

Introduction to Cloud Computing and Virtualization

Talk Layout

錠 Cloud Computing

- 錠 Need
- 錠 Features
- 錠 Feasibility

錠 Virtualization of Machines

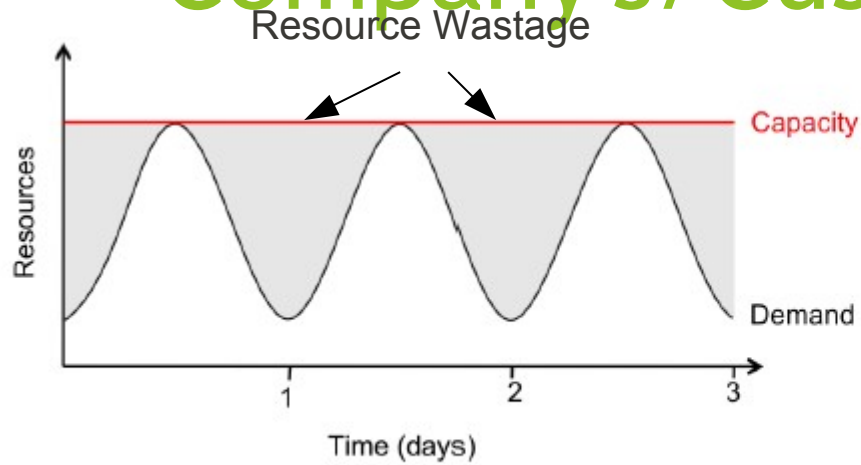
- 錠 What is it
- 錠 Implementation
- 錠 techniques Benefits

錠 XEN's internals

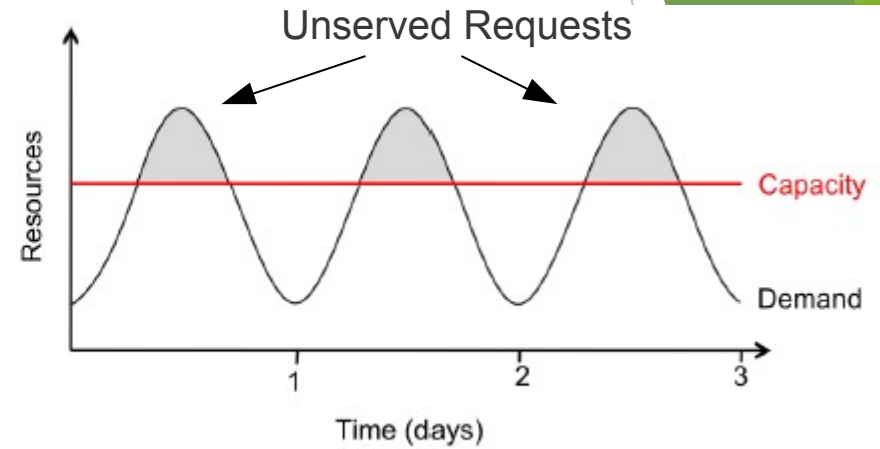
- 錠 Domains
- 錠 CPU Sharing
- 錠 HyperCall
- 錠 Memory Sharing
- 錠 IO Sharing

錠 Conclusion

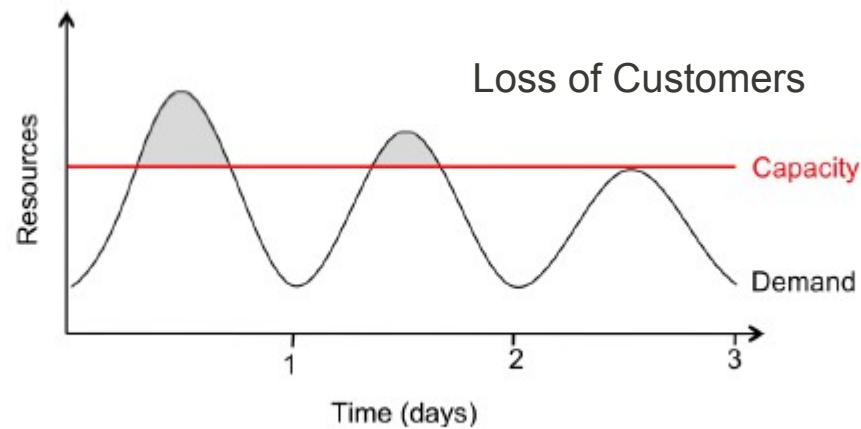
Resource Provisioning - Company's/Customer's View



(a) Provisioning for peak load



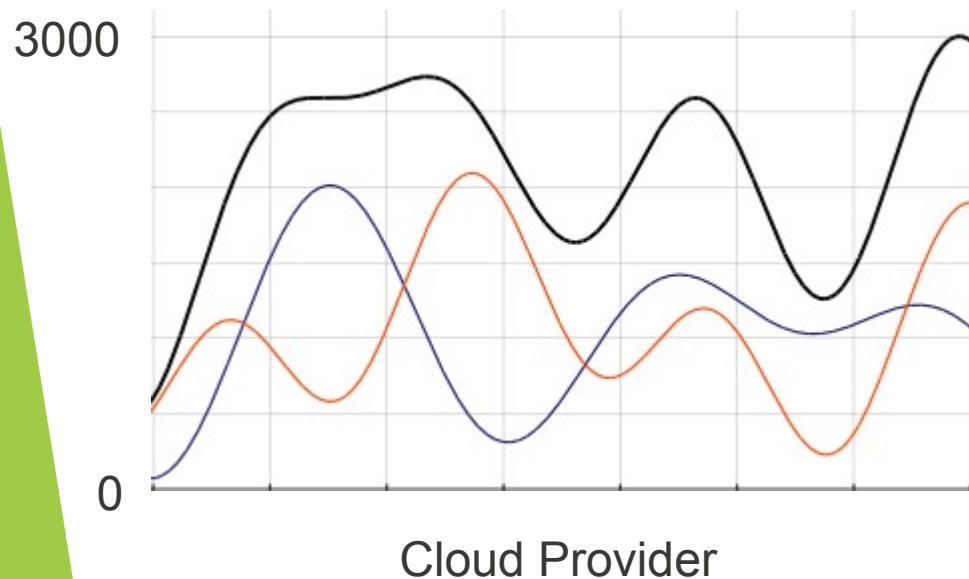
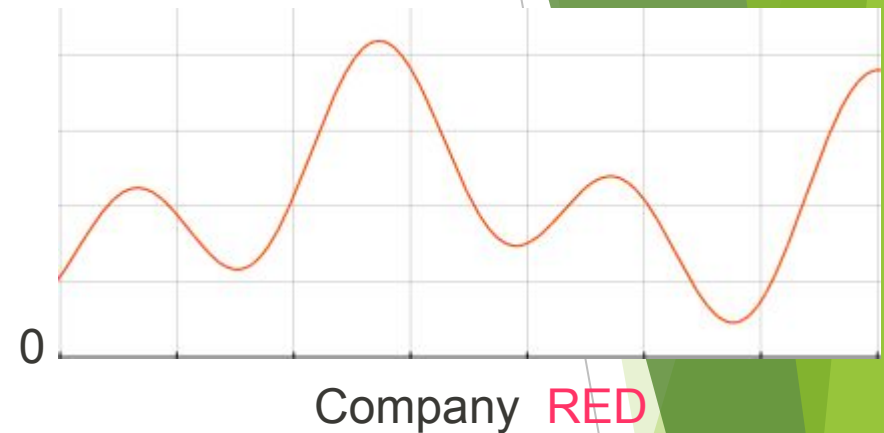
(b) Underprovisioning 1



(c) Underprovisioning 2

Big Headache

Resource Provisioning - DataCenter/Cloud Provider's View



Charge for 4050 machines, Work with 3000

Good Business

Computing as a service or
utility.

Cloud Computing

scale your infrastructure **on demand** within minutes or even seconds, instead of days or weeks, thereby **avoiding underutilization (idle servers) and overutilization**

a broad array of webbased services aimed at allowing users to obtain a wide range of functional capabilities on a '**payasyougo**' basis

Cloud computing really is accessing resources and services needed to perform functions with **dynamically changing needs**. ... The cloud is a **virtualization of resources that maintains and manages itself**.

- 罳 On-demand service : User not worried about maintenance and setup issues etc.
- 罳 Networked Shared Resources : Large capacity distributed/multiplexed over several users
- 罳 Flexible Provisioning : Dynamically scale resources
- 罳 Fine-grained metering : pay-as-you-use model

What is required

By Cloud Provider

Fast scalability – Quick addition and removal of servers

Service to customers should not be denied.

SLA should not be Violated

Efficient Resource Utilization

Constraints with physical machines

High Provisioning time.

Lower Resource Utilization.

Space, Power, Cooling.

Low fault tolerance

Less Isolation - misbehaving application can affect all others.

High downtime.

What is Virtualization

- ▶ Wikipedia says “Virtualization, in computing, is the creation of a virtual (rather than actual) version of something, such as a hardware platform, operating system, a storage device or network resources”

💡 Concept is not new.

Multi Programming – Each Process thinks it has complete control on all of the resources.

💡 Virtual Memory

💡 CPU Sharing

Similarities and Differences with Multiprogramming

Multi Programming

CPU is shared among processes

Memory is shared using Page Tables.

Process knows it is being managed- uses system calls.

Virtualization

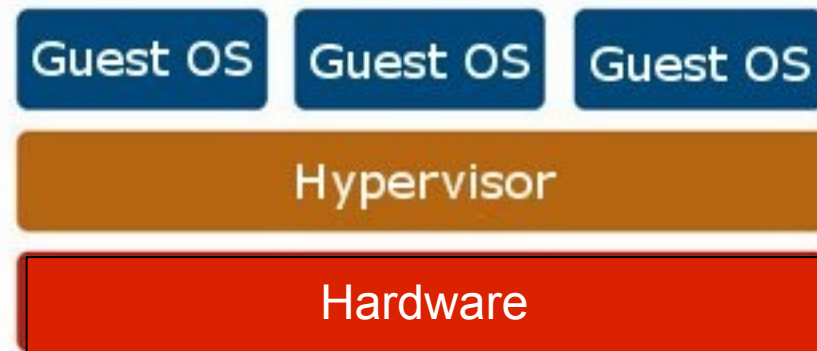
CPU is shared among OSs.

Memory is shared using more level of indirections. Multiple Page tables.

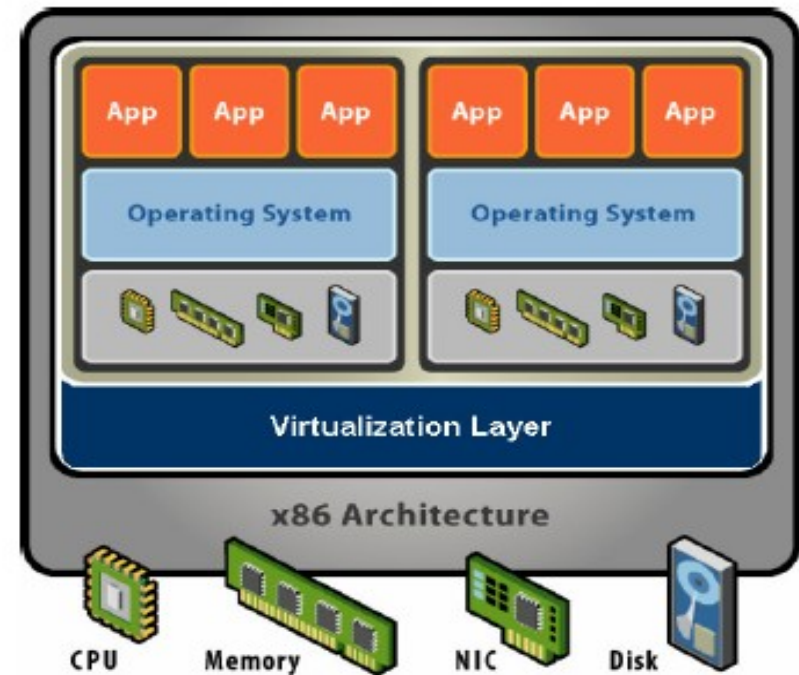
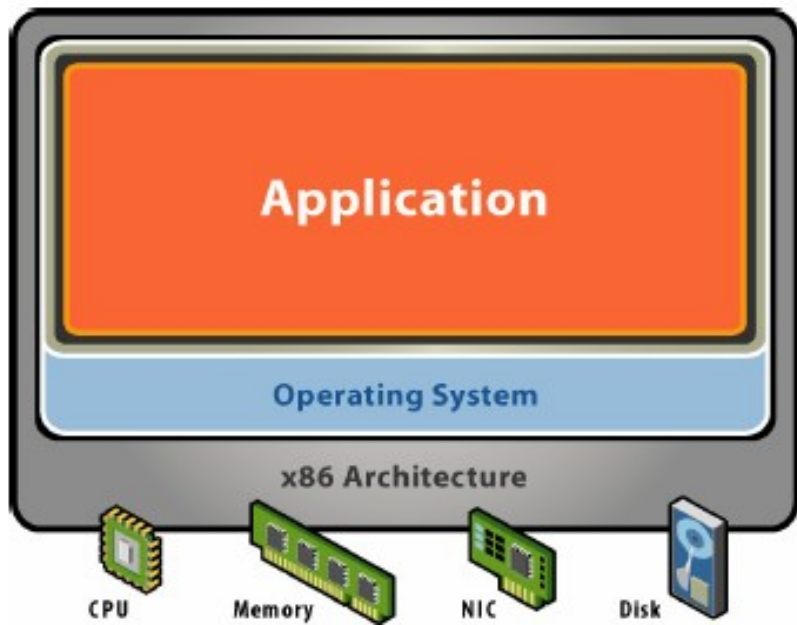
OS may or may not know that it is being managed.

Virtualization Architecture

- OS assumes complete control of the underlying hardware.
- Virtualization architecture provides this illusion through a hypervisor/VMM.
- Hypervisor/VMM is a software layer which:
 - Allows multiple Guest OS (Virtual Machines) to run simultaneously on a single physical host
 - Provides hardware abstraction to the running Guest OSs and efficiently multiplexes underlying hardware resources.



Physical vs. Virtual Machine



- Single OS
- h/w + s/w tightly coupled
- Application crashes affect all
- Resource under-utilization

- Machine view to OS is independent of hardware
- Multiple OS (isolated apps)
- Safely multiplex resources across VMs

Types of Virtual Machines

Process view of machine

- ▶ memory, user-level instr., system calls for OS functions
- ▶ OS interface to hardware defines view of process
- ▶ **Process VM**
- ▶ e.g. Java, .Net, Emulators

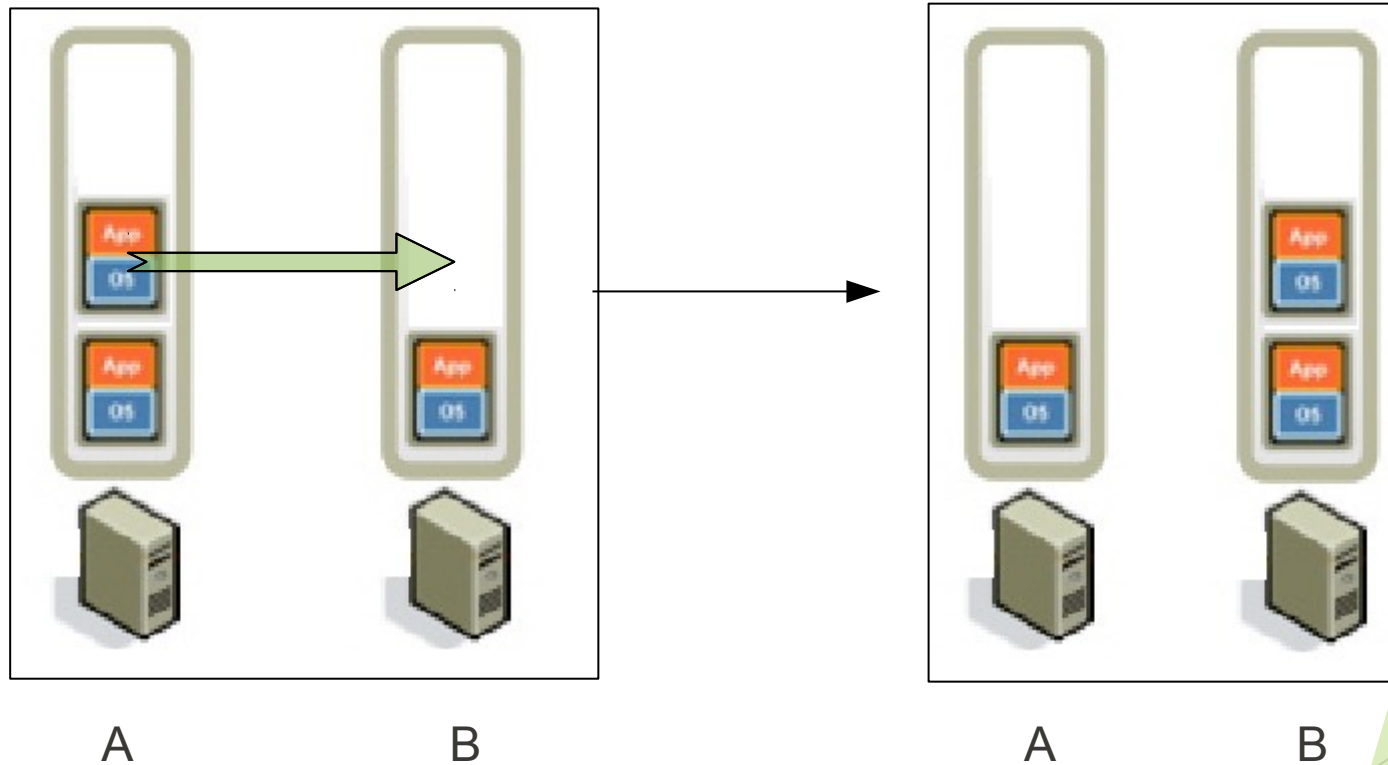
System view of machine

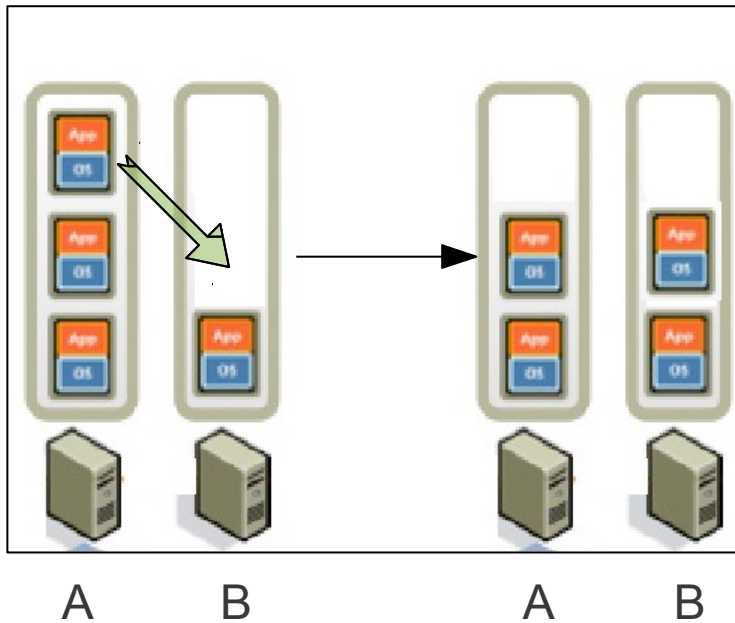
- ▶ environment to support multiple processes
- ▶ sharing resources
- ▶ hardware characteristics defines system view
- ▶ **System VM**
- ▶ e.g., Xen, kvm, VMware, VirtualBox, UMLinux

Benefits of using Virtual Machines

- Instant provisioning - fast scalability
- Live Migration is possible
- Load balancing and consolidation in a Data Center is possible.
- Low downtime for maintenance
- Virtual hardware supports legacy operating systems efficiently
- Security and fault isolation

VM Migration

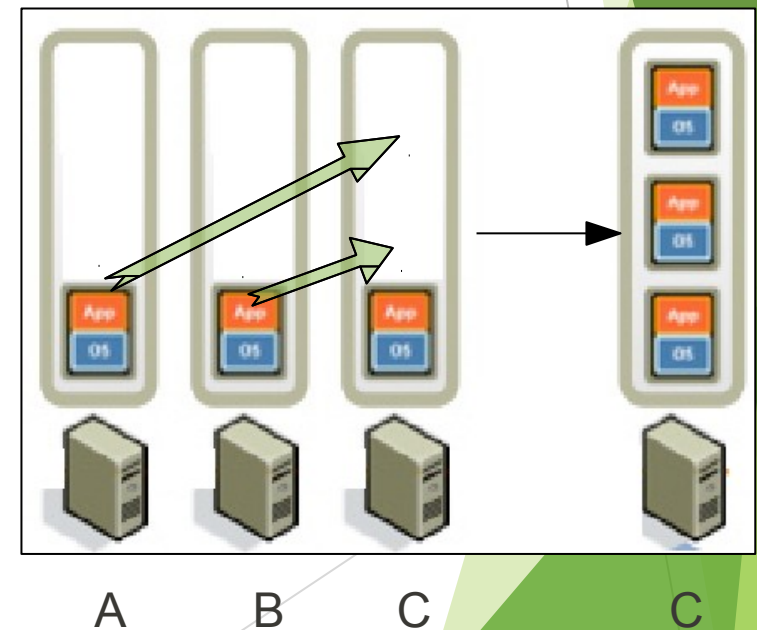




Load Balancing
Better Response time

Consolidation

Reduces number of Physical
Machine requirement



Importance of Virtualization in Cloud Computing

Cloud can exist without Virtualization, although it will be difficult and inefficient.

Cloud makes notion of “Pay for what you use”, “infinite availability- use as much you want”.

These notions are practical only if we have
lot of flexibility

efficiency in the back-end.

This efficiency is readily available in Virtualized Environments and Machines.

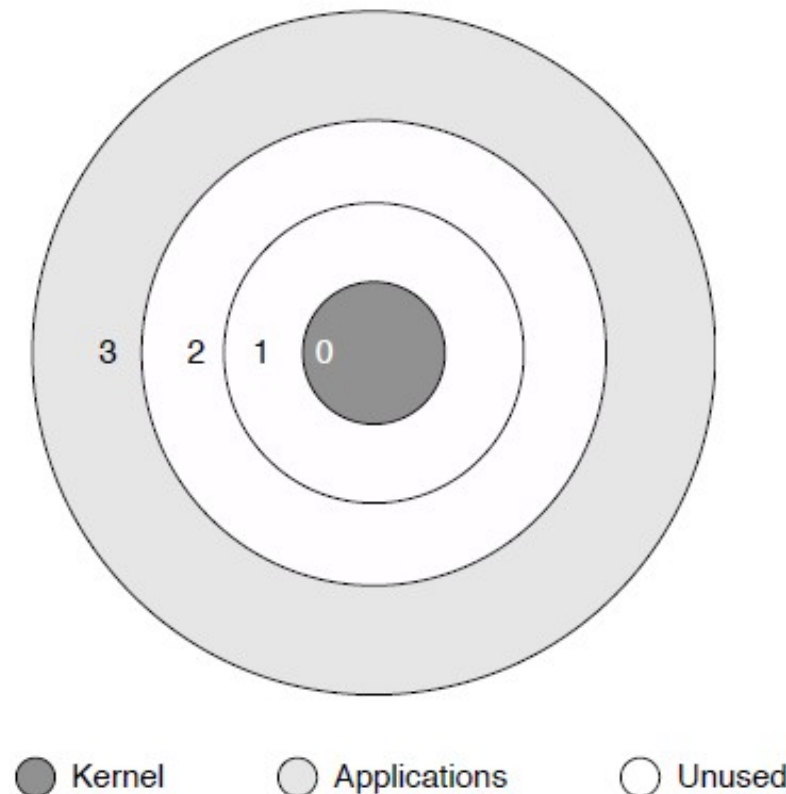
Requirement for Virtualizability

Popek and Goldberg mentioned a set of requirements that must be met in their 1974 paper.

- They divided instructions into three categories:
 - Privileged instructions: execute in a privileged mode, but will trap otherwise.
 - Control sensitive instructions: attempt to change the config of resources
 - Behavior sensitive instructions: are those that behave in a different way depending on the config of resources
- They said that all sensitive instructions must also be privileged instructions.
- Hypervisor must be able to intercept any instructions that changes the state of the machine in a way that impacts other processes.

Privilege Rings

- Memory page has a 2 bit code which is checked by CPU before executing the instruction.
- If privilege level is insufficient the CPU does not execute the instruction.



0 – Highest Privilege

VM Implementation Techniques

1. Binary Translation
2. Paravirtualization
3. Hardware Supported Virtualization

1. Binary Translation

Used in VMWare

Binary image of OS is manipulated at the runtime.

Privileged instructions are rewritten to point to their emulated versions.

Performance from this approach is not ideal, particularly when doing anything I/O intensive.

Caching of the locations of unsafe instructions can speed Up

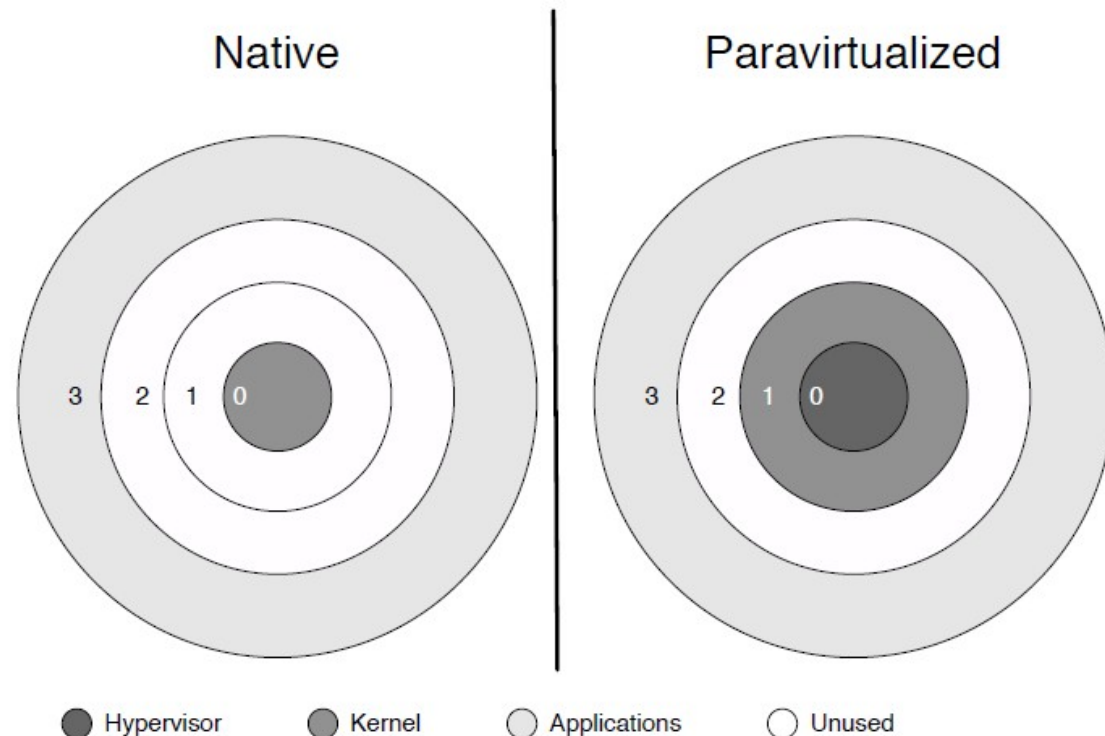
2. Paravirtualization

Used in XEN

Make OS aware of underlying Virtualization env.

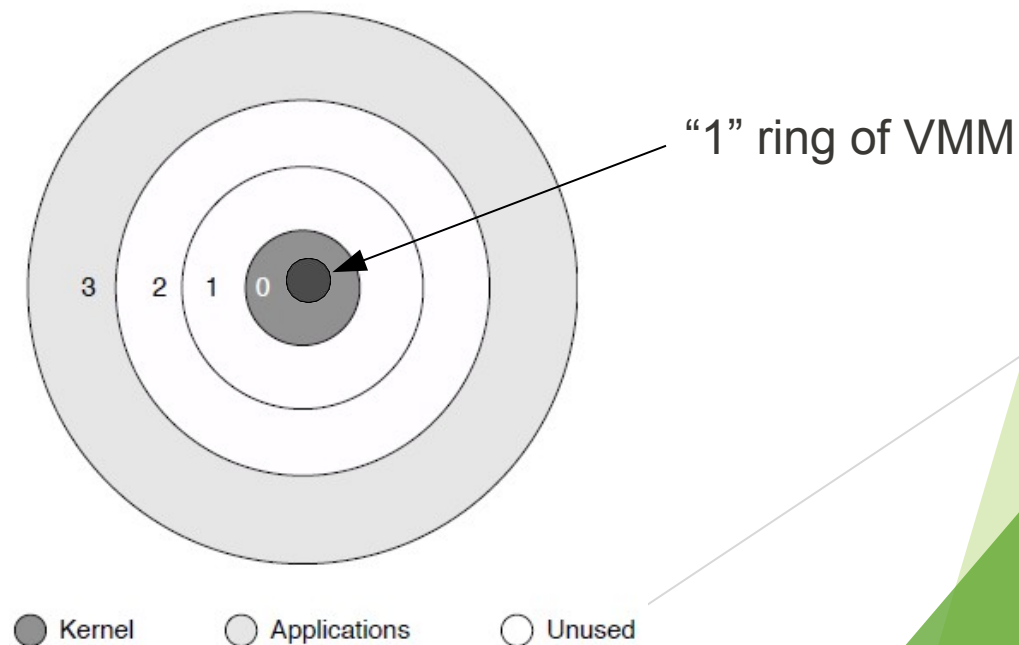
OS's code is manipulated.

Important system calls are changed to point to the implementation provided by the VMM.



3. HW Supported Virtualization

- Added new instructions which makes Virtualization considerably easier for x86.
 - Intel – IVT(Intel Virtualization Technology)
 - AMD – introduced AMD-V
- OS stays in its original privilege level 0.
- Attempts to access the hardware directly are caught and passed to VMM. In other words a new privilege ring is setup for the VMM.



XEN

- 罳 XEN Domains
- 罳 CPU Sharing
- 罳 Hyper Calls
- 罳 Memory Sharing
- 罳 IO Sharing
- 罳 XEN Split Driver Technique
- 罳 IO Ring

XEN Domains

鯨 Xen runs guests in environments known as domains which encapsulate a complete running virtual environment

鯨 There are two types pf Domains:

鯨 DomU -

鯨 the “U” stands for unprivileged.

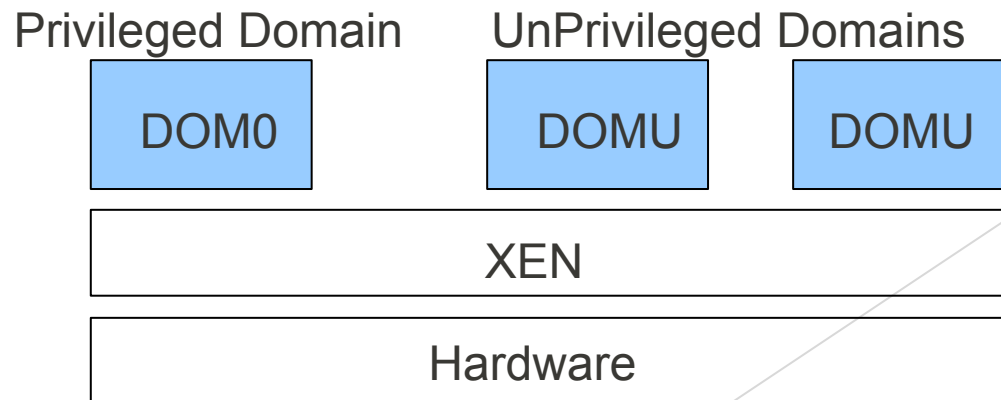
鯨 Guest OSs run in this domain.

鯨 Dom0

鯨 has elevated privileges

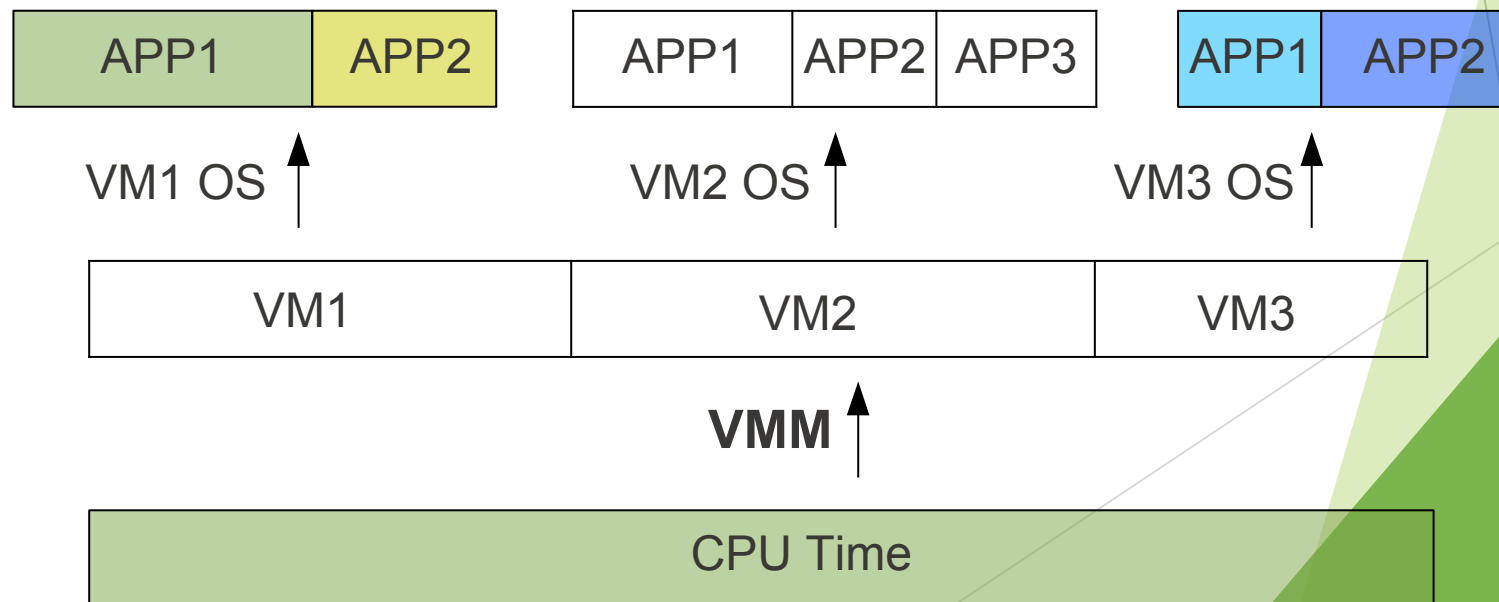
鯨 Provides device drivers

鯨 Provides tools/mechanisms to configure Virtualization environment

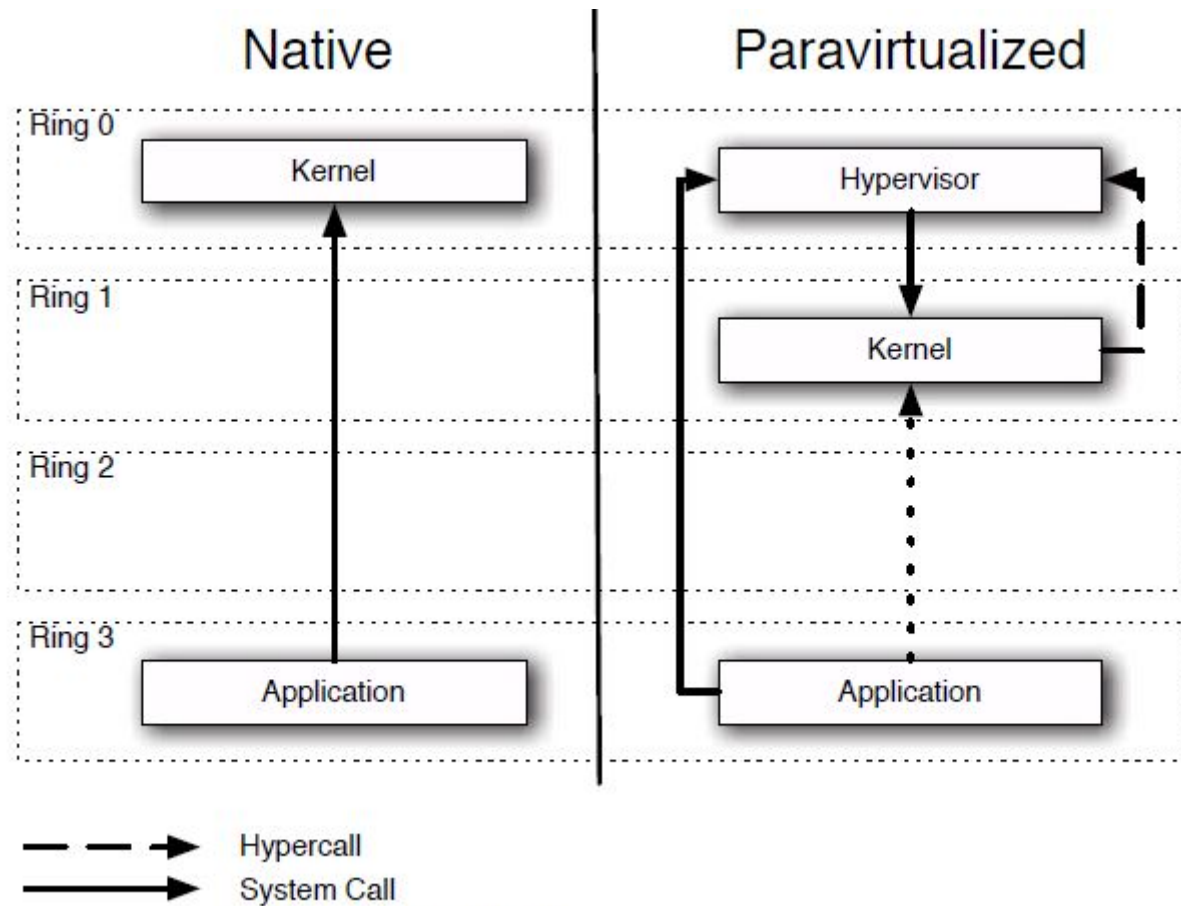


CPU Sharing

- VM or Hypervisor provides a virtual view of CPU to VMs.
- In multi processing, CPU is allotted to the different processes in form of time slices by the OS.
- Similarly VMM or Hypervisor allots CPU to different VMs.



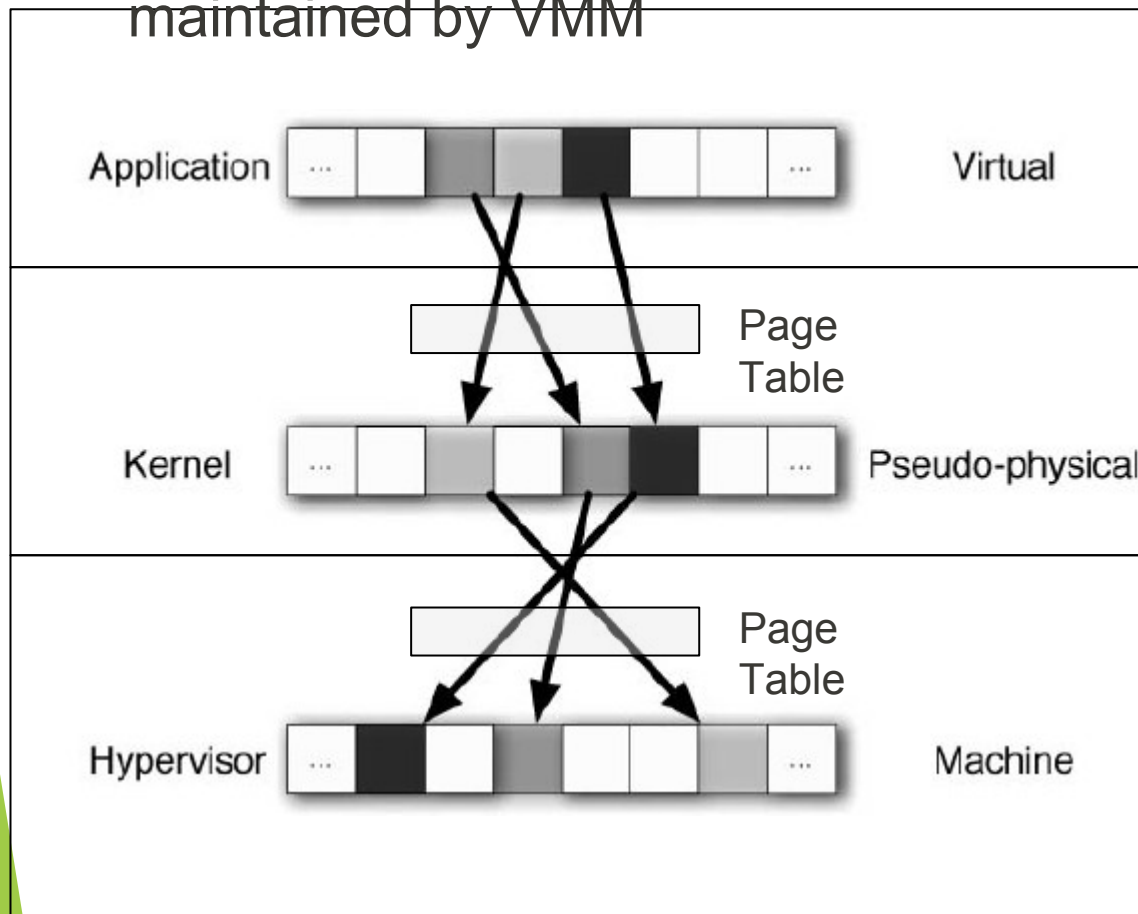
XEN Hypercall



Memory Sharing

In Multiprogramming there is a single level of indirection maintained by Kernel.

In case of Virtual Machines there is one more level of indirection maintained by VMM



Applications use Virtual Addresses

Kernel translates Virtual Addresses to PseudoPhysical Addresses

Hypervisor translates Pseudo Physical Addresses to Machine addresses

IO Sharing

鐳 DMA Problem

鐳 Device needs to use Physical Memory location.

鐳 In a virtualized environment, the kernel is running in a hypervisor-provided virtual address space

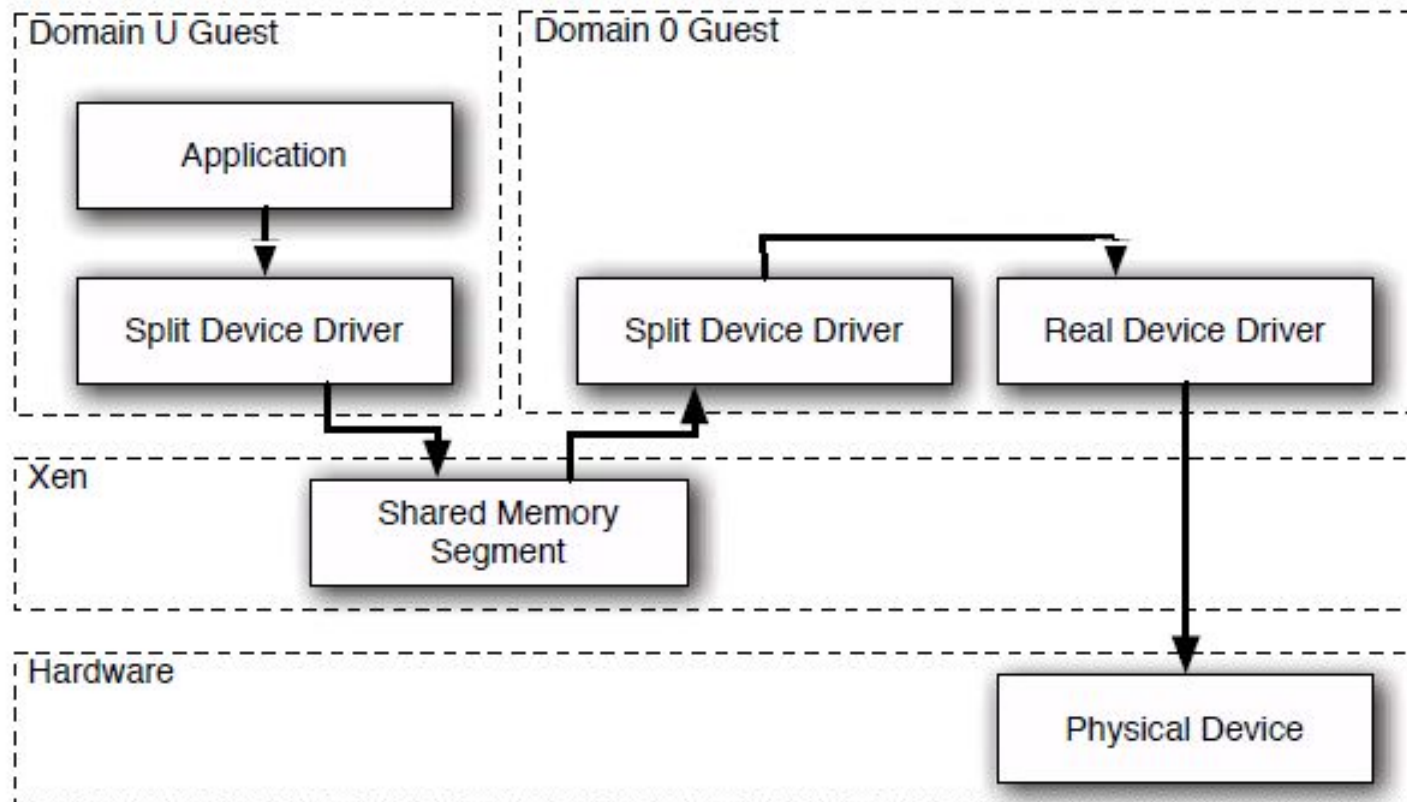
鐳 Allowing the guest kernel to convey an arbitrary location to device for writing is a serious security hole

鐳 Detecting a DMA instruction is nontrivial. Each device defines its own protocol for talking to drivers.

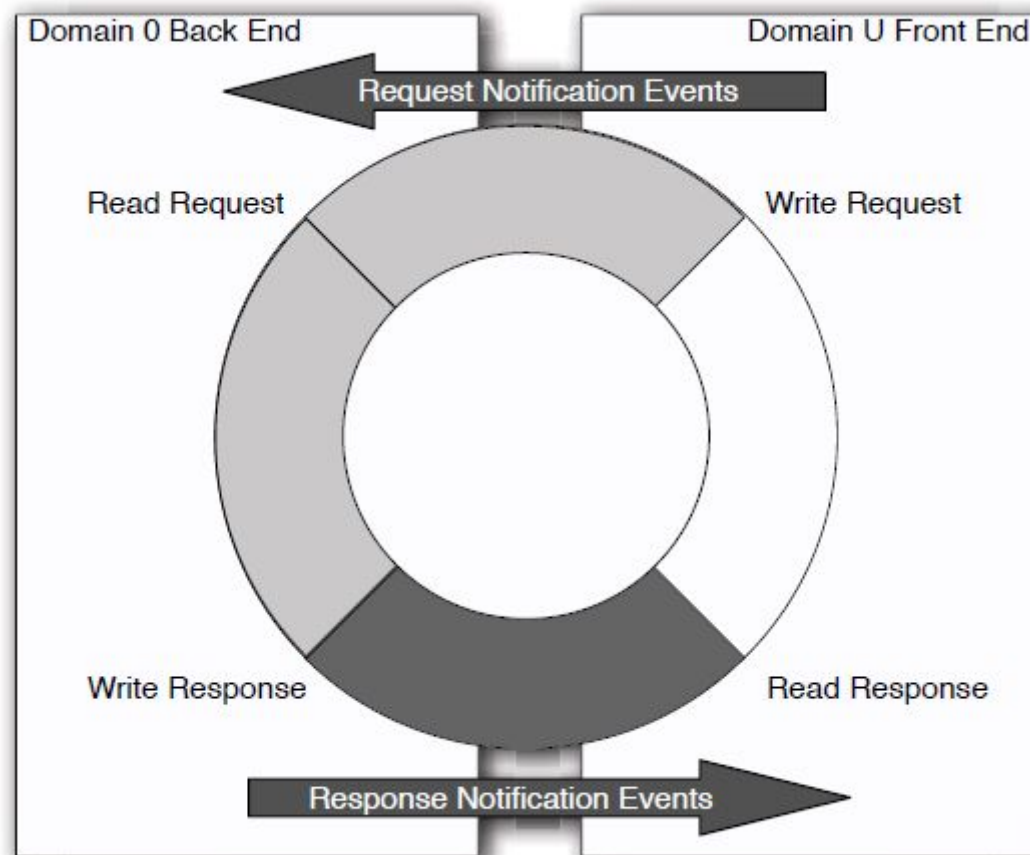
鐳 **XEN Follows Split Driver Model: Dom 0 does the IO on behalf of all the other guests.**

鐳 As DOM0 is privileged the IO has no problem

XEN IO Split Device Driver



Shared memory is used with event based synchronization



Conclusions

- Notion of Cloud is possible without Virtualization, but it will be inefficient and inflexible.
- Virtualization is an attempt to manage OS.
- There are many levels and many ways to implement Virtualization.

References

Amburst et al. “Above the Clouds: A Berkeley view of cloud computing”

David Chisnall. “The Definitive Guide to XEN Hypervisor”.

Prof. Purushottam Kulkarni, CSE, IITB, His Presentation.

Vmware “www.vmware.com”

Wikipedia and Internet.