- **Competition with Foreign (Local) Jobs**: Scheduling becomes more complicated when both cluster jobs and local jobs are running. The local jobs should have priority over cluster jobs.

1. **Migration Scheme IssuesNode Availability**: Can the job find another available node to migrate to?
    - Berkeley study : Even during peak hours, 60% of workstations in a cluster are available.

2. **Migration Overhead**: The migration time can significantly slow down a parallel job.
    - Berkeley study : a slowdown as great as 2.4 times.
    - Slowdown is less if a parallel job is run on a cluster of twice the size.
    - e.g. a 32-node job on a 60-node cluster – migration slowdown no more than 20%, even when migration time of 3 minutes.

3. **Recruitment Threshold**: the amount of time a workstation stays unused before the cluster considers it an idle node.

UNIT -2

Levels of Virtualization Implementation

- Virtualization is a computer architecture technology by which multiple virtual machines (VMs) aremultiplexed in the same hardware machine.
- After virtualization, different user applications managed by their own operating systems (guest OS) can run onthe same hardware independent of the host OS
- done by adding additional software, called a virtualization layer
- This virtualization layer is known as hypervisor or virtual machine monitor (VMM)

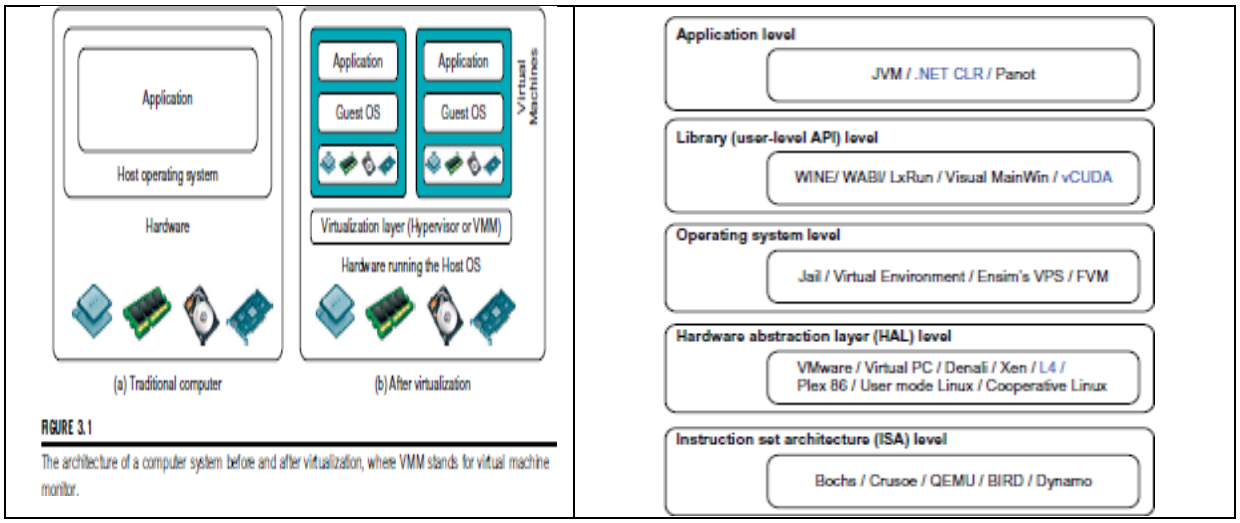

FIGURE 3.1

The architecture of a computer system before and after virtualization, where VMM stands for virtual machine monitor.

- function of the software layer for virtualization is to virtualize the physical hardware of a host machine into virtual resources to be used by the VMs
- Common virtualization layers include the instruction set architecture (ISA) level, hardware level, operating system level, library support level, and application level

Instruction Set Architecture Level

- At the ISA level, virtualization is performed by emulating a given ISA by the ISA of the host machine. For example, MIPS binary code can run on an x86-based host machine with the help of ISA emulation. With this approach, it is possible to run a large amount of legacy binary code written for various processors on any given new hardware host machine.
- Instruction set emulation leads to virtual ISAs created on any hardware machine. The basic emulation method is through code interpretation. An interpreter program interprets the source instructions to target instructions one by one. One source instruction may require tens or hundreds of native target instructions to perform its function. Obviously, this process is relatively slow. For better performance, dynamic binary translation is desired.
- This approach translates basic blocks of dynamic source instructions to target instructions. The basic blocks can also be extended to program traces or super blocks to increase translation efficiency.
- Instruction set emulation requires binary translation and optimization. A virtual instruction set architecture (V-ISA) thus requires adding a processor-specific software translation layer to the compiler.

Hardware Abstraction Level

- Hardware-level virtualization is performed right on top of the bare hardware.
- Thisapproach generates a virtual hardware environment for a VM.
- The process managesthe underlying hardware through virtualization. The idea is to virtualize a computer's resources, such asits processors, memory, and I/O devices.
- The intention is to upgrade the hardware utilization rate bymultiple users concurrently. The idea was implemented in the IBM VM/370 in the 1960s.
- Morerecently, the Xen hypervisor has been applied to virtualize x86-based machines to run Linux or otherguest OS applications.

Operating System Level

- This refers to an abstraction layer between traditional OS and user applications.
- OS-level virtualizationcreates isolated containers on a single physical server and the OS instances to utilize the hardwareand software in data centers.
- The containers behave like real servers.
- OS-level virtualization iscommonly used in creating virtual hosting environments to allocate hardware resources among alarge number of mutually distrusting users.
- It is also used, to a lesser extent, in consolidating serverhardware by moving services on separate hosts into containers or VMs on one server.

Library Support Level

- Most applications use APIs exported by user-level libraries rather than using lengthy system callsby the OS.
- Since most systems provide well-documented APIs, such an interface becomes anothercandidate for virtualization.
- Virtualization with library interfaces is possible by controlling the communicationlink between applications and the rest of a system through API hooks.

User-Application Level

- Virtualization at the application level virtualizes an application as a VM.
- On a traditional OS, anapplication often runs as a process. Therefore, application-level virtualization is also known as process-level virtualization.
- The most popular approach is to deploy high level language (HLL)VMs. In this scenario, the virtualization layer sits as an application program on top of the operatingsystem,
- The layer exports an abstraction of a VM that can run programs written and compiledto a particular abstract machine definition.
- Any program written in the HLL and compiled for thisVM will be able to run on it. The Microsoft .NET CLR and Java Virtual Machine (JVM) are twogood examples of this class of VM.

VMM Design Requirements and Providers

- layer between real hardware and traditional operating systems. This layer is commonly called the Virtual Machine Monitor (VMM)
- three requirements for a VMM
- a VMM should provide an environment for programs which is essentially identical to the original machine
- programs run in this environment should show, at worst, only minor decreases in speed
- VMM should be in complete control of the system resources.
- VMM includes the following aspects:
- (1) The VMM is responsible for allocating hardware resources for programs;
- (2) it is not possible for a program to access any resource not explicitly allocated to it;
- (3) it is possible under certain circumstances for a VMM to regain control of resources already allocated.

Virtualization Support at the OS Level

- Why OS-Level Virtualization? :
  - it is slow to initialize a hardware-level VM because each VM creates its own image from scratch.
- OS virtualization inserts a virtualization layer inside an operating system to partition a machine's physical resources.
- It enables multiple isolated VMs within a single operating system kernel.
- This kind of VM is often called a virtual execution environment (VE), Virtual Private System (VPS), or simply container
- The benefits of OS extensions are twofold:
  - (1) VMs at the operating system level have minimal startup/shutdown costs, low resource requirements, and high scalability;
  - (2) for an OS-level VM, it is possible for a VM and its host environment to synchronize state changes when necessary**.**

Middleware Support for Virtualization

- Library-level virtualization is also known as user-level Application Binary Interface (ABI) or API emulation.
- This type of virtualization can create execution environments for running alien programs on a platform

Hypervisor and Xen Architecture

- The hypervisor software sits directly between the physical hardware and its OS.
- This virtualization layer is referred to as either the VMM or the hypervisor

Xen Architecture

- Xen is an open source hypervisor program developed by Cambridge University.
- Xen is a microkernel hypervisor
- The core components of a Xen system are the hypervisor, kernel, and applications
- The **guest OS**, which has control ability, is called **Domain 0**, and the others are called **Domain U**
- Domain 0 is designed to access hardware directly and manage devices

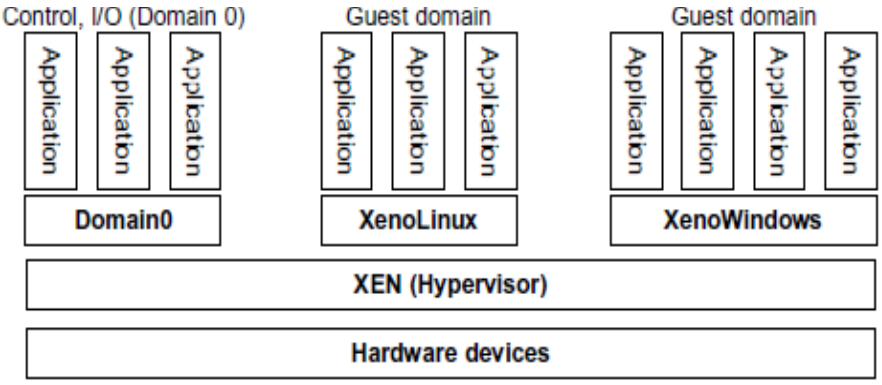

FIGURE 3.5

The Xen architecture's special domain 0 for control and I/O, and several guest domains for user applications.
(Courtesy of P. Barham, et al. [7])

- VM state is akin to a tree: the current state of the machine is a point that progresses monotonically as the software executes.
- VMs are allowed to roll back to previous states in their execution (e.g., to fix configuration errors) or rerun from the same point many times

Full virtualization

- Full virtualization, noncritical instructions run on the hardware directly while critical instructions are discovered and replaced with traps into the VMM to be emulated by software
- **VMware** puts the **VMM at Ring 0** and the **guest OS at Ring 1**.
- The VMM scans the instruction stream and identifies the privileged, control- and behavior-sensitive instructions.
- When these instructions are identified, they are trapped into the VMM, which emulates the behavior of these instructions.
- The method used in this emulation is called binary translation.
- Therefore, **full virtualization combines binary translation and direct execution.**

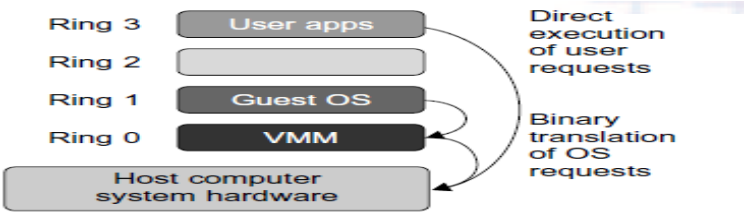

FIGURE 3.6

Indirect execution of complex instructions via binary translation of guest OS requests using the VMM plus direct execution of simple instructions on the same host.
(Courtesy of VM Ware [71])

Para-Virtualization

- Para-virtualization needs to modify the guest operating systems
- A para-virtualized VM provides special APIs requiring substantial OS modifications in user applications

CPU Virtualization

- A CPU architecture is virtualizable if it supports the ability to run the VM's privileged and unprivileged instructions in the CPU's user mode while the VMM runs in supervisor mode.
- Hardware-Assisted CPU Virtualization: This technique attempts to simplify virtualization because full or paravirtualization is complicated

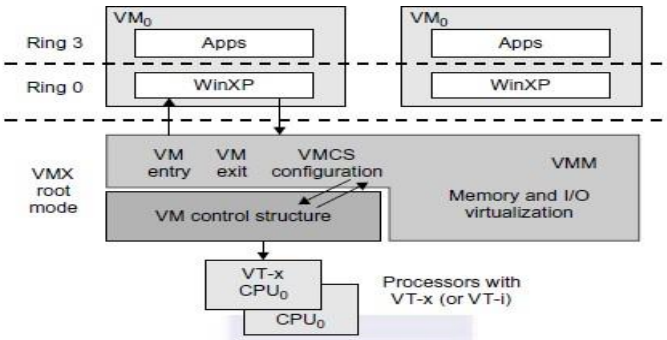

FIGURE 3.11

Intel hardware-assisted CPU virtualization.

(Modified from [68], Courtesy of Lizhong Chen, USC)

Memory Virtualization

- **Memory Virtualization** :the operating system maintains mappings of virtual memory to machine memory using page table
- All modern x86 CPUs include a memory management unit (MMU) and a translation lookaside buffer (TLB) to optimize virtual memory performance
- Two-stage mapping process should be maintained by the guest OS and the VMM, respectively: virtual memory to physical memory and physical memory to machine memory.
- The VMM is responsible for mapping the guest physical memory to the actual machine memory.

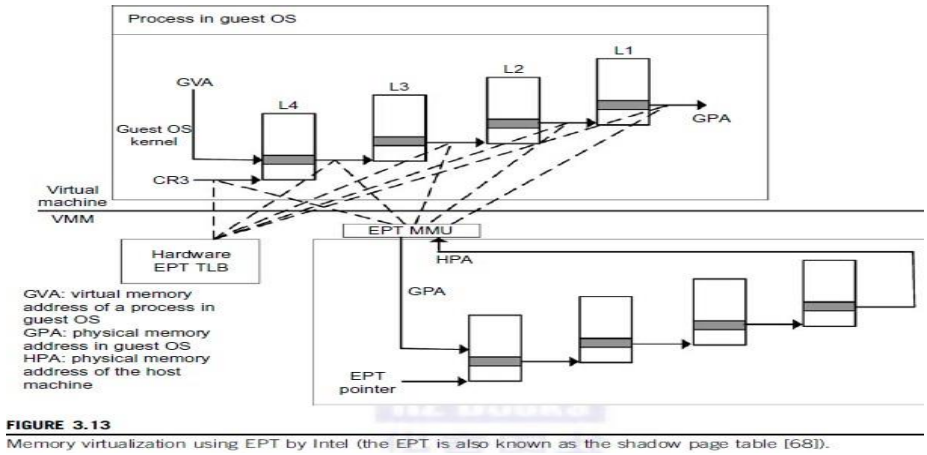

FIGURE 3.13
Memory virtualization using EPT by Intel (the EPT is also known as the shadow page table [68]).

I/O Virtualization

- I/O Virtualization managing the routing of I/O requests between virtual devices and the shared physical hardware
- managing the routing of I/O requests between virtual devices and the shared physical hardware
- Full device emulation emulates well-known, real-world devices All the functions of a device or bus infrastructure, such as device enumeration, identification, interrupts, and DMA, are replicated in software. This software is located in the VMM and acts as a virtual device
- Two-stage mapping process should be maintained by the guest OS and the VMM, respectively: virtual memory to physical memory and physical memory to machine memory.
- The VMM is responsible for mapping the guest physical memory to the actual machine memory.

Virtualization in Multi-Core Processors

- **Muti-core virtualization** has raised some new **challenges**
- **Two difficulties**: Application programs must be parallelized to use all cores fully, and software must explicitly
- Assign tasks to the cores, which is a very complex problem
- The **first challenge**, new programming models, languages, and libraries are needed to make parallel programming easier.
- The **second challenge** has spawned research involving scheduling algorithms and resource management policies
- **Dynamic heterogeneity** is emerging to mix the fat CPU core and thin GPU cores on the same chip

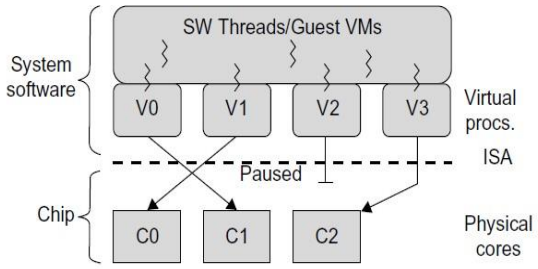

FIGURE 3.16

Multicore virtualization method that exposes four VCPUs to the software, when only three cores are actually present.

(*Courtesy of Wells, et al. [74]*)

- In **many-core chip multiprocessors (CMPs)**→
- Instead of supporting **time-sharing jobs** on one or a few cores, use the abundant cores →**space-sharing**, where single-threaded or multithreaded jobs are simultaneously assigned to separate groups of cores

Physical versus Virtual Clusters

- Virtual clusters are built with VMs installed at distributed servers from one or more physical clusters.
- Assign tasks to the cores, which is a very complex problem
- Fast deployment
- High-Performance Virtual Storage
- reduce duplicated blocks

Virtual Clusters

- **Four ways to manage a virtual cluster**.
- First, you can use a **guest-based manager**, by which the cluster manager resides on a guest system.
- The **host-based manager** supervises the guest systems and can restart the guest system on another physical machine
- Third way to manage a virtual cluster is to use an **independent cluster manager** on both the host and guest systems.
- Finally, use an **integrated cluster** on the guest and host systems.
- This means the manager must be designed to distinguish between virtualized resources and physical resources

Virtualization for data-center automation

- **Data-center automation** means that huge volumes of hardware,software, and database resources in these data centers can be allocated dynamically to millions of Internet users simultaneously, with guaranteed QoS and cost-effectiveness
- This **automation** process is triggered by the growth of virtualization products and cloud computing services.

- The latest virtualization development highlights high availability (HA), backup services, workload balancing, and further increases in client bases.

Server Consolidation in Data Centers

- heterogeneous workloads -chatty workloads and noninteractive workloads
- **Server consolidation** is an approach to improve the low utility ratio of hardware resources by reducing the number of physical servers

Virtual Storage Management

- **storage virtualization** has a different meaning in a system virtualization environment
- **system virtualization,** virtual storage includes the storage managed by VMMs
- and guest OSes data stored in this environment
- can be classified into two categories: VM images and application data.

Cloud OS for Virtualized Data Centers

- Data centers must be virtualized to serve as cloud providers
- **Eucalyptus for Virtual Networking of Private Cloud** :
- Eucalyptus is an **open source software system** intended mainly for supporting **Infrastructure as a Service (IaaS)** clouds
- The system primarily supports **virtual networking** and the management of **VMs**;
- **virtual storage is not supported.**
- Its purpose is to build **private clouds**
- three resource managers
    - Instance Manager
    - Group Manager
    - Cloud Manager