NET – 332

NETWORK SECURITY LAB

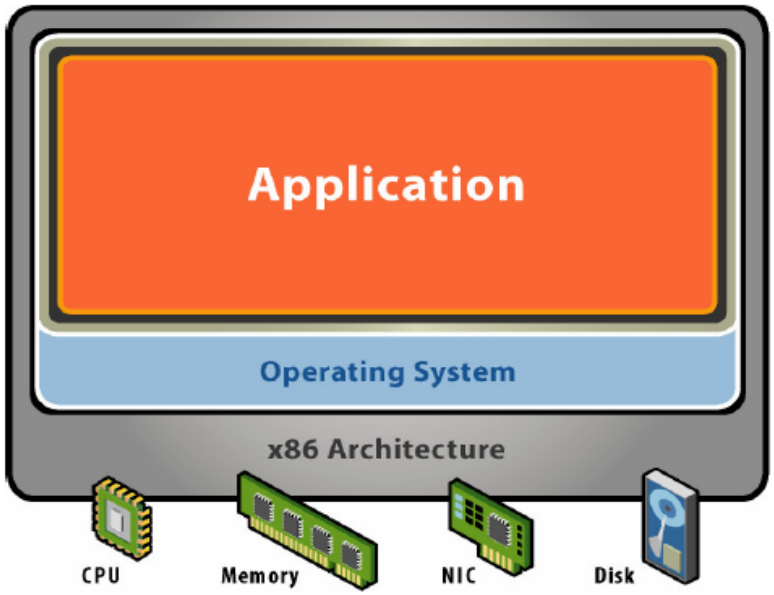
Lab 5. Virtual Machines

What is Virtualization?

- Virtualization deals with
  - "extending or replacing an existing interface so as to mimic the behavior of another system"
- Virtual system examples:
  - Virtual Private Network (VPN)
  - Virtual Machines

Virtualization Functional View

Starting Point: A Physical Machine



Physical Hardware

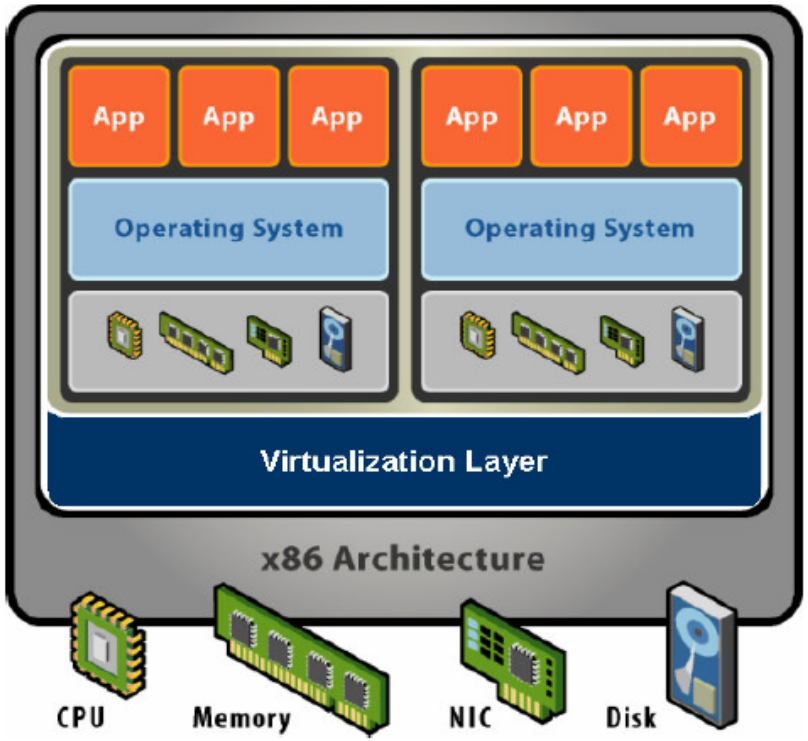
- Processors, memory, chipset, I/O bus and devices, etc.
- Physical resources often underutilized

Software

- Tightly coupled to hardware
- Single active OS image
- OS controls hardware

Virtual Machine

What is a Virtual Machine?



Hardware-Level Abstraction

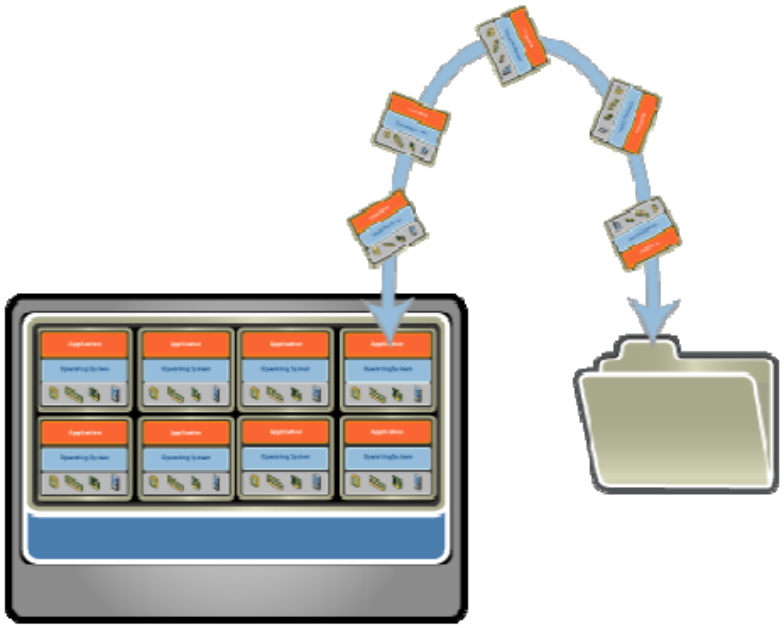
- Virtual hardware: processors, memory, chipset, I/O devices, etc.
- Encapsulates all OS and application state

Virtualization Software

- Extra level of indirection decouples hardware and OS
- Multiplexes physical hardware across multiple “guest” VMs
- Strong isolation between VMs
- Manages physical resources, improves utilization

Virtual Machine-Encapsulation

VM Encapsulation



Entire VM is a File

- OS, applications, data
- Memory and device state

Snapshots and Clones

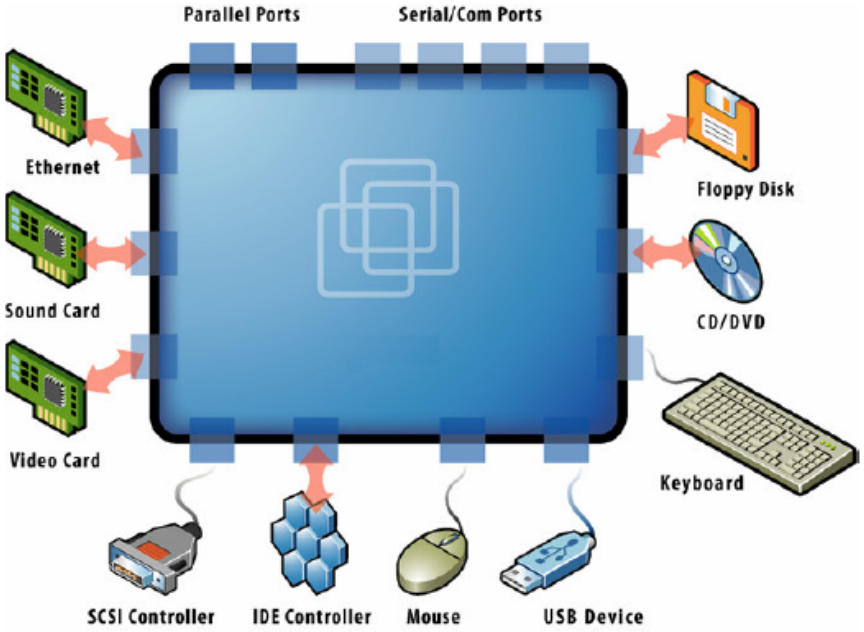
- Capture VM state on the fly and restore to point-in-time
- Rapid system provisioning, backup, remote mirroring

Easy Content Distribution

- Pre-configured apps, demos
- Virtual appliances

Virtual Machine-Compatibility

VM Compatibility



Hardware-Independent

- Physical hardware hidden by virtualization layer
- Standard virtual hardware exposed to VM

Create Once, Run Anywhere

- No configuration issues
- Migrate VMs between hosts

Legacy VMs

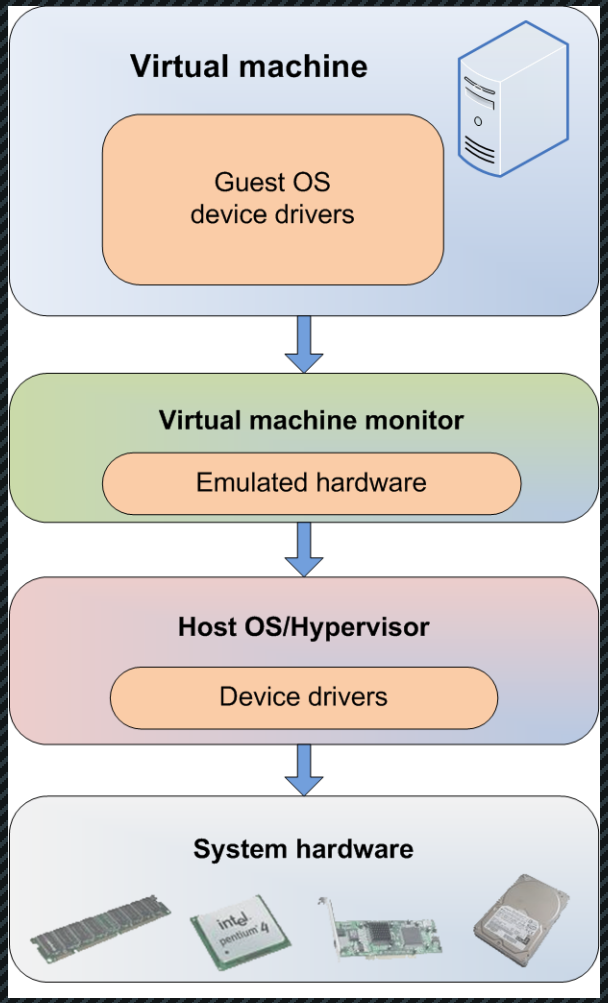
- Run ancient OS on new platform
- *E.g.* DOS VM drives virtual IDE and vLance devices, mapped to modern SAN and GigE hardware

Virtual Machine-Approaches

- Virtualization deals with
  - "extending or replacing an existing interface so as to mimic the behavior of another system"
- Virtual system examples:
  - Virtual Private Network (VPN)
  - Virtual Machines

Full Virtualization

- Everything is virtualized
- Full hardware emulation
- Emulation = latency

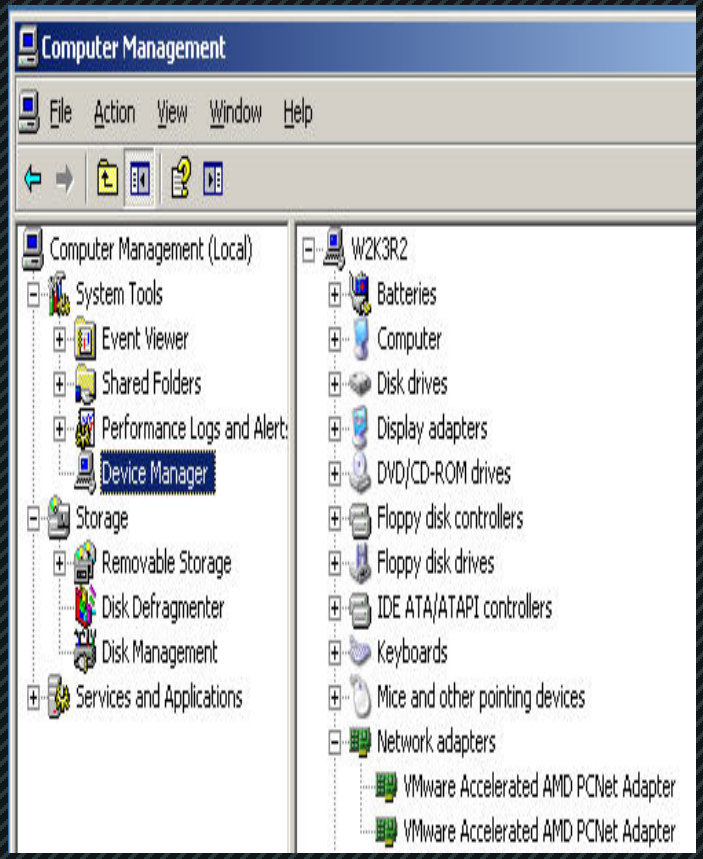


Pros and Cons

- Pros
  - Disaster recovery, failover
  - Virtual appliance deployment
  - Legacy code on non-legacy hardware
- Cons – LATENCY of core four resources
  - RAM performance reduced 25% to 75%
  - Disk I/O degraded from 5% to 20%
  - Network performance decreased up to 10%
  - CPU privileged instruction dings nearing 1% to 7%

Paravirtualization

- OS or system devices are virtualization aware
- Requirements:
  - OS level – recompiled kernel
  - Device level – paravirtualized or “enlightened” device drivers

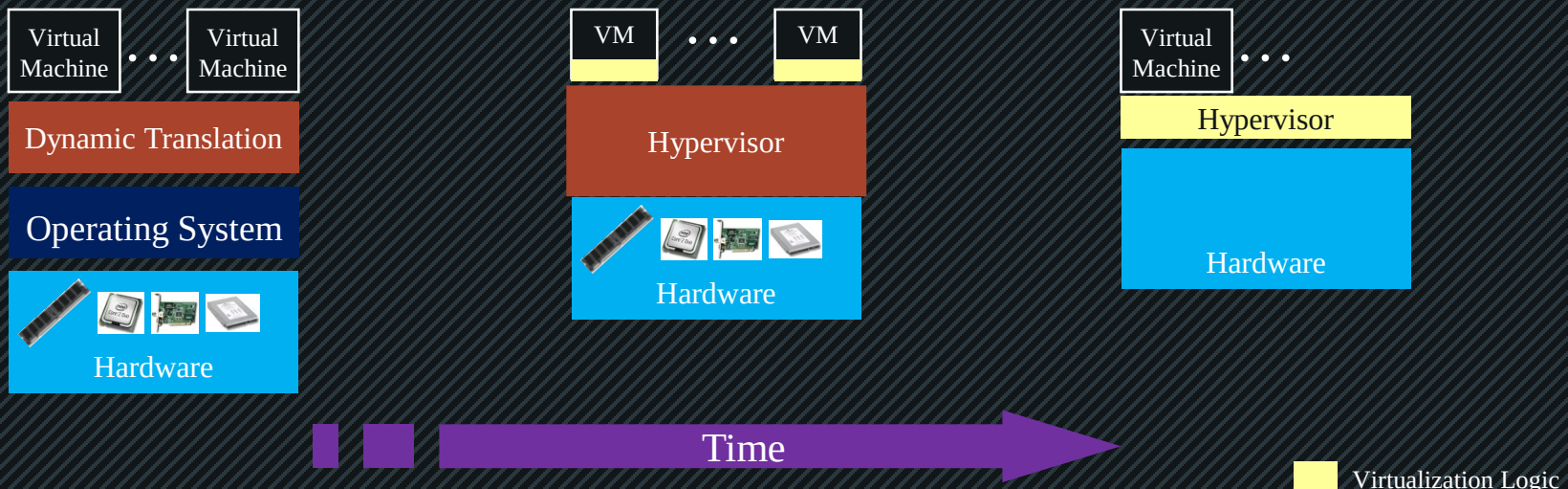


Pros and Cons

- Pro
  - fast
- Cons – LATENCY of core four resources
  - requires a specially modified guest OS, thus precludes the ability to run off-the-shelf and legacy OS in paravirtual environments

Evolution of Software solutions*

- 1st Generation: Full virtualization (Binary rewriting)
  - Software Based
  - VMware and Microsoft
- 2nd Generation: Paravirtualization
  - Cooperative virtualization
  - Modified guest
  - VMware, Xen
- 3rd Generation: Silicon-based (Hardware-assisted) virtualization
  - Unmodified guest
  - VMware and Xen on virtualization-aware hardware platforms



Hands-On

- Instructions can be found in the student lab manual for this topic.