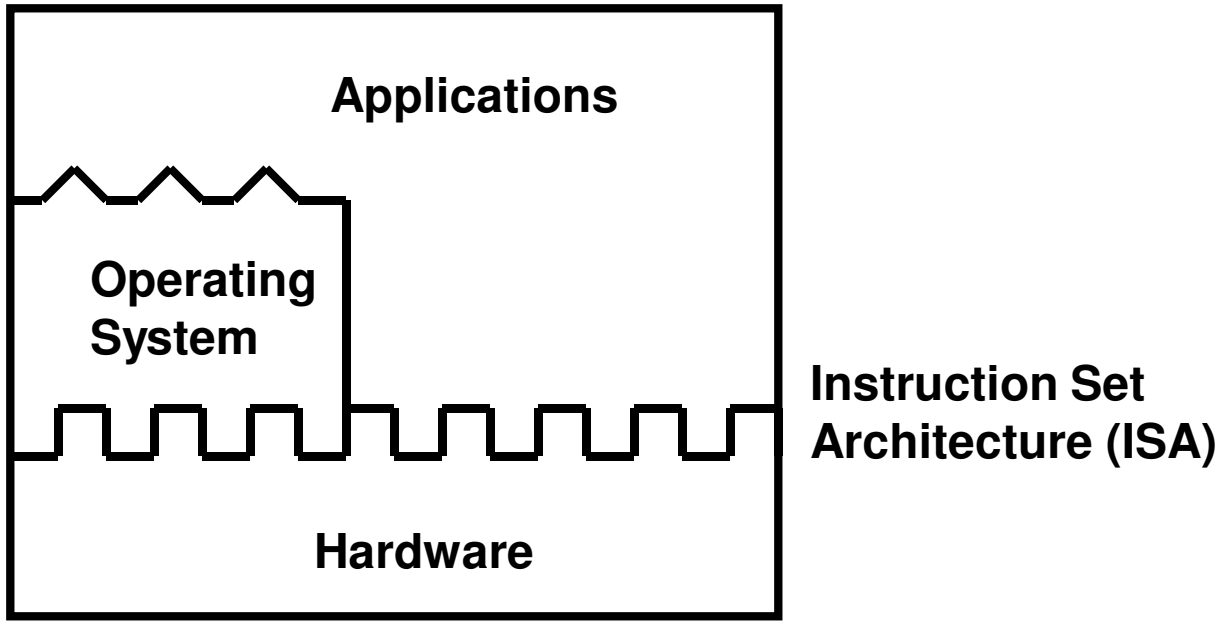
Virtualization

Dr. Natarajan Meghanathan
Associate Professor of Computer Science
Jackson State University
E-mail: natarajan.meghanathan@jsums.edu

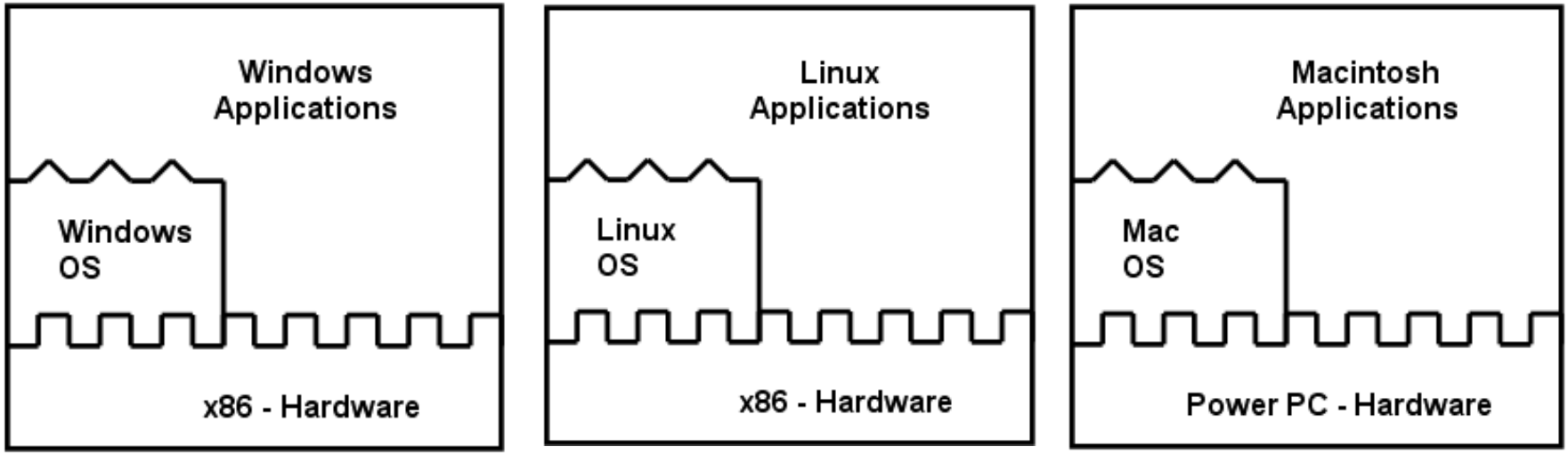
Conventional System Architecture

- The three major components (hardware, operating systems, application programs) are largely decoupled (from a development point of view). But, still they work together only in the proper combinations.
- Application software compiled for a particular ISA will not run on hardware that implements a different ISA. For e.g., Windows application binaries will not run on a Power PC (Mac) processor.

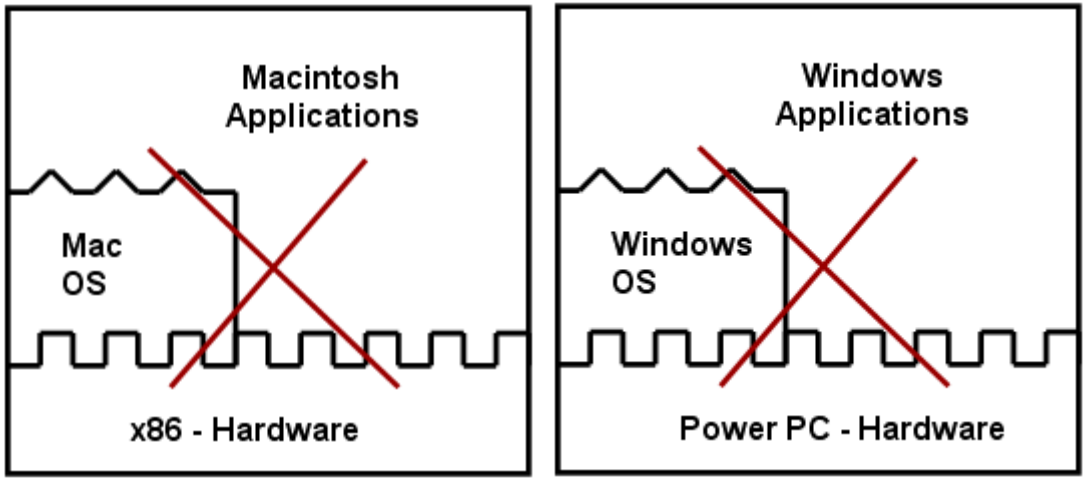


Conventional System Architecture

Working Combinations



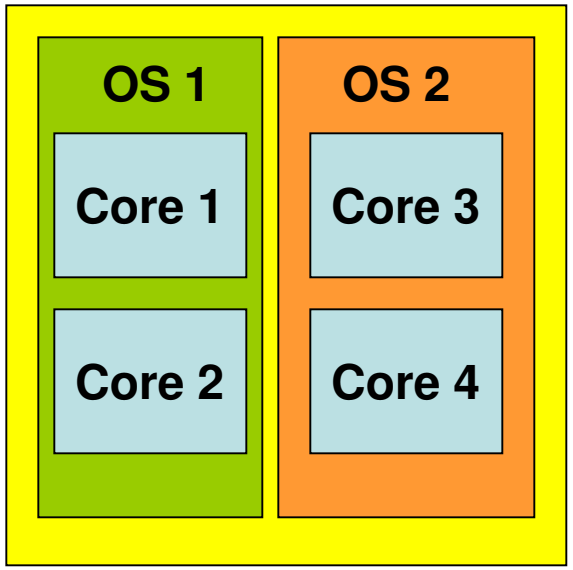
Combinations that are NOT Interoperable



<u>Motivation</u> – If a software is restricted to run on only certain nodes (OS/ ISAs) in the network, then a great deal of flexibility and interoperability is lost, especially with the significant growth of the Internet and the heterogeneity of the nodes that are part of it.

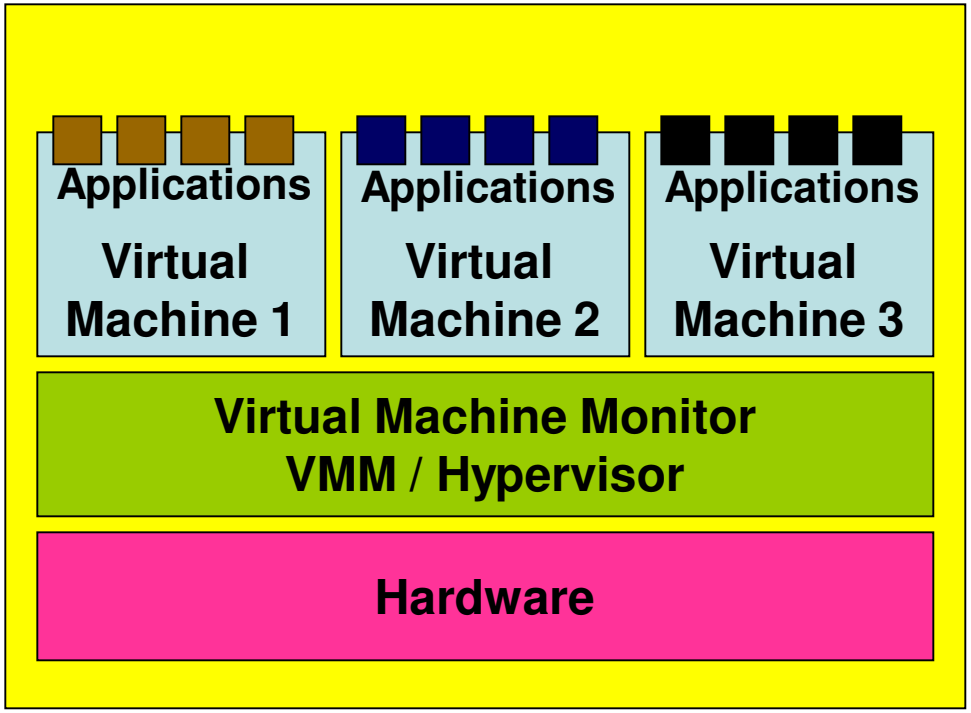
Virtualization Basics

- Virtualization is a technology that facilitates running more than one operating systems side-by-side just on the same processing hardware.
- Virtualization allows processing that would have been achieved on multiple computers to run on just one powerful multi-core processor.
- As multi-core processors with 4, 8 and 16 cores on a chip are becoming common, many processor cores are likely to be underutilized in a typical system.
- Most applications will have a finite amount of parallel tasks that can be executed at a given time, leaving many processor cores idle.
- Virtualization could be used to allocate groups of processor cores to individual operating systems running in parallel.



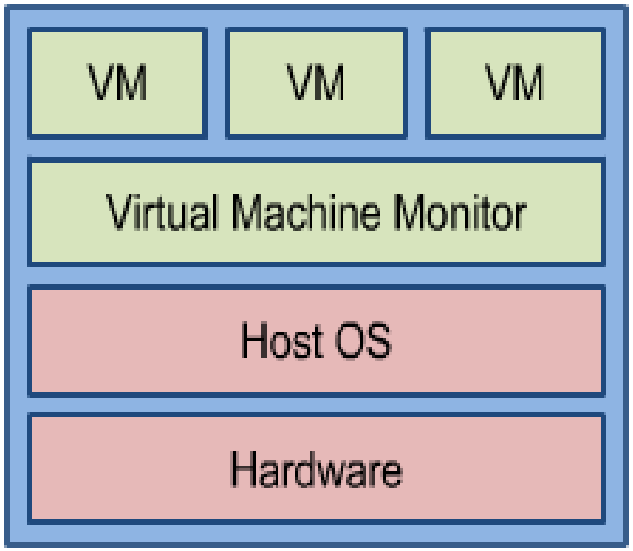
Virtualization Basics

- To virtualize a given computer, a piece of software called the Virtual Machine Monitor - VMM (also commonly referred to as a hypervisor) must be installed.
- VMM enables running multiple operating systems (OS) to run in parallel on the same hardware
- Each instance of an OS is called a virtual machine (VM).
- Each VM can run on its own operating system (called guest operating system), applications, etc.
- The objective of virtualization is to make each VM act like a standalone machine would.

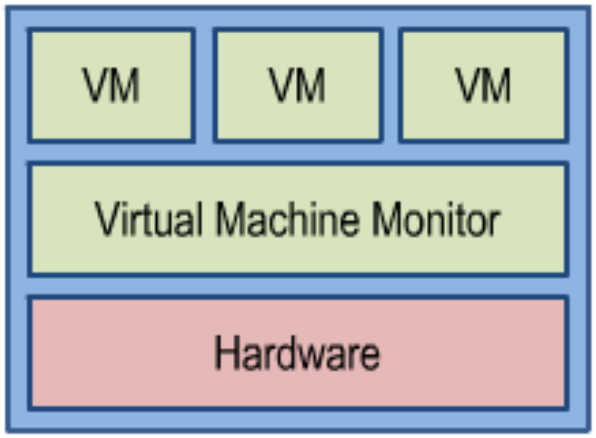


Virtualization Architectures

- There are two major virtualization architectures that can be used when installing virtualization software: hosted and bare-metal.
- Hosted: A VMM is installed on top of a host OS
- Bare-metal: A VMM is installed directly on the computer hardware for more low-level access and relying on host OS is not much desirable.
- Both architectures differ in installation, access to I/O devices for data acquisition and performance



Hosted Architecture



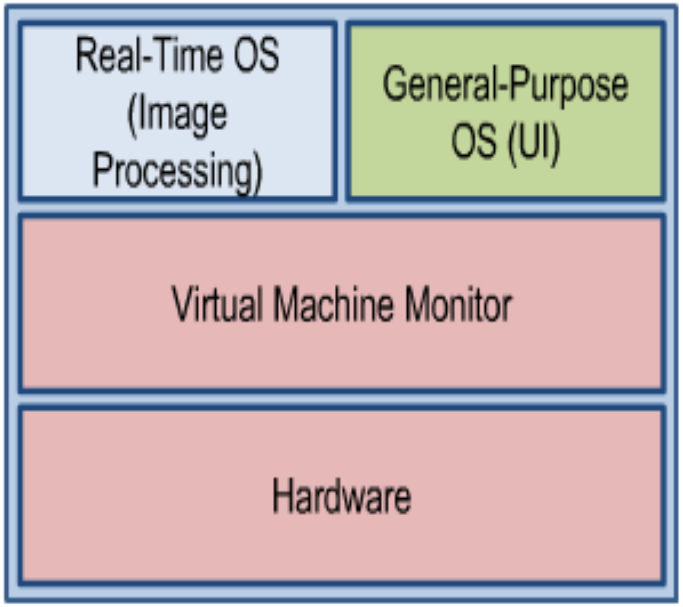
Bare-metal Architecture

Use Cases for Hosted Virtualization Architecture

- The hosted architecture is typically used during the development process.
- For example, a hosted VM architecture could be used for testing alpha and beta software as each individual VM is isolated from each other. If one VM (may be a VM running beta software) corrupts the operating system – it will not affect any other VM OS and the host OS.
- Other Use Cases
- The hosted architecture could be used to run applications written for legacy applications preventing the need of a dedicated computer to run an older software.
- The hosted architecture enables running applications written for several different operating systems on one computer. One VM may be used for running Windows applications and another VM for running Linux applications and so on.
- Example for Hosted Virtualization Architecture: VMWare Workstation

Use Case for Bare-metal Virtualization Architecture

- The low-level nature of bare-metal architectures makes them useful for deployed applications that use multiple operating systems.
- Imagine building a medical imaging device that needs to process medical data in real-time and also simultaneously provide an interactive GUI to users. Both the real-time OS and general purpose can be run simultaneously.
- The VMM (Hypervisor) for a Base-is rather “thin” (less code, lightly-loaded) compared to the Hypervisors of a Hosted architecture where they have to do a lot of work!!
- Bare-metal architectures are mainly preferred for real-time embedded applications where we need the I/O latency to be within deterministic limits.

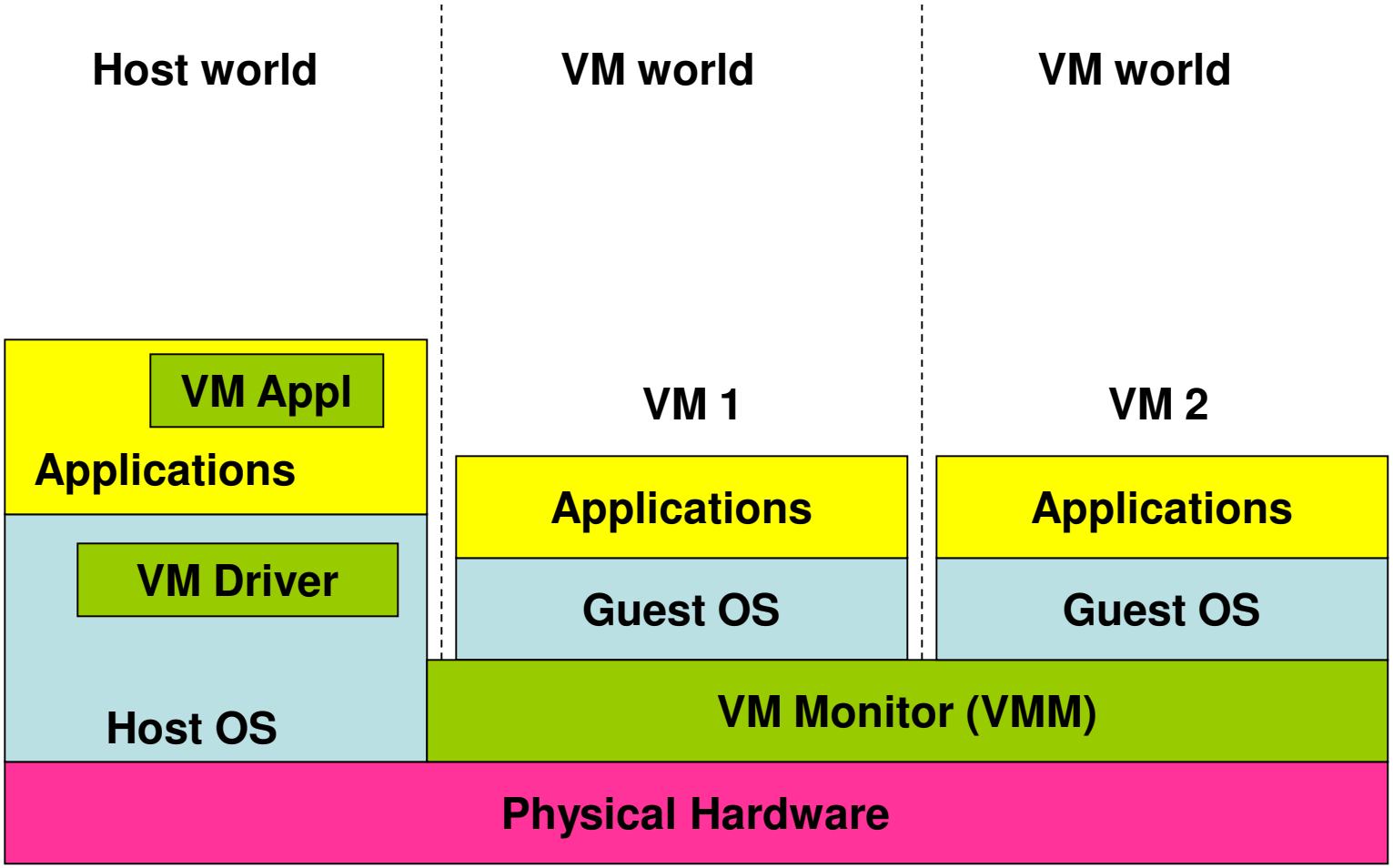


Examples of Hypervisors: VMWare ESX Server, WindRiver Hypervisor, LynuxWorks LynxSecure

Benefits of Virtualization

- Save hardware cost and footprint
  - No need of separate computer for multiple operating systems
  - Reduced system footprint (total resources exclusively allocated) for deployed applications
- Take advantage of the capabilities offered by different operating systems on just one set of hardware.
  - For example, one can use the graphics services provided by Windows in conjunction with the deterministic processing provided by a real-time OS.
- Make use of multi-core processors
  - More cores can be assigned to the different OS on a need-basis
- Test Beta software and maintain legacy applications
- Virtual machines are freely portable across different physical machines
  - A VM may have an OS, instruction set, or both, that differ from those implemented on the underlying real hardware.
- Increase system security
  - One can create multiple VMs at different security levels and operate them in parallel. A compromise or failure of one VM will not drastically affect the other VMs.
  - If needed (for e.g., in military applications), the VMM/ Hypervisor can isolate certain virtual machines and their data from being accessed by the peer virtual machines.

Hosted Virtualization – Detailed View



For example, the VM Application running on the host OS could be VMWorkstation

VM Workstation – Hosted Virtualization

- VMware Workstation installs like a normal application on the host operating system.
- The application portion (VMApp) of VM Workstation uses a driver (VMDriver) loaded into the host operating system to establish the privileged Virtual Machine Monitor (VMM) – running with the same privilege as the host OS – that runs directly on the hardware.
- From now on, the physical hardware is either in the host world or the VM world, with the VMDriver facilitating the transfer of control between the two worlds.
- Switching between the host and VM worlds involves saving and restoring all user and system visible state on the CPU, and is hence more resource consuming (heavy-weight) than a normal process switch.

Virtualization Techniques

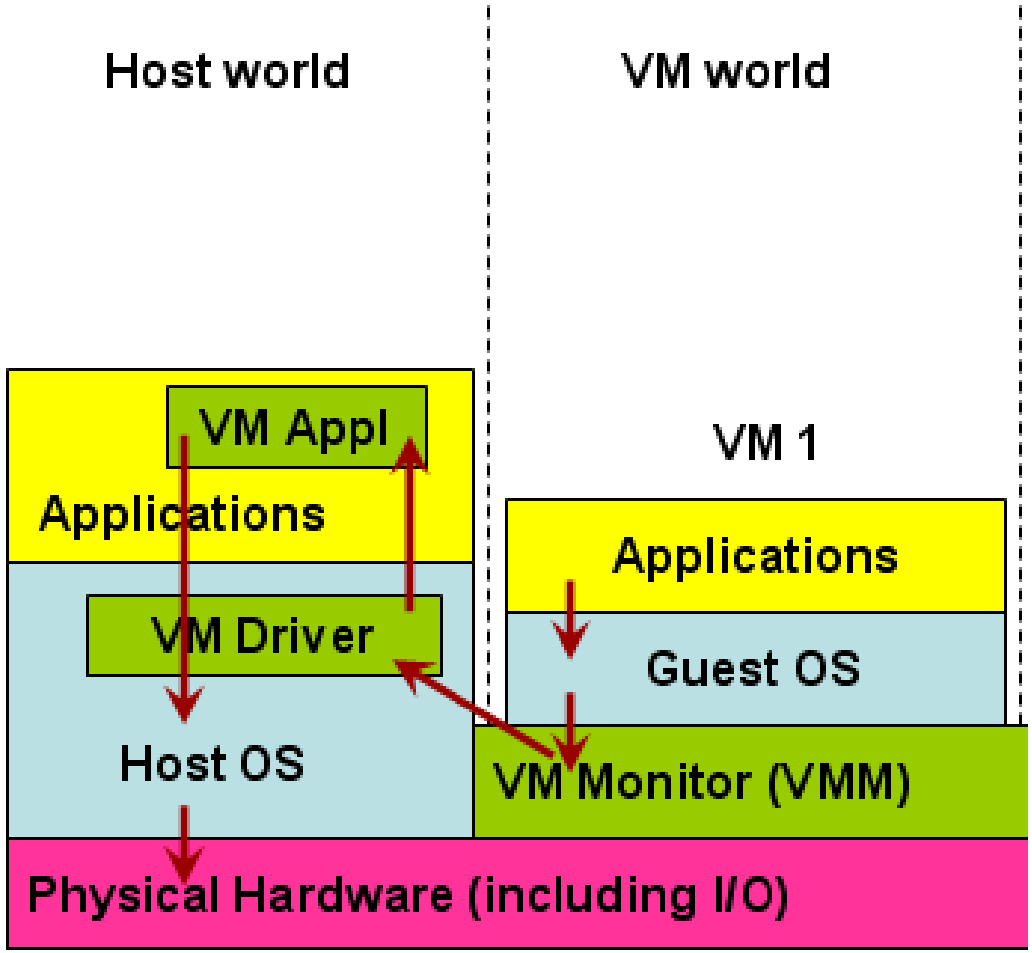
- The VMM for the hosted and bare-metal architectures may use one of the following techniques to isolate the individual virtual machines from computer hardware.
  - Binary Translation (Full Virtualization)
  - Para Virtualization
  - Hardware Assist
- All of the above techniques have the same end goal: to intercept any virtual machine instruction that could affect system state (shared resources) in any way. The techniques differ in the way they achieve this goal.
- The Binary Translation approach is the most common of the three techniques and Virtualization realized through this technique is referred to as Full Virtualization, a common approach for most of the well-known Hosted Virtualization architectures such as VM Workstation.

Example for Full Virtualization

- When a user application running on a virtual machine attempts to access the host hardware in non-privileged mode, the VMM gets out of the way and allows the virtual machine to execute directly on the hardware – providing CPU virtualization.
- Whenever the compiled code at a virtual machine contains a privileged instruction (e.g., accessing an I/O device) and the code is about to be executed, the underlying VMM traps the code, uses binary translation to dynamically alters the executing code (to avoid affecting the system state) and appropriately redirects the I/O request to the host OS to prevent conflicts between individual VMs.
- The VMM translates the instructions to match to the ISA of the host hardware and transfers control to the host world to the VMApp (running on the host OS) through the VMDriver.
- The VMApp will perform the I/O on behalf of the VM through appropriate system calls issued through the host OS.
- There is more work for the VMM (to translate the code) and the host OS (for switching between the VM world and the host world), leading to degradation in performance. To minimize the number of such switches and reduce the performance impact suffered by a user, VMMs typically inspects and translates groups of instructions at a time.

Full Virtualization and its Weakness

- The weakness of the binary translation approach of full virtualization is the potential degradation in I/O performance due to excessive CPU overhead. Because I/O emulation is done in the host world, a VM executing an I/O intensive workload will have to incur additional CPU time frequently switching between the VM and host worlds, as well as significant time in the host world performing I/O to the native hardware.



Control Flow for Processing I/O Request from an Application Running in the VM

Para Virtualization

- The Binary Translation approach keeps the guest OS out of the picture and overburdens the VMM and the VMApp running in the host world to do all the emulation of the underlying hardware for each of the VMs running on the host.
- With Para Virtualization, the code for each of the guest OS in a specific VM is modified statically (i.e., before running the applications) to coincide with the ISA of the underlying hardware and hence would be able to directly handle the privileged instructions of the applications running in the VM.
- The guest OS would be updated with the necessary drivers and the ISA to be able to directly access the underlying physical hardware.
- Since the guest OS implemented as part of each VM can do this virtualization in parallel, the technique is referred to as Parallel Virtualization – in short, widely called as Para Virtualization.
- The objective of Para Virtualization is to minimize the number of times the VMM is called and the switch between the host world and VM world occurs to execute the privileged instructions of the applications running in the VM.
- Limitations: Certain OS like Windows are not open source and cannot be modified.
- *Para Virtualization is mostly still in research stage.*
- The Para Virtualization is mainly focused on only open source OS.
- Example for Para Virtualization: Xen virtualization software

Hardware Assisted Virtualization

- Rather than requiring hardware emulation (binary translation) or operating system modifications (para virtualization), an alternative technique is to develop the hardware that is virtualization-aware and hence would directly allow multiple operating systems to simultaneous share the resources in a safe and efficient manner.
- Intel-VT and AMD-V are hardware technologies developed to be virtualization-aware.
- Hardware-assisted virtualization is mainly used with the bare-metal virtualization architecture; whereas, the binary translation-based full virtualization and para virtualization techniques are mainly used with the hosted virtualization architectures.

Networking Issues with VMs

- There are three popular methods to connect a VM to the host PC and the Internet.

- **Bridged:** Under the Bridged method, the VM will directly contact the DHCP server of the local network and apply for a unique local IP address in the LAN. The VM will be then able to directly access the Internet and all available resources on the LAN; other PCs and resources on the LAN can also directly access the VM using its IP address. This is the preferred connection method, if we run any server on the VM.

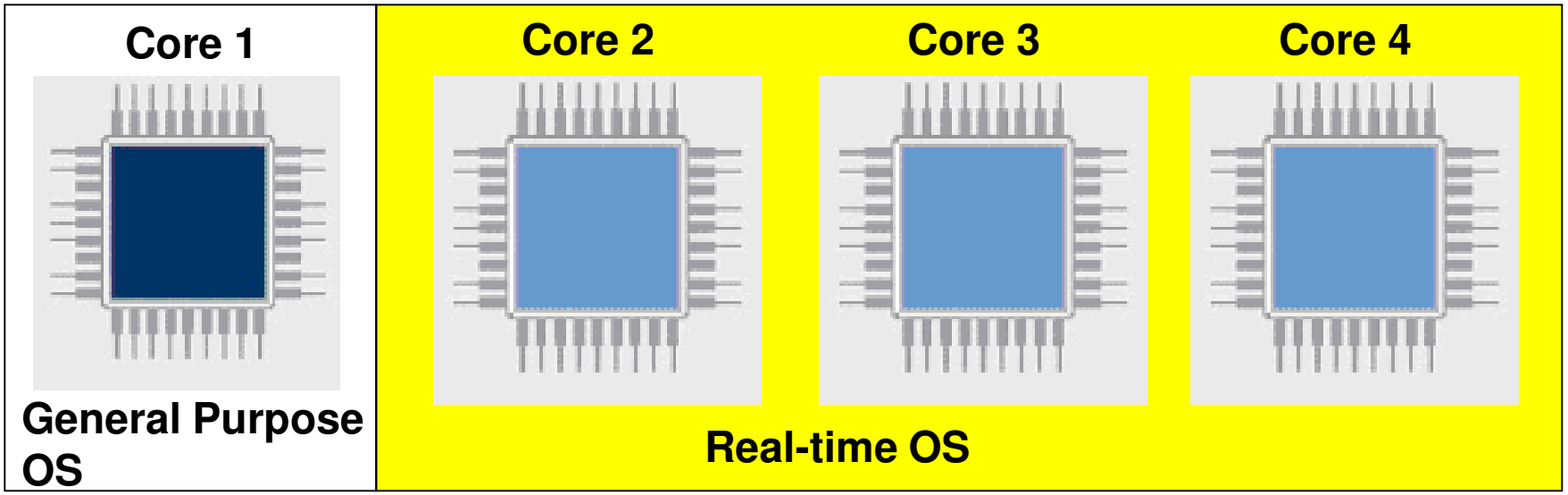
- **NAT (Network Address Translation):** Under this method, the VM accesses the Internet and the local LAN using the IP address of the host PC and internally, we have a virtual private network involving the host PC and the VMs running on it. The other PCs on the LAN cannot directly access the VM and they have to go through the NAT process at the host PC. In other words, the host PC acts as the first-stop gateway router for the VM.

- **Host-only:** This method is same as the NAT except the VMs cannot access the Internet. The VMs running on the same host PC and using the same host-only method can access each other's services.

Strategies for Performance Improvement of Virtualized Environment

- In multi-core machines, the VMM/ Hypervisor can schedule the different VMs to run on separate cores so that there is minimal intervention from the VMM to access hardware resources that are dedicated to each of the cores.
- Computation-intensive and real-time VMs could be assigned multiple cores where as more user-interactive general purpose OS could be assigned fewer cores.

Virtualization with Quad-Core Processor



Strategies for Performance Improvement of Virtualized Environment

- Partitioning of Resources across VMs: If we know that certain devices are to be exclusively used for specific OSs, then it is better to statically set access permissions to these resources at the VMM and all it has to worry about at run-time is to check whether the access request for a specific resource is coming from the appropriate VM.