



SaaS For Consumers



Summary of Major Vendor Emphasis



Note: This is not an evaluation of capabilities, but rather of emphasis.



* The provider may offer public, community or virtual private services

TCO / ROI Models

- ❑ *Models the total cost of ownership for running an infrastructure environment*
- ❑ *TCO is used to :*
 - ✓ *Compare the costs of running an entire infrastructure environment*
 - ✓ *The cost of specific workload*
 - ✓ *Build the business case for migration*

...and that these benefits increase over time



According to IDC, this relationship between length of time using AWS and return is due to customers leveraging the more optimized environment to generate more applications along a learning curve.

Source: IDC Business Value of AWS Accelerates over time

Application Migration Common Methods and Approaches





Design Patterns in Cloud Computing

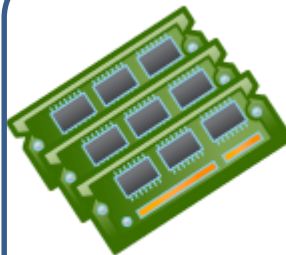
- Sharing, Scaling and Elasticity Patterns
- Reliability, Resiliency and Recovery Patterns
- Data Management and Storage Device Patterns
- Virtual Server and Hypervisor Connectivity and Management Patterns
- Monitoring, Provisioning and Administration Patterns
- Cloud Service and Storage Security Patterns
- Network Security, Identity & Access Management and Trust Assurance Patterns
- Common Compound Patterns

Resource Pooling Pattern

- **Goals:**
 - Organize IT resources
 - Support Dynamic Sharing



CPU pool



Memory pool



Storage pool



Network pool

Resource Pooling Pattern Cont'd

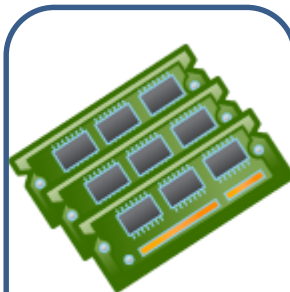
- **Problem:**
 - Need of *synchronization*
 - When assembling IT resources
 - Need of more *attention* in sharing and scalability purposes
 - Shared Resources
 - Dynamic scalability
 - Challenges in manually *maintaining the level of synchronicity* in collections of shared IT resources

Resource Pooling Pattern

- **Solution:**
 - *Grouping* IT resources into Resource Pools
 - Maintaining by an *Automatic System* to remain synchronized



CPU pool



Memory pool



Storage pool



Network pool

Resource Pooling Pattern

- **Application:**
 - Establish a *Reliable Synchronization System*
 - Support of Shared Resources
 - Support of Dynamic Scalability
 - Can be further supported by the application of *Resource Reservation Pattern*



Resource Pooling Pattern

- ***Application:***

Networked servers with OS
and other applications



Physical Server pool

Ubuntu Servers with
2 Gigabytes of RAM



Virtual Server pool

Empty or filled
Cloud Storage Devices.



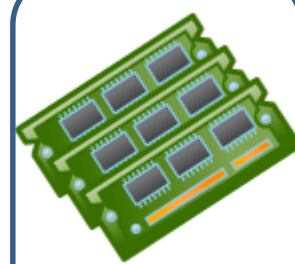
Storage pool



Network pool



CPU pool



Memory pool

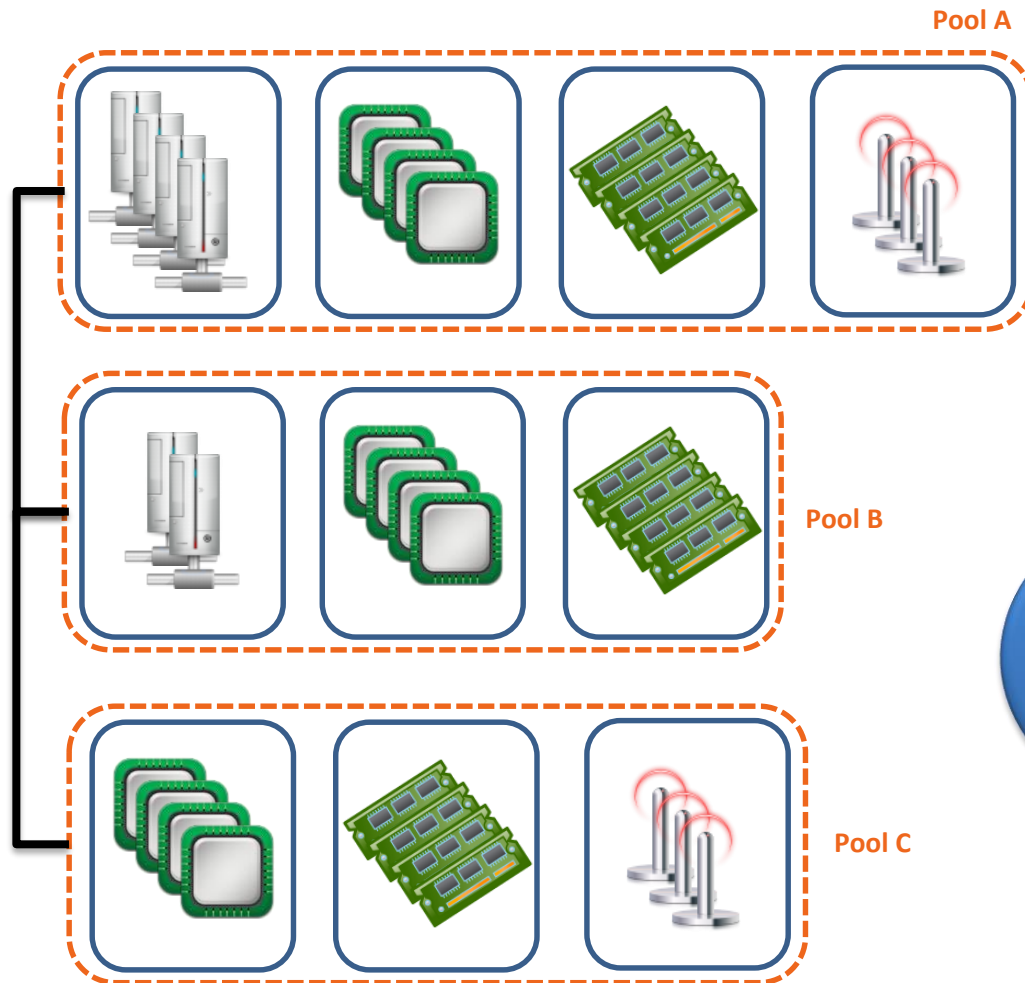
A pool of virtual firewall devices

Processor Pools to be
allocated to Virtual Servers

Memory Pools used in
physical servers

Resource Pooling Pattern

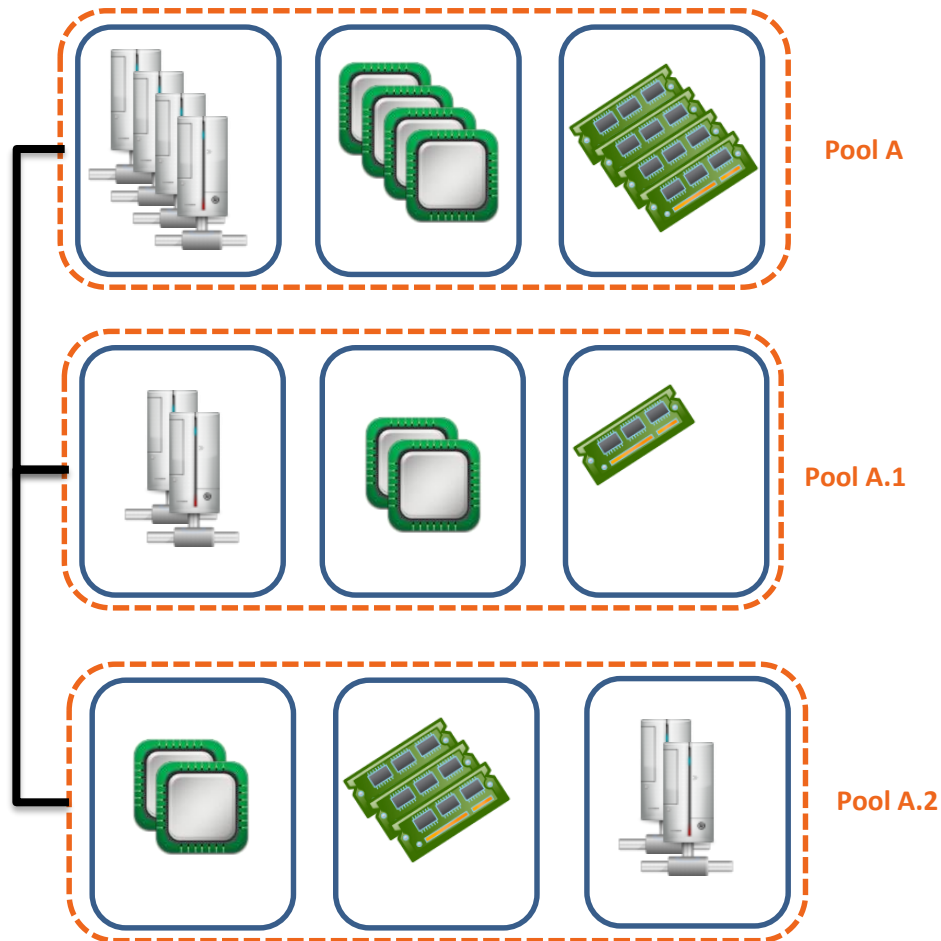
- *Application:*



Shared Pools B and C are sibling pools taken from the larger Pool A

Resource Pooling Pattern

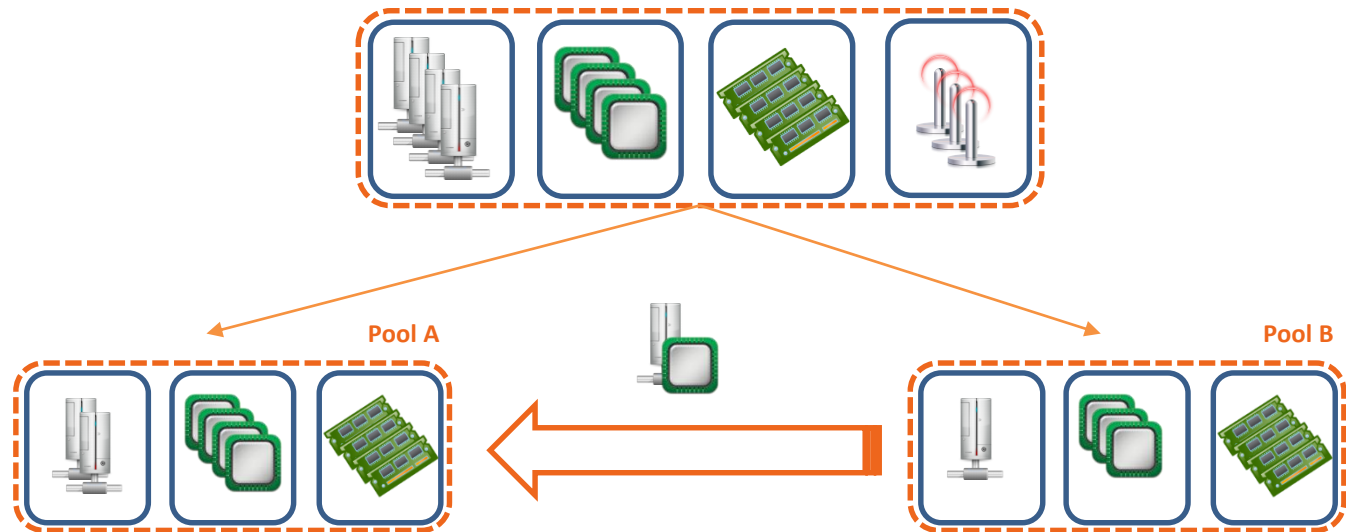
- *Application:*



Nested Pools A.1 and A.2 are comprised of the same IT resources as Pool A, but in different quantities

Resource Reservation Pattern

- **Goals:**
 - *Protect IT resources from conflicts of Concurrent Access*



Resource Reservation Pattern

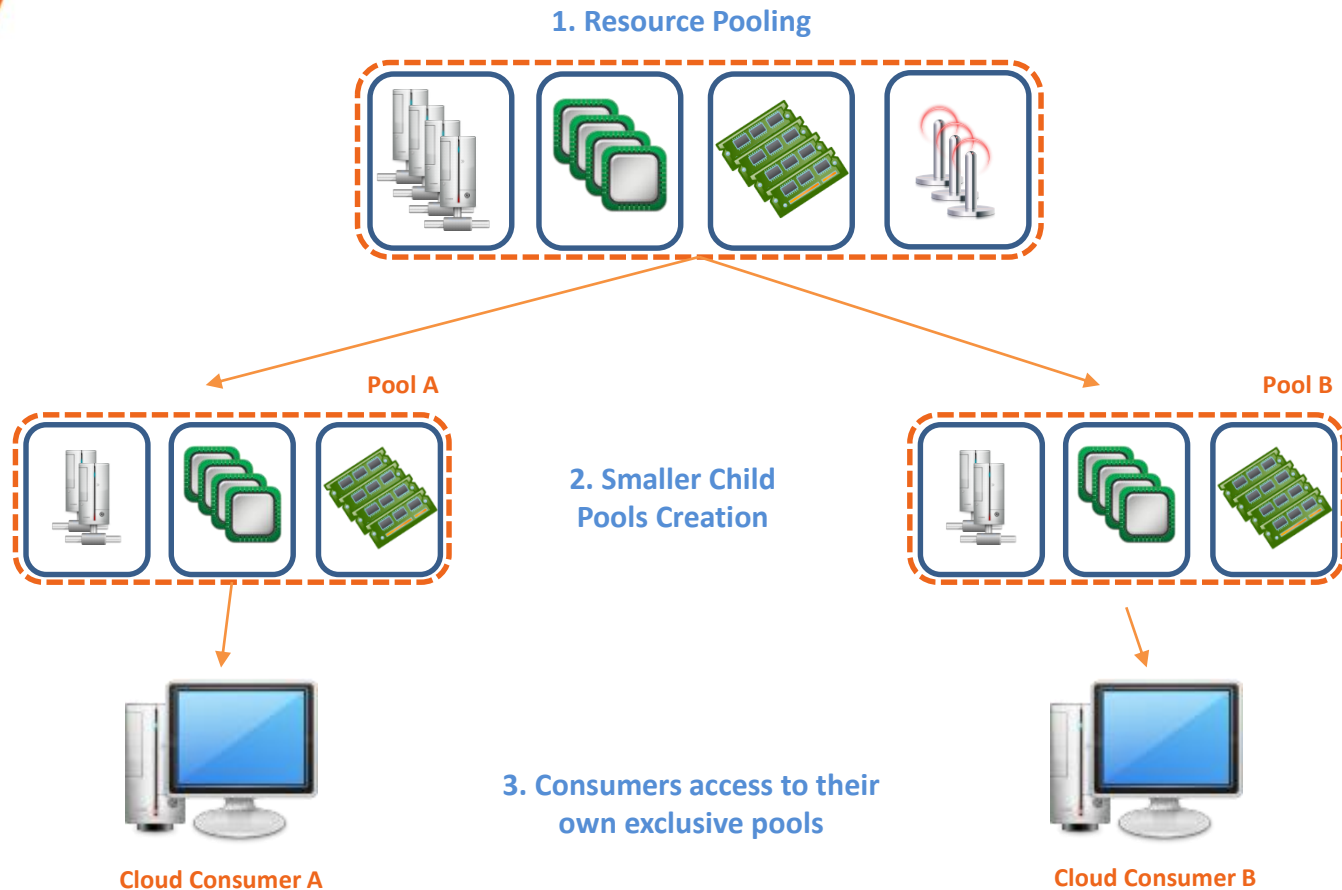
- **Problem:**
 - Multiple cloud service consumers may access to *the same IT resources*
 - *Resource Constraint* exception:
 - Several consumers share an IT resource with *no capacity* for processing requirements
 - Result: degraded performance or rejection
 - *Resource Borrowing* exception:
 - One pool *temporarily borrows IT resources* from other pools

Resource Reservation Pattern

- **Solution:**
 - IT resource *Reservation System*
 - Guarantees a *minimum amount* of an IT resource for each consumer
 - Exclusively available *only* to the designated cloud consumer
 - Solve *resource constraints* and *resource borrowing*
 - The reserved IT resources are never actually shared!

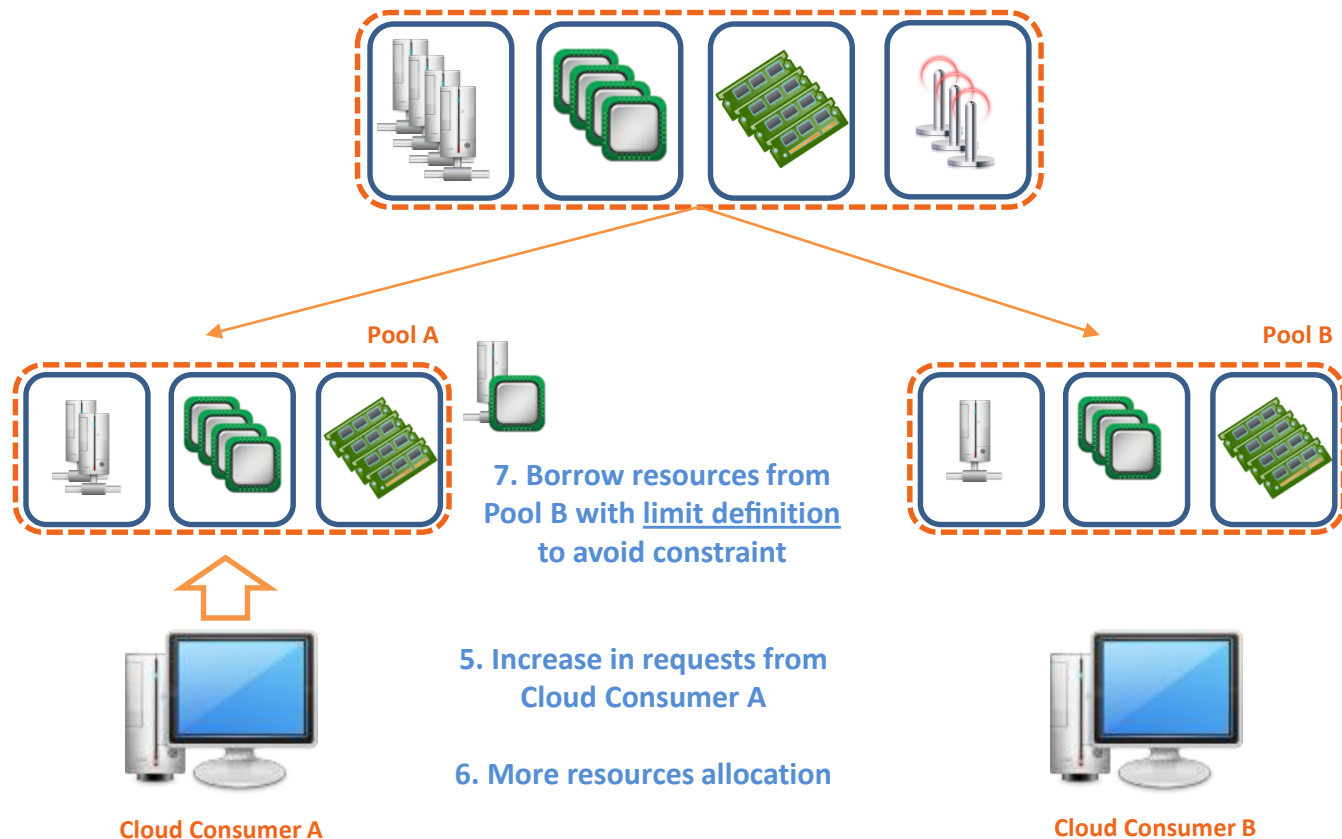
Resource Reservation Pattern

- **Application:**



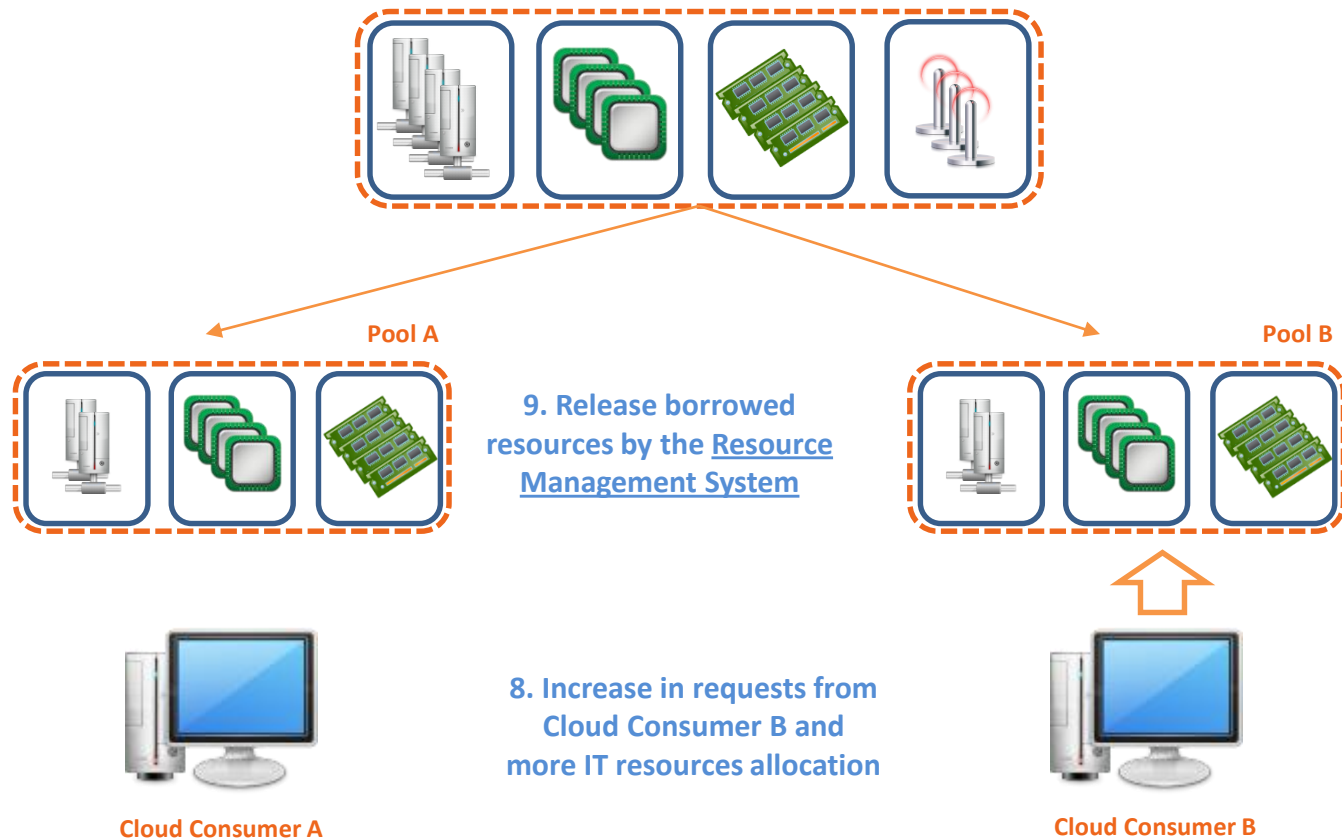
Resource Reservation Pattern

- *Application:*



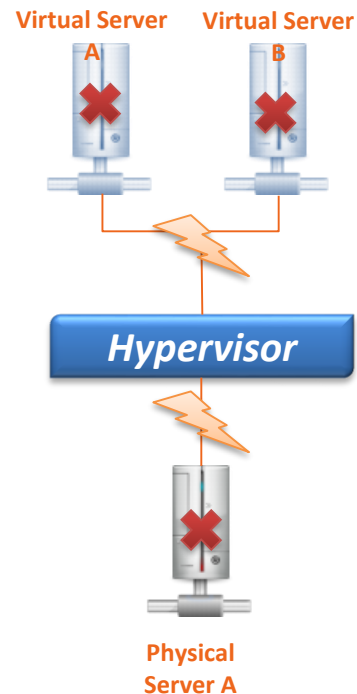
Resource Reservation Pattern

- *Application:*



Hypervisor Clustering Pattern

- **Goals:**
 - Virtual Servers should survive the *failure* of its hosting Hypervisor/physical server

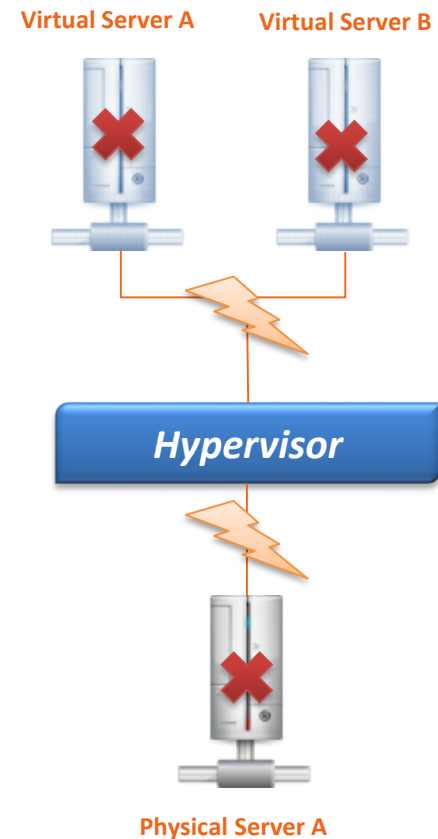
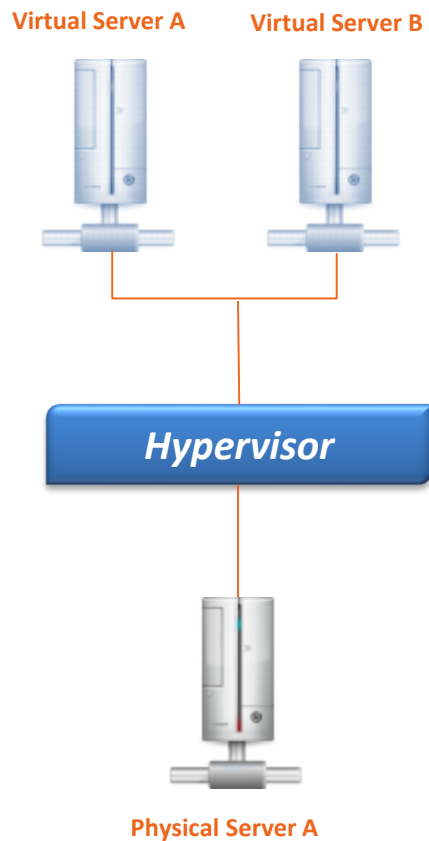


Hypervisor Clustering Pattern

- **Problem:**
 - *Hypervisor*: an application to create and run virtual servers
 - Failure of *hypervisor* or underlying *physical server*:
 - Cascades to all of its hosted virtual servers

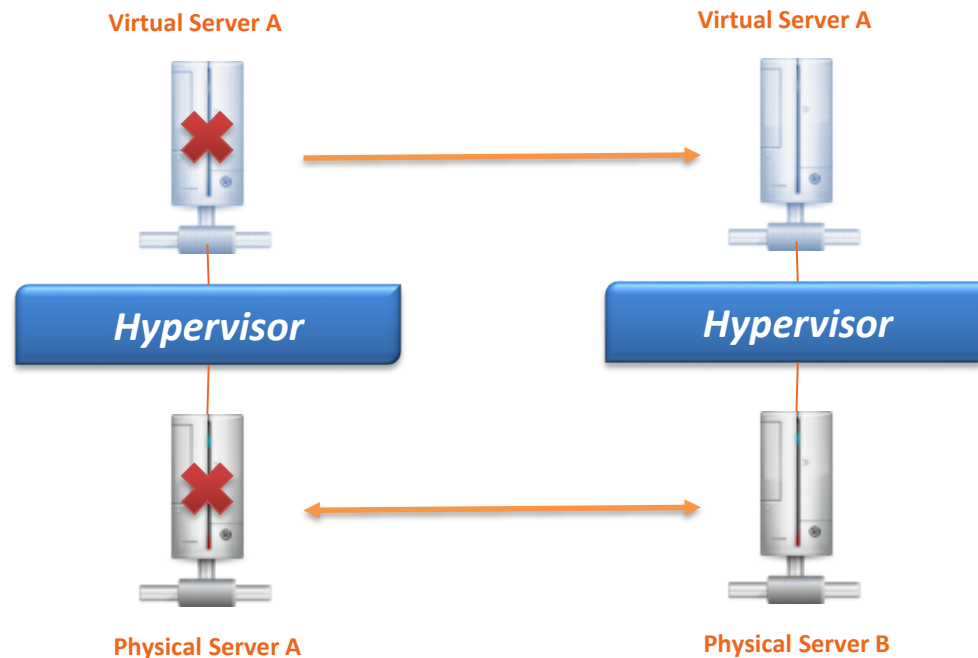
Hypervisor Clustering Pattern

- ***Problem:***



Hypervisor Clustering Pattern

- **Solution:**
 - Create High-availability *Hypervisor Clusters*
 - Failure of physical server or hypervisor
 - *Move* Virtual Servers to another Physical Server or hypervisor

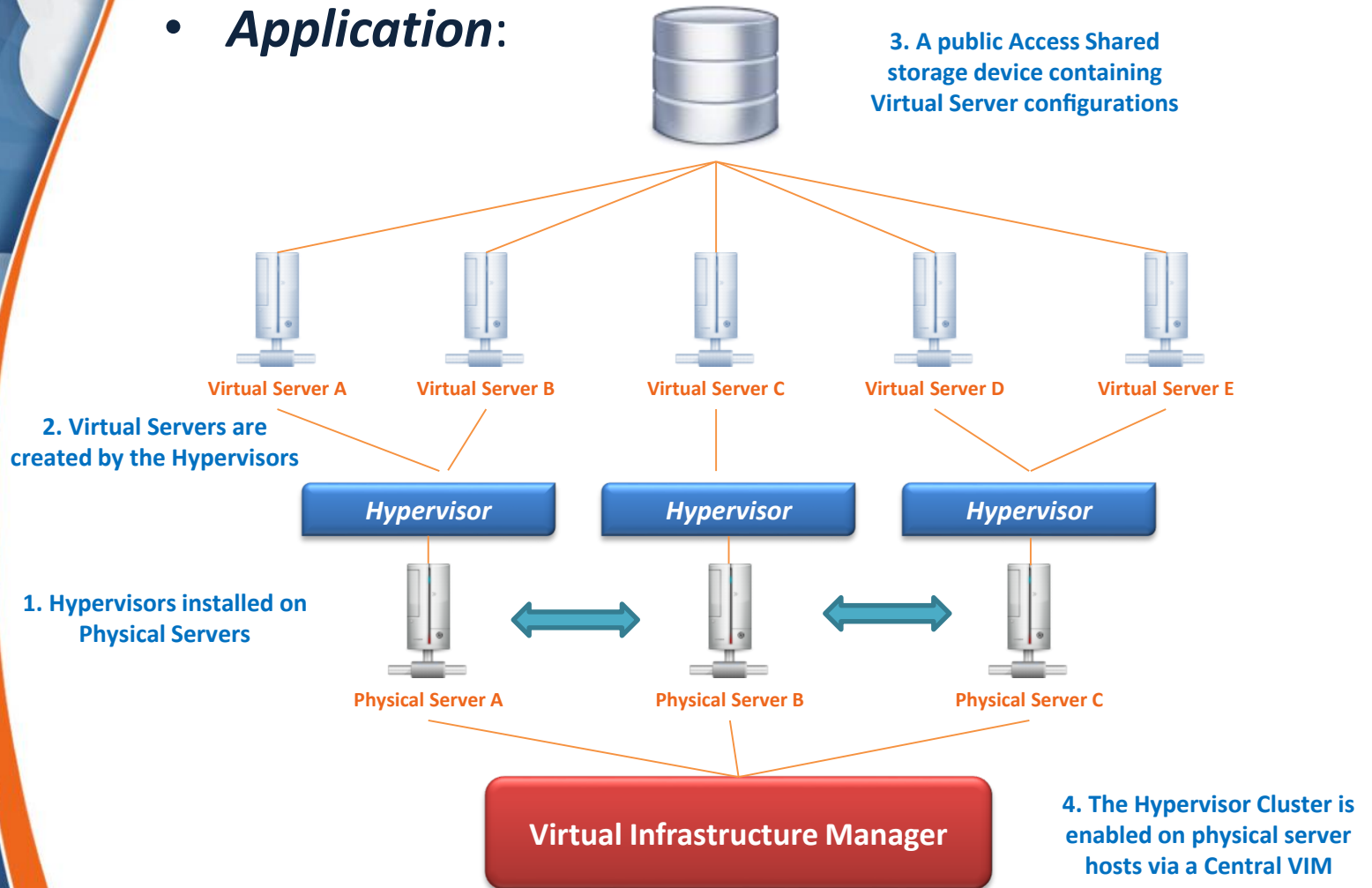


Hypervisor Clustering Pattern

- ***Application:***
 - *Hypervisor Cluster control via a central VIM*
 - Sends regular *heartbeat messages* to the hypervisors
 - *Check* if running and up
 - *Fail to acknowledge:*
 - Initiate the live VM *migration* program
 - *Move* affected virtual servers to a new host

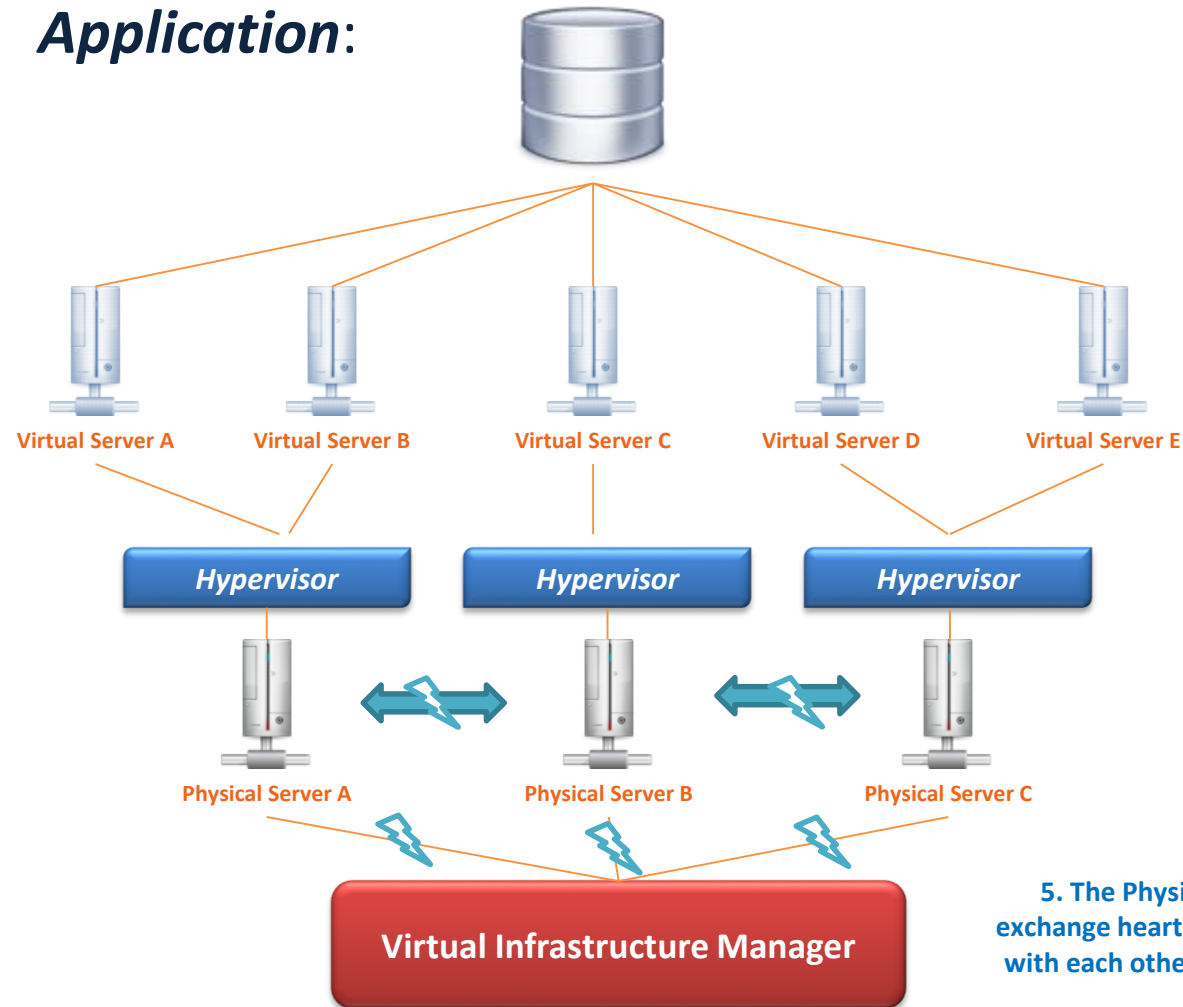
Hypervisor Clustering Pattern

- Application:**



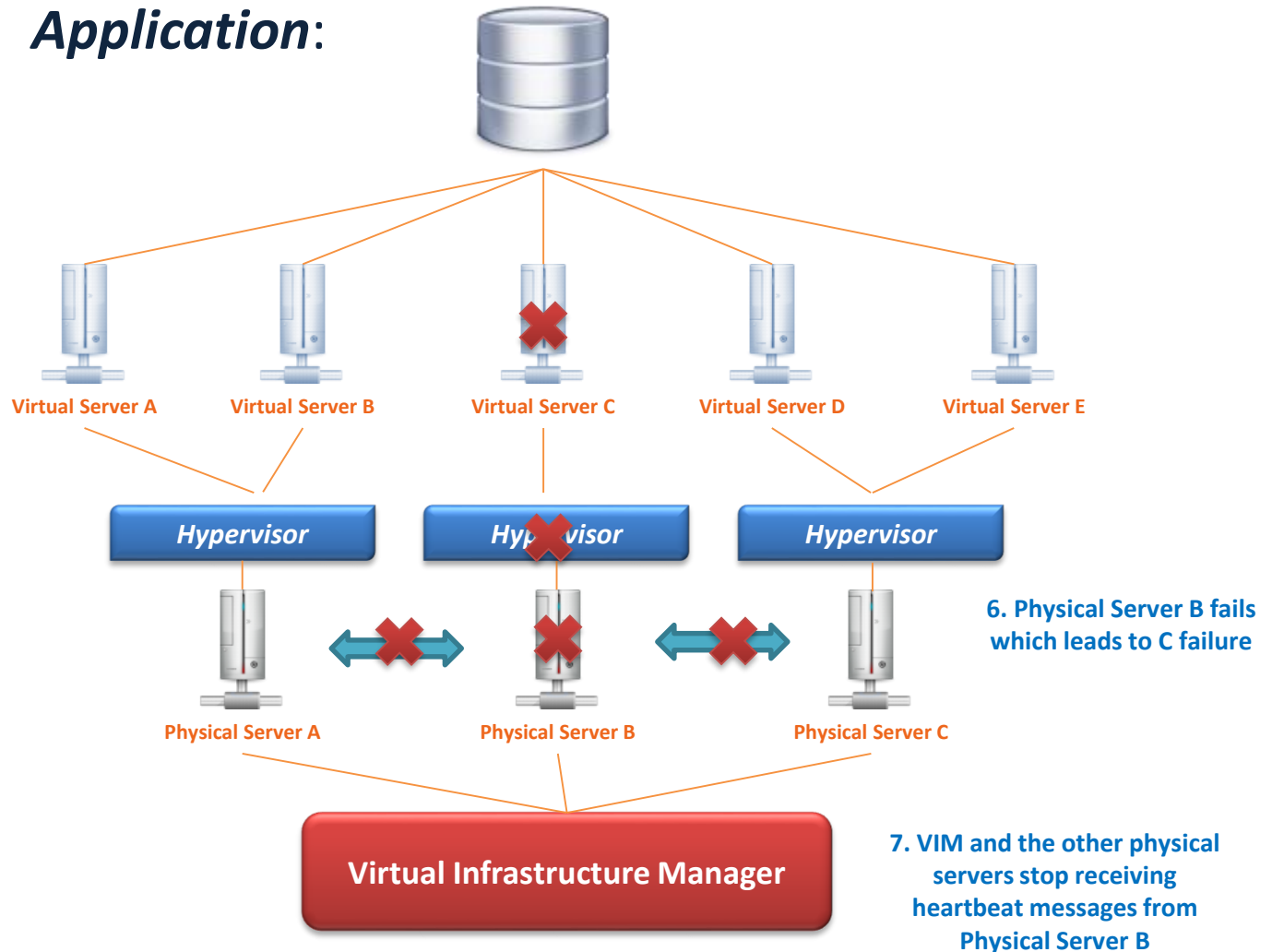
Hypervisor Clustering Pattern

- Application:***



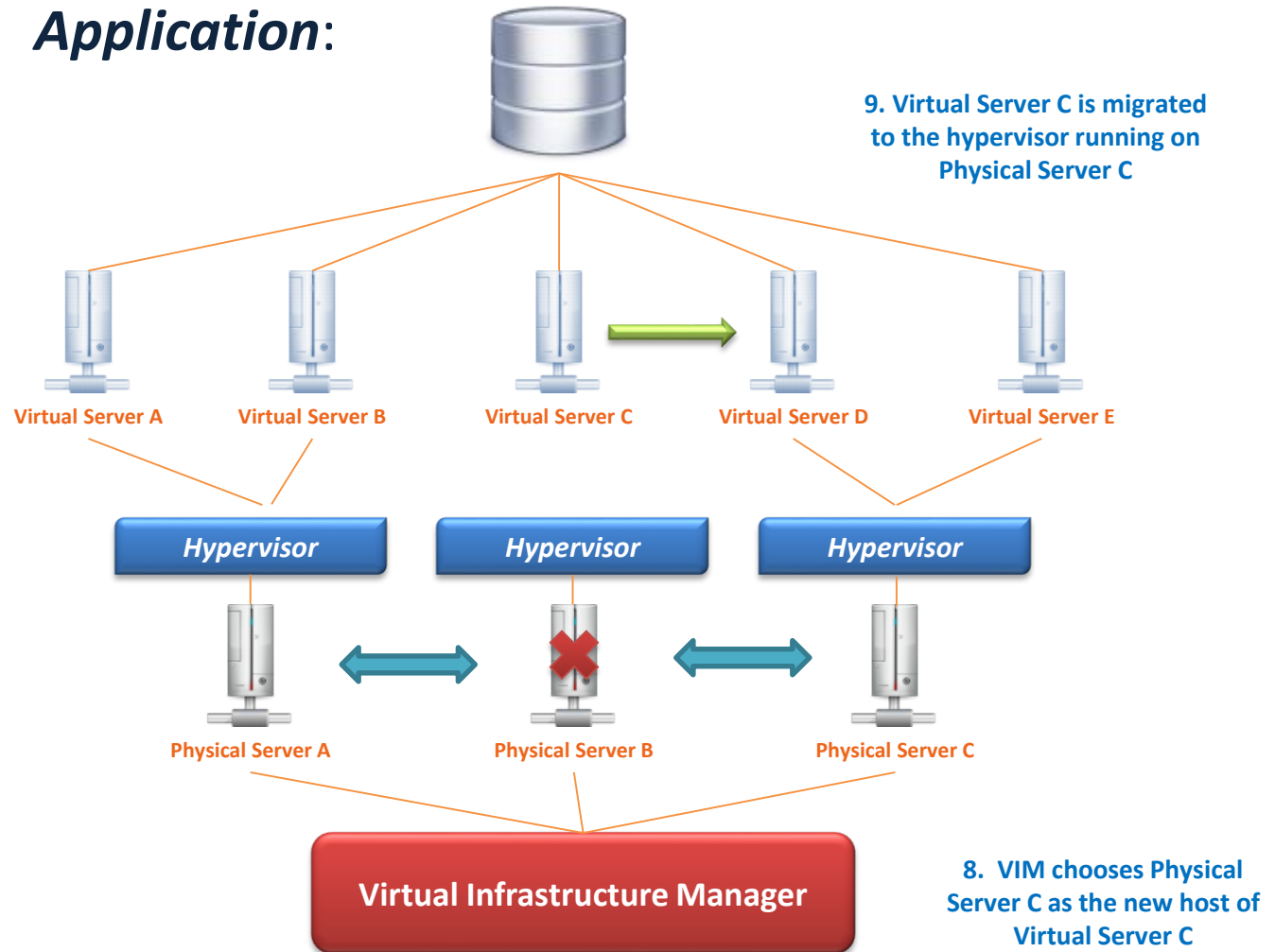
Hypervisor Clustering Pattern

- Application:**



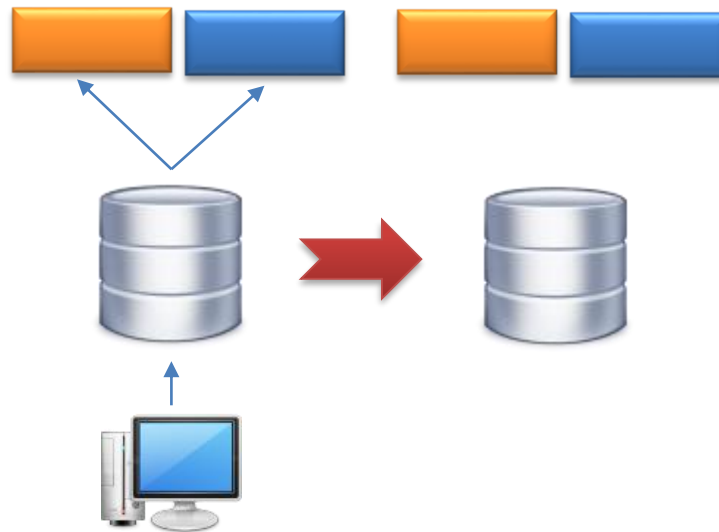
Hypervisor Clustering Pattern

- Application:**



Redundant Storage Pattern

- **Goals:**
 - *Reliability* and *availability* of cloud storage devices survival in failure conditions

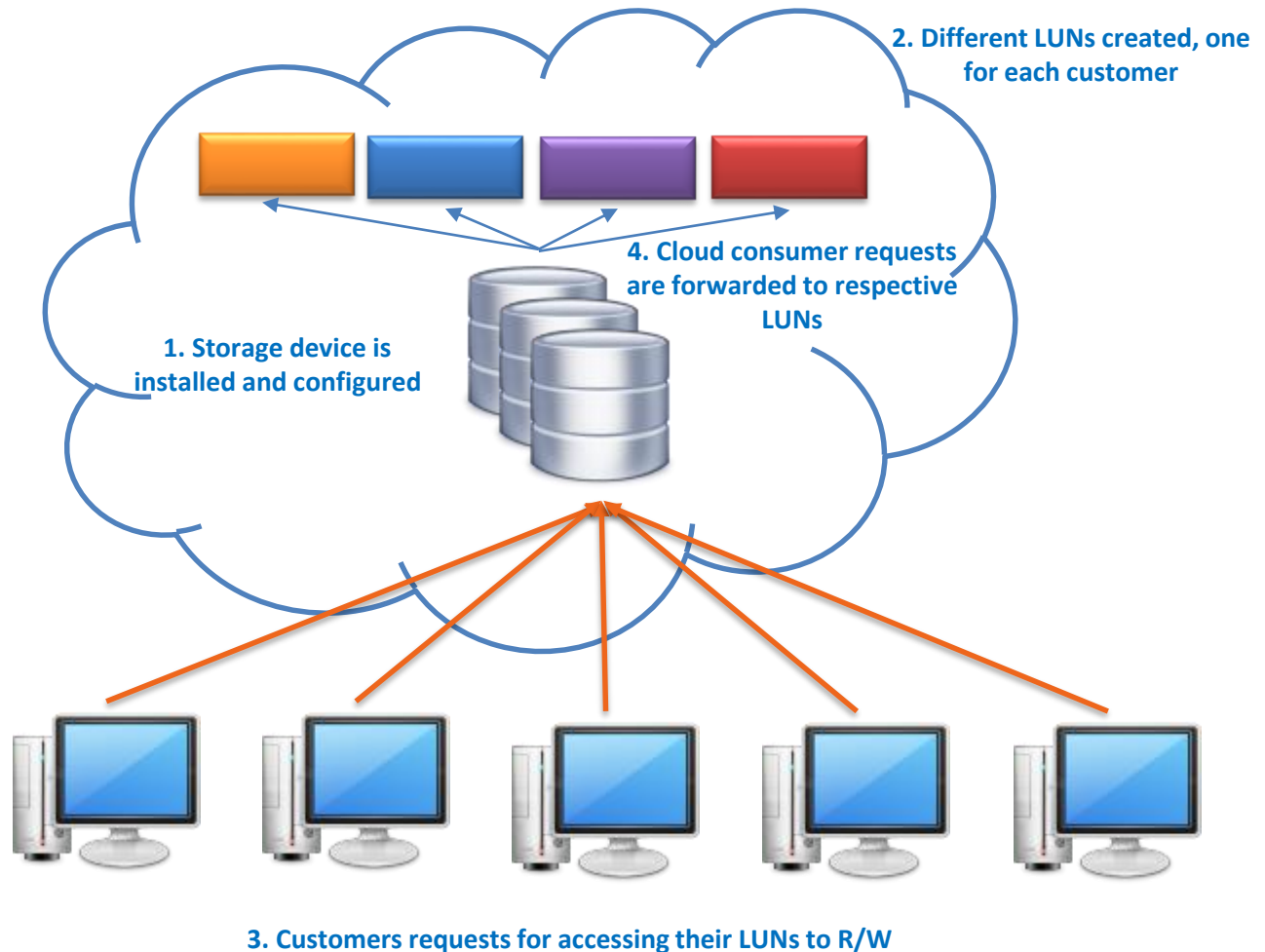


Redundant Storage Pattern

- **Problem:**
 - Failure of *cloud storage devices*
 - Network connectivity issues
 - Controller failures
 - General hardware failure
 - Security breaches
 - Effects of failure across
 - Cloud services
 - Cloud-based applications
 - Cloud infrastructure

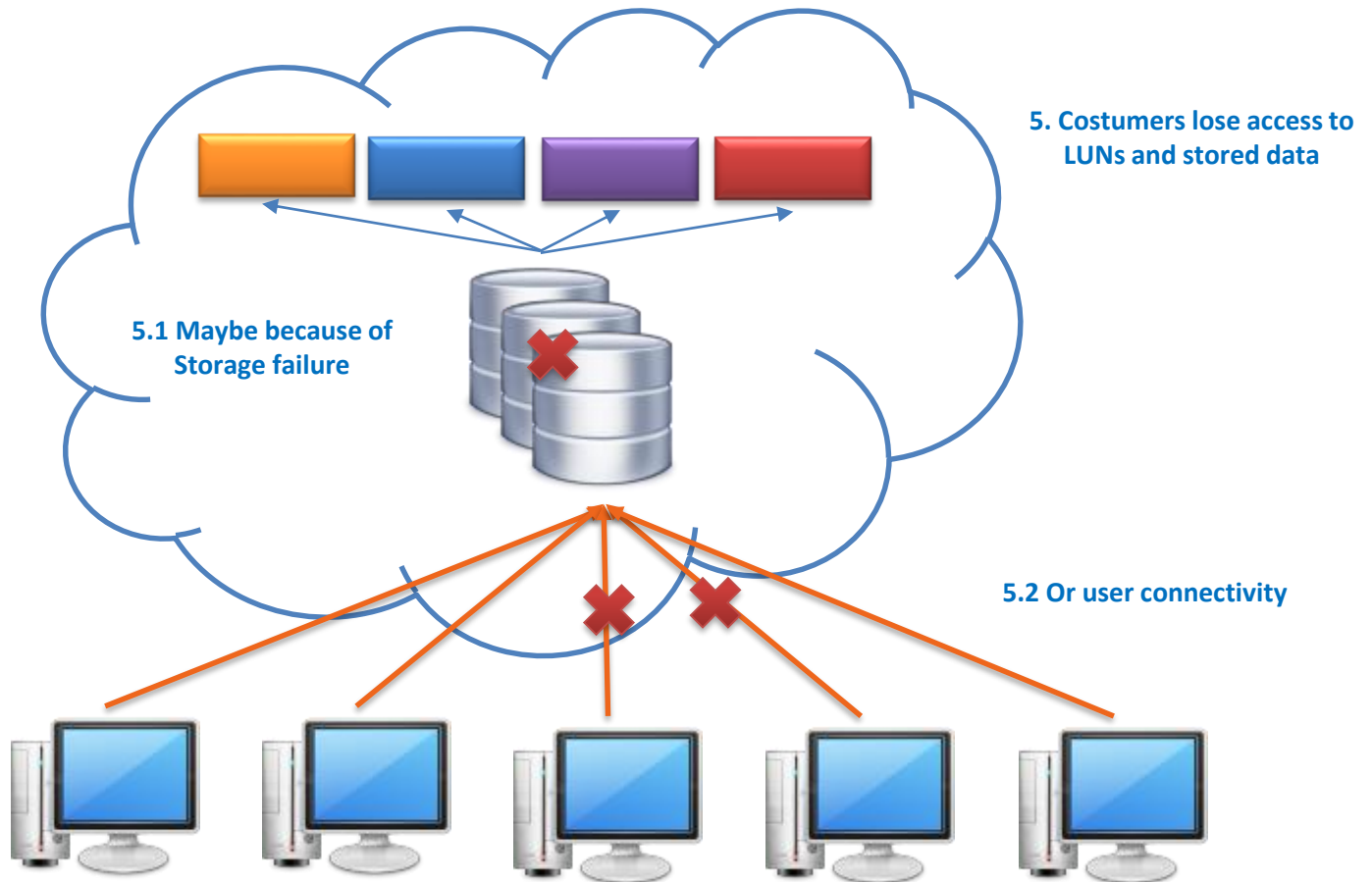
Redundant Storage Pattern

- **Problem:**



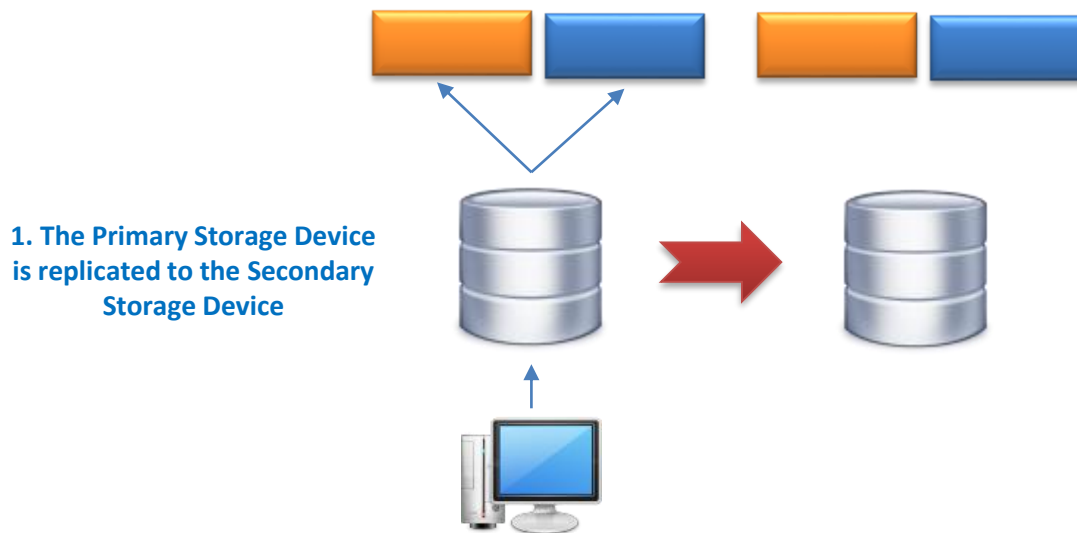
Redundant Storage Pattern

- **Problem:**



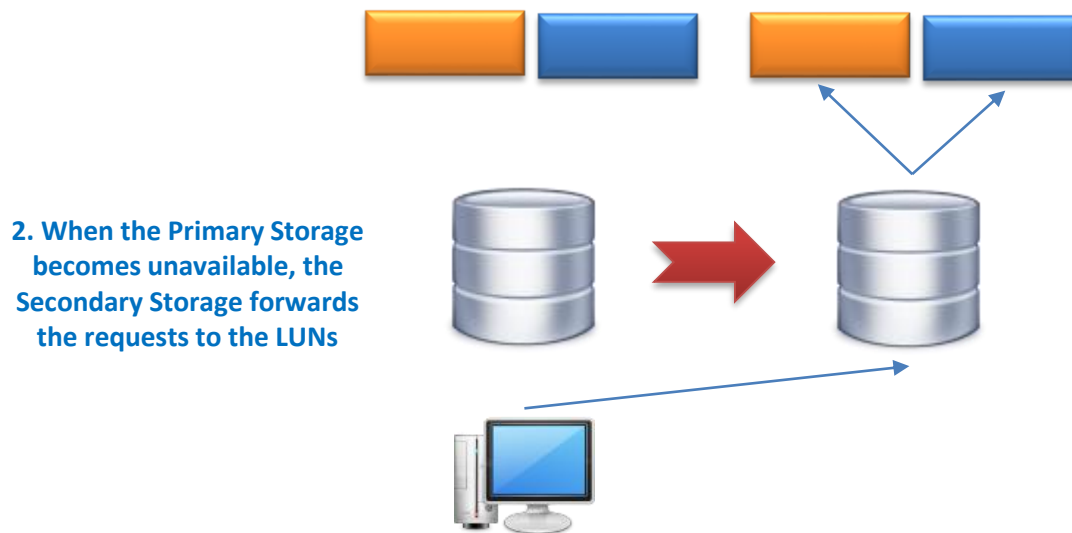
Redundant Storage Pattern

- **Solution:**
 - A *Secondary Redundant Storage Device*
 - *Synchronizes its data with the data in the primary cloud storage device*



Redundant Storage Pattern

- **Solution:**
 - A *Secondary Redundant Storage Device*
 - *Synchronizes its data with the data in the primary cloud storage device*

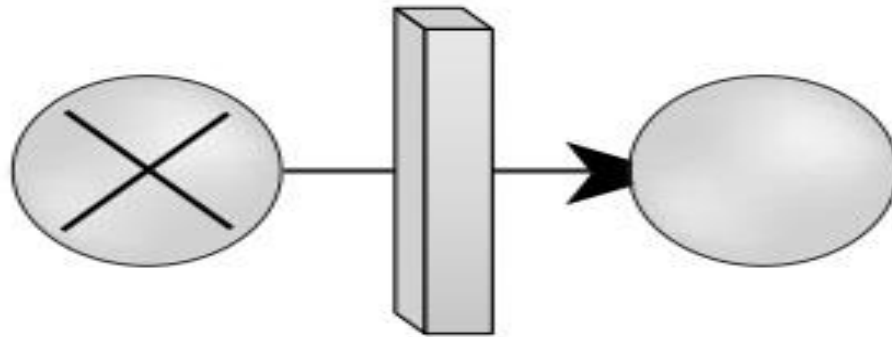


Redundant Storage Pattern

- **Application:**
 - Uses *Resource Replication Mechanism* to keep the primary storage device synchronized
 - Usually locate Secondary cloud storage devices in *a different geographical region* than the primary one
 - May dictate the protocol of synchronization
 - Some replication transport protocols have *Distance Restrictions*

Dynamic Failure Detection and Recovery

- **Goals:**
 - Improve and Auto Correct **failures**





Dynamic Failure Detection and Recovery

- *Problem:*
 - cloud-based IT resources **Fail**
 - **manual** intervention unacceptable

Dynamic Failure Detection and Recovery

- *Solution:*
 - Watchdog Monitor
 - Watchdog System



WATCHDOG SYSTEM®

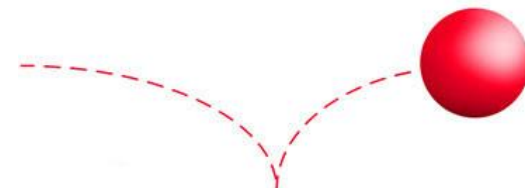
Dynamic Failure Detection and Recovery

- *Application:*
 - Intelligent Watchdog Monitor
 - Resilient Watchdog System



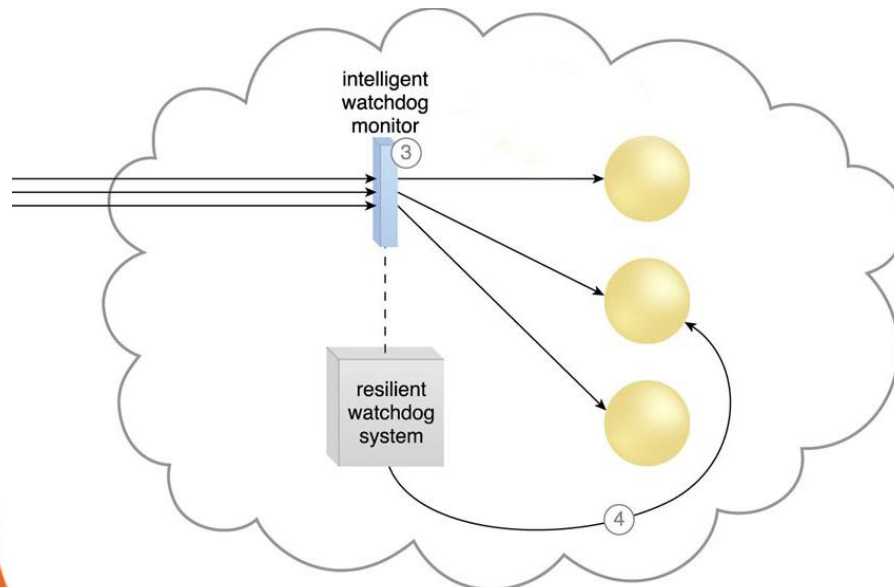
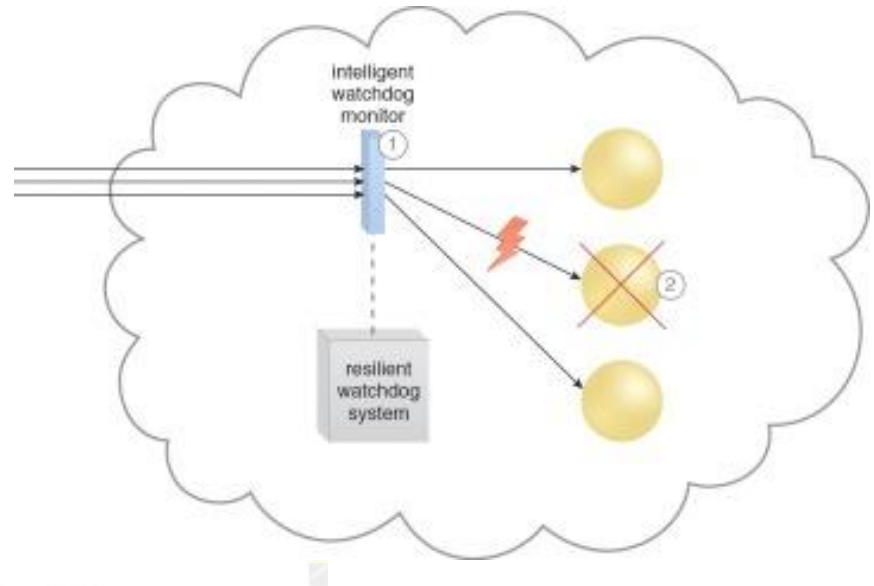
Intelligent

Resilience



Dynamic Failure Detection and Recovery

- ***Application:***



Dynamic Failure Detection and Recovery

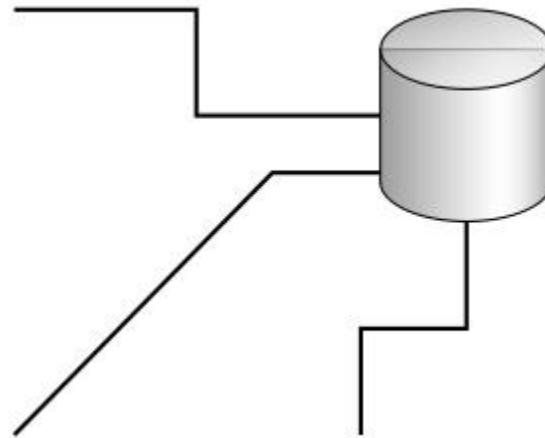
- *Application:*



- **Watching**
- Deciding upon an event
- **Acting upon an event**
- Reporting
- Escalating

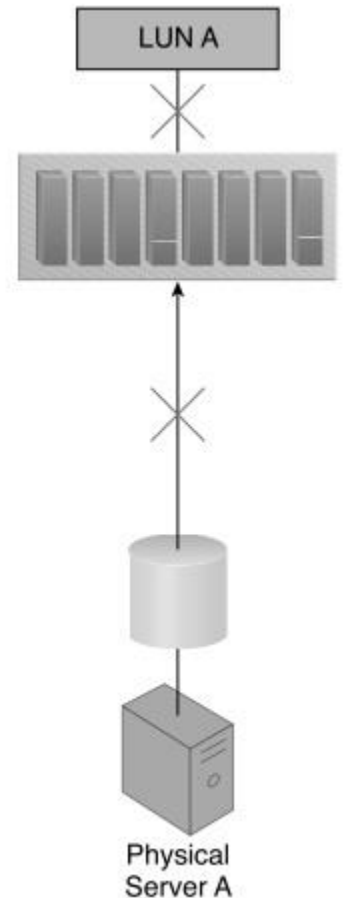
Multipath Resource Access

- **Goals:**
 - Access to the Source is **Unavailable** or **Lost**



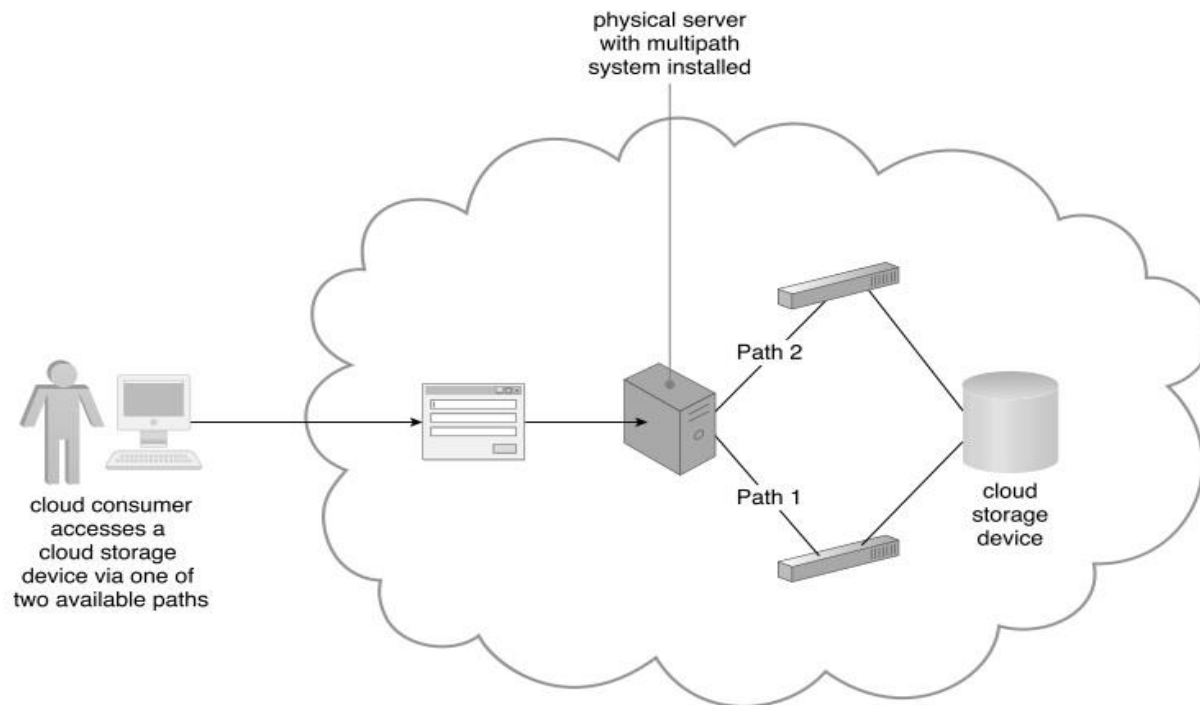
Multipath Resource Access

- **Problem:**
 - Access to resources
Inadvertently lost path
Incorrectly defined path



Multipath Resource Access

- ***Solution:***
 - **Alternative** paths (multi patching system)
Programmatically
Manually



Multipath Resource Access

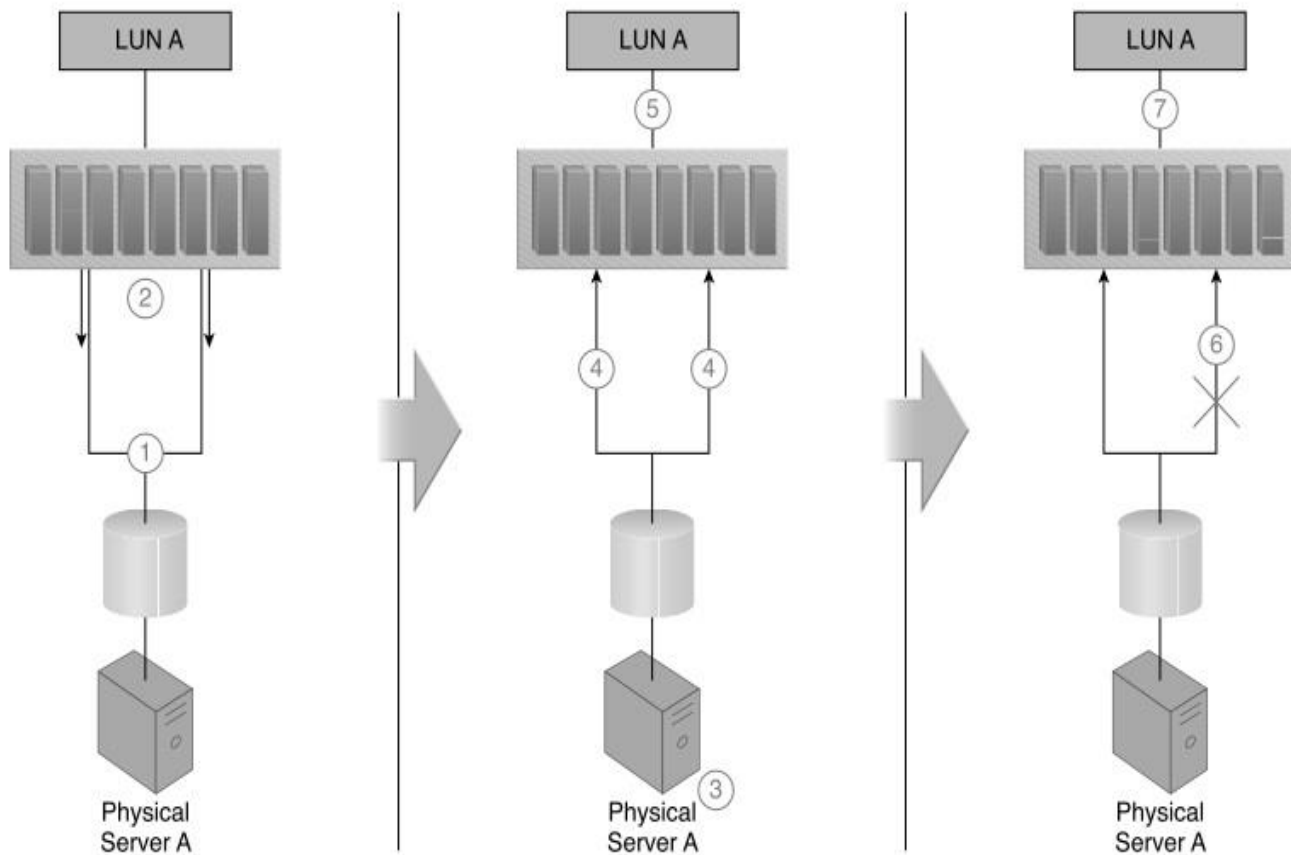
- **Application:**
 - **Multi pathing** system
Resides on the Server
hypervisor



Enable **Intel VT-X** in
BIOS or UEFI Firmware

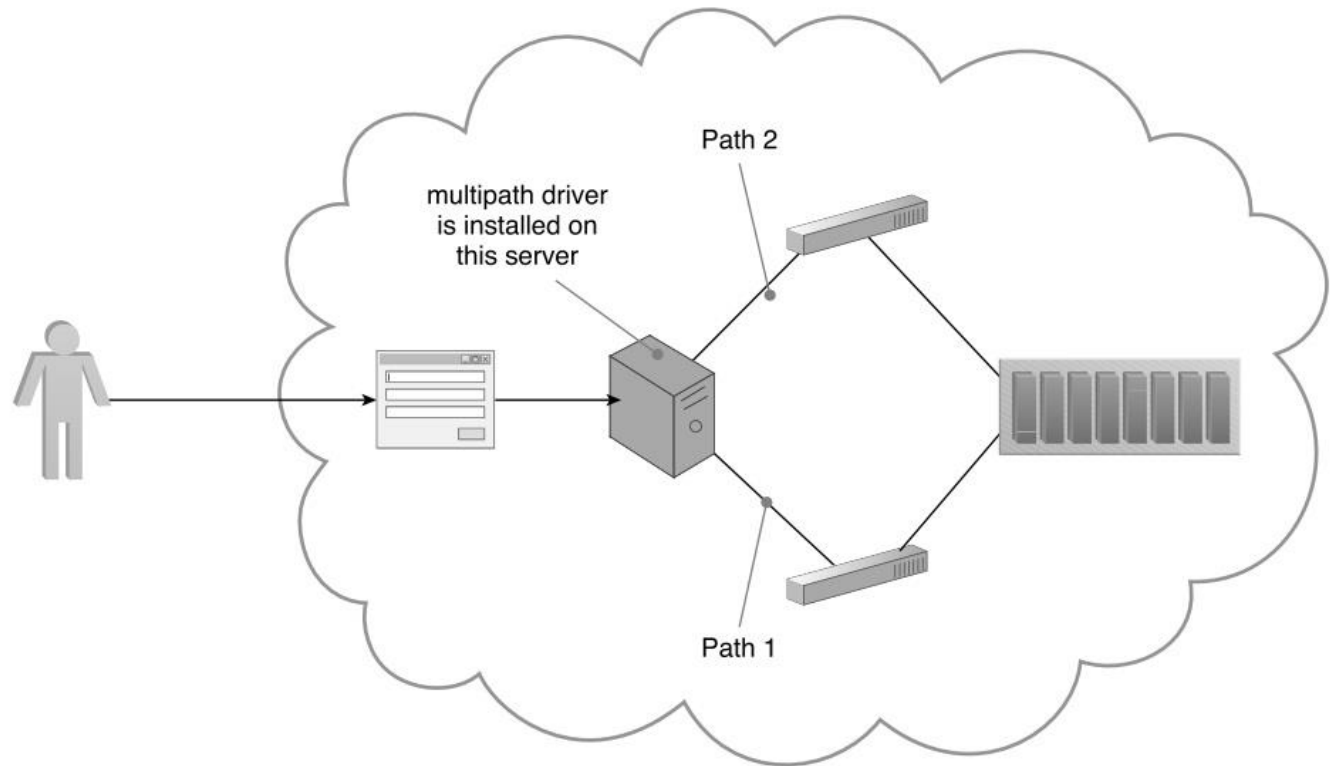
Multipath Resource Access

- Application:***



Multipath Resource Access

- *Application:*
 - **Driver** Installed On Server



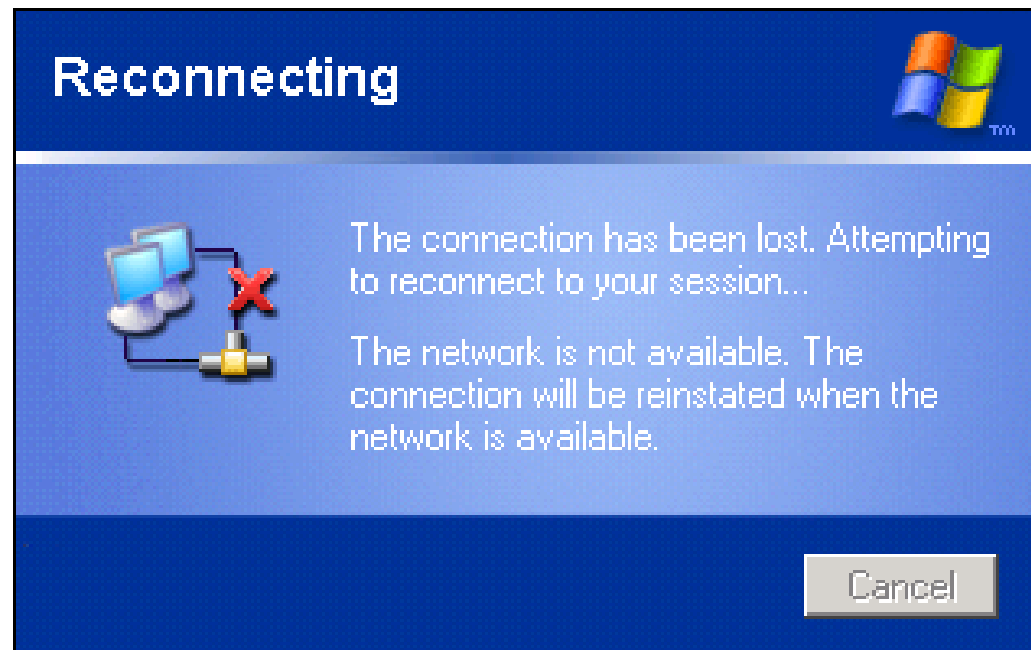
Redundant Physical Connection for Virtual Servers

- **Goals:**
 - **Uncertainty associated with the network**
Cable disconnection or Port failure



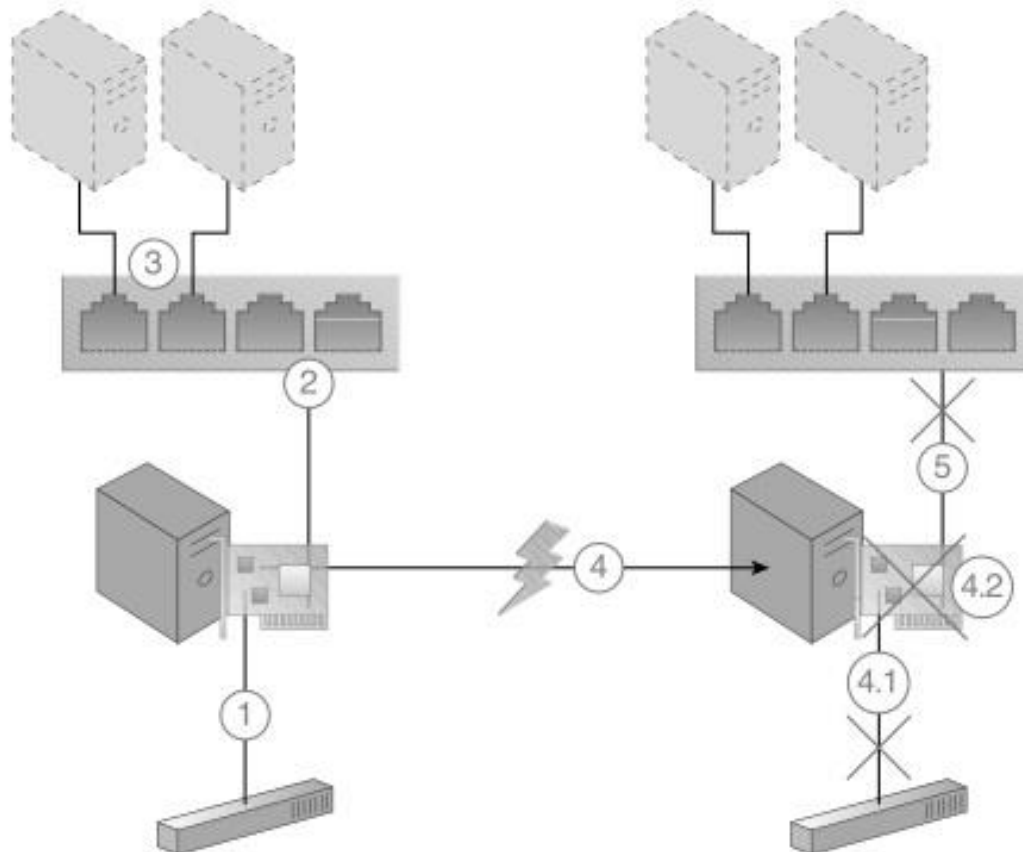
Redundant Physical Connection for Virtual Servers

- **Problem:**
 - virtual server :
isolated or Disconnect



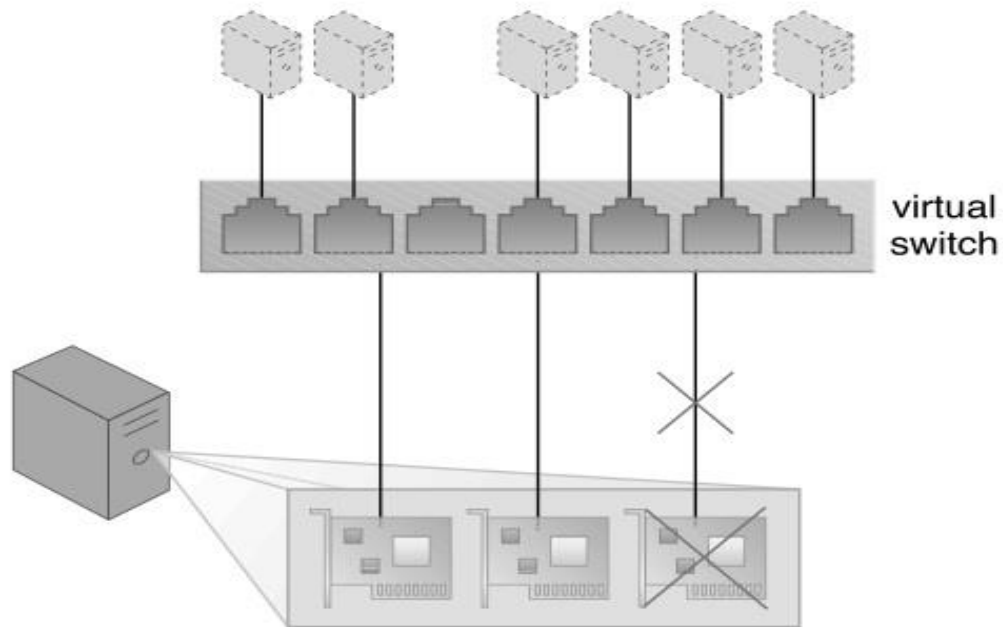
Redundant Physical Connection for Virtual Servers

- *Problem:*



Redundant Physical Connection for Virtual Servers

- *Solution:*
 - physical **backup** network connection



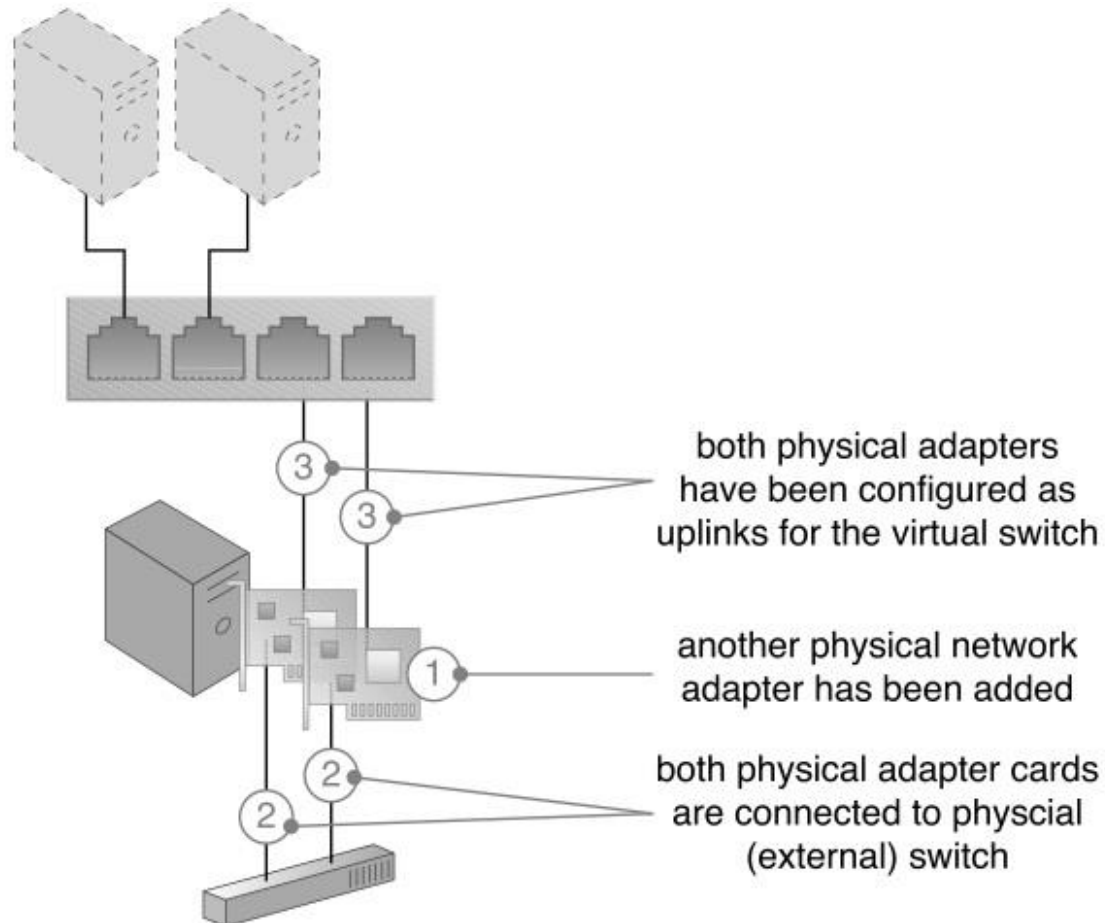
Redundant Physical Connection for Virtual Servers

- ***Application:***
 - A **second** Physical Network Card



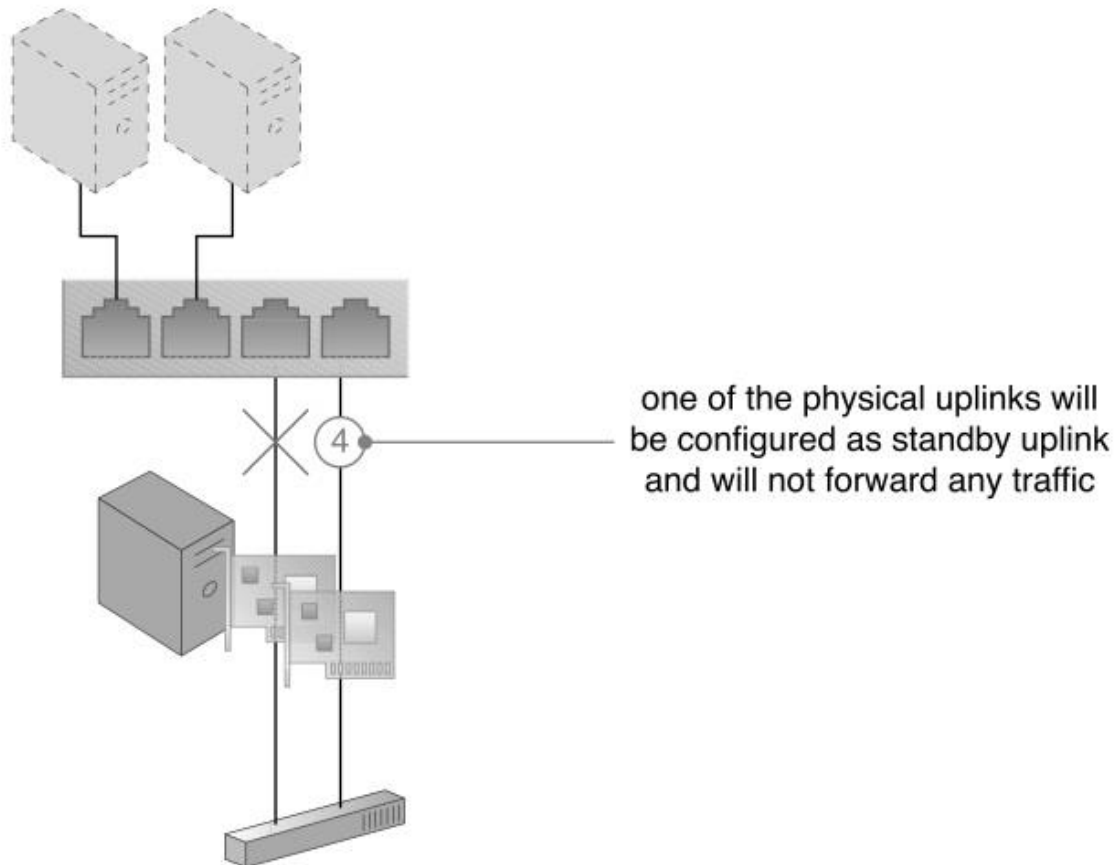
Redundant Physical Connection for Virtual Servers

- ***Application:***

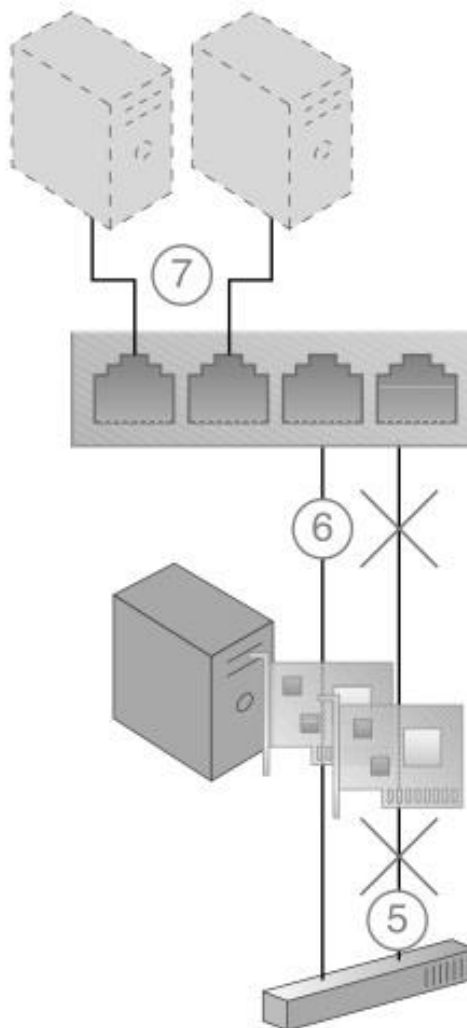


Redundant Physical Connection for Virtual Servers

- ***Application:***

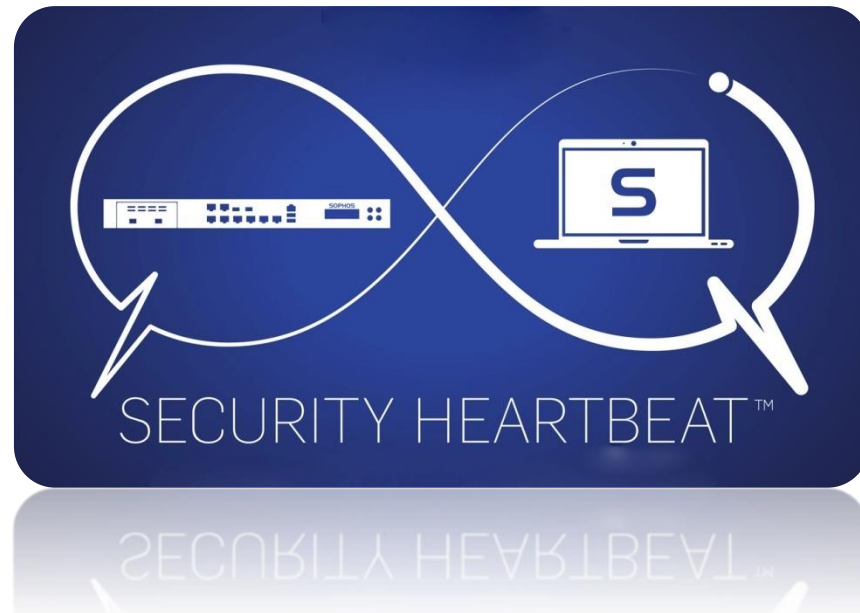


- ***Application:***



Synchronized Operating State

- **Goals:**
 - Use **high availability** and **clustering technology** by the consumer



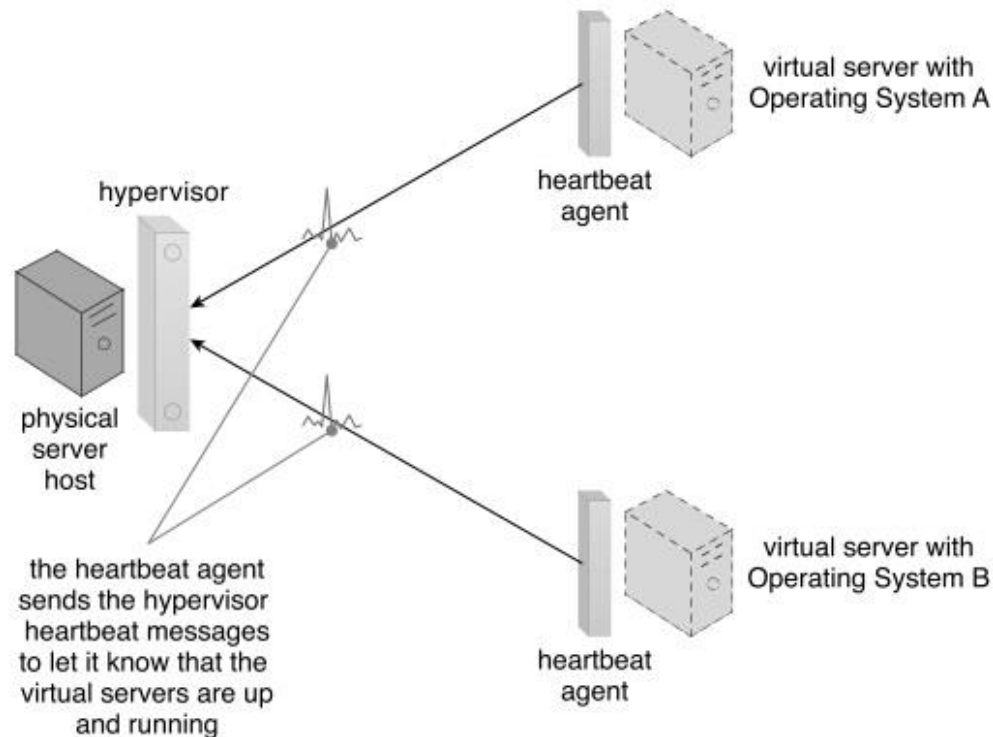
Synchronized Operating State

- **Problem:**
 - Lack of productivity of the consumer
 - Technical restrictions
 - licensing restrictions
 - other reasons



Synchronized Operating State

- **Solution :**
 - Use of **Heartbeat** Message

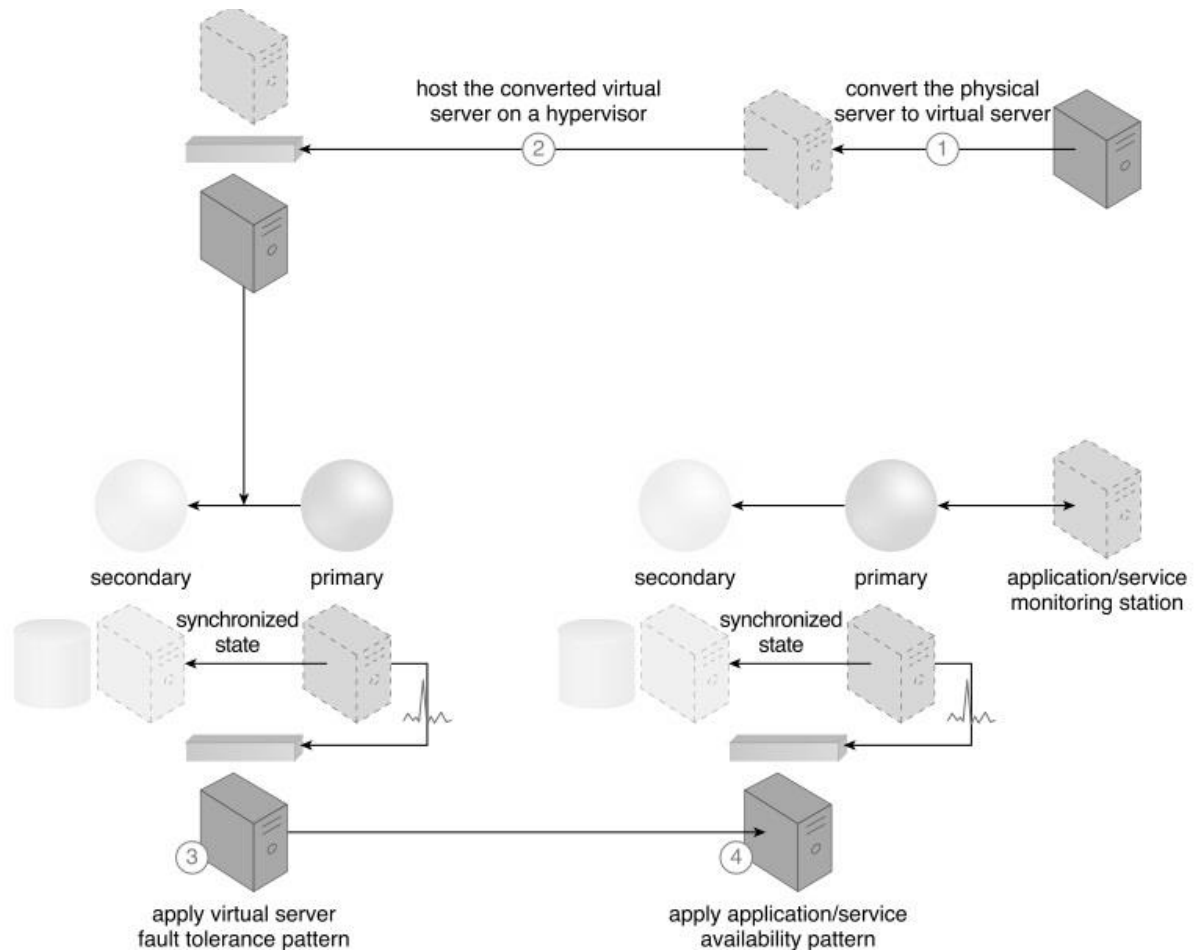


Synchronized Operating State

- *Application :*
 - Cloud Storage Device
 - Failover System
 - Hypervisor
 - Resource Replication
 - State Management Database
 - Virtual Server

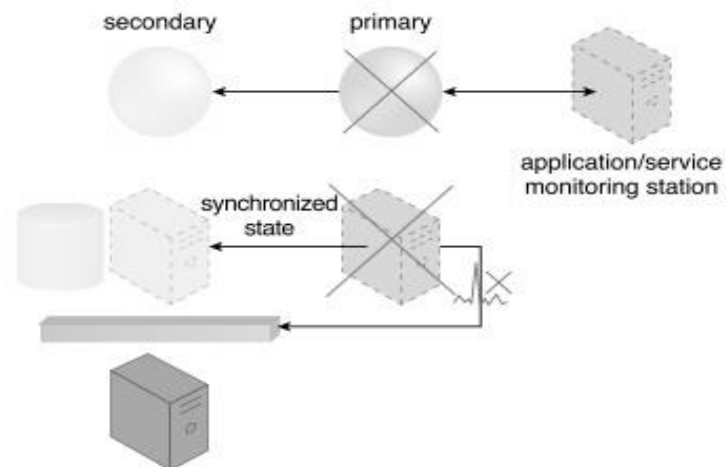
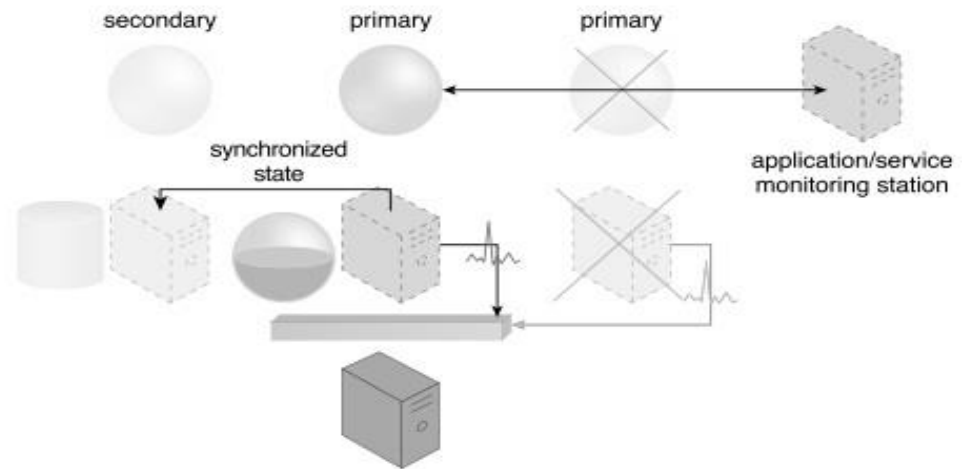
Synchronized Operating State

- **Application :**



Synchronized Operating State

- Application :***



References

1. Cloud Computing Design Patterns By Robert Cope and Amin NaserPour
2. Lynda.com (Learning Cloud Computing Core Concepts By David Linthicum)
3. Design Pattern in Cloud Computing , 2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing

