Virtual Machine Systems
Question

- Can a “small” operating system simulate the hardware of some machine so that
  - Another operating system can run in that simulated hardware?
  - More than one instance of that operating system run on the same hardware at the same time?
  - More than one different operating system can share the same hardware at the same time?

- **Answer:** Yes
Solution – Virtual Machine

- A virtual machine provides interface identical to underlying bare hardware
  - i.e., all devices, storages, memory, page tables, etc.

- Virtual Machine Operating System creates **illusion** of multiple processors
  - Each VM executes independently
  - No sharing, except via network protocols
IBM Cambridge Scientific Center

Ran on IBM 360/67

- Alternative to TSS/360, which never sold very well

Replicated hardware in each “process”

- Virtual 360/67 processor
- Virtual disk(s), virtual console, printer, card reader, etc.

Cambridge Monitor System (CMS)

- A single user, interactive operating system

Commercialized as VM370 in mid-1970s
History (cont.)

- Various other attempts with other machines
- VMware
  - Workstation
  - Servers (for IT centers)
“Classic” Virtual Machine

- Copy of a real machine
  - “Any program run under the VM has an effect identical with that demonstrated if the program had been run in the original machine directly” ¹
- Isolated from other virtual machines
  - “…transforms the single machine interface into the illusion of many” ²
- Efficient
  - “A statistically dominant subset of the virtual processor’s instructions is executed directly by the real processor” ²
- Also known as a “system VM”

Classic Virtual Machines

- Virtualization of instruction sets (ISAs)
  - Language-independent, binary-compatible (not JVM)
  - 70's (IBM 360/370..) – 00’s (VMware, Microsoft Virtual Server/PC, z/VM, Xen, Power Hypervisor, Intel Vanderpool, AMD Pacifica ...)
- ISA+ OS + libraries + software = execution environment
Definitions

- **Host Operating System:**
  - The operating system actually running on the hardware
  - Together with virtualization layer, it simulates environment for ...

- **Guest Operating System:**
  - The operating system running in the simulated environment
  - To do some thing or resource allocation
Process vs. System VMs

- In Smith and Nair’s “The architecture of Virtual machines”, Computer, May 2005

Figure 3. Process and system VMs. (a) In a process VM, virtualizing software translates a set of OS and user-level instructions composing one platform to those of another. (b) In a system VM, virtualizing software translates the ISA used by one hardware platform to that of another.
Must Virtual Machine be Replica of Host Machine?

- No, virtualization layer can simulate any architecture
  - Typically used for debugging specialized systems
  - Real-time systems, niche products, etc.

- Guest architecture does not even have to be real hardware!
Example – Page tables

- Suppose guest OS has its own page tables then virtualization layer must
  - Copy those tables to its own
  - Trap every reference or update to tables and simulate it
- During page fault
  - Virtualization layer must decide whether fault belongs to guest OS or self
    - If guest OS, must simulate a page fault
  - Likewise, virtualization layer must trap and simulate every privileged instruction in machine!
Virtual Machines (cont.)

- The resources of the physical computer are shared to create the virtual machines
  - CPU scheduling can create the appearance that each user has own processor
  - Spooling and a file system provide
    - virtual card readers, virtual line printers
  - Disk partitioned to provide virtual disks
  - A normal user time-sharing terminal serves as the virtual machine operator’s console

Spool: simultaneous peripheral operations on-line
Virtual Machines (cont.)

- Virtual-machine concept provides complete protection of system resources
  - Each **virtual machine** is **isolated** from all other virtual machines.
  - However, it does not directly share the resources.
    - Virtualization layer
- Virtual-machine system is a good vehicle for operating-systems research and development.
  - System development is done on the virtual machine does not disrupt normal operation.
  - Multiple concurrent developers can work at same time.
Virtual Machines (cont.)

(a) Nonvirtual machine

(b) virtual machine
Some hardware architectures or features are impossible to virtualize

- Certain registers or state not exposed
- Unusual devices and device control
- Clocks, time, and real-time behavior
On-demand computing

- Embodiments:
  - Data-centers
  - Grid-computing
    - “coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations”
      - In the “The anatomy of the Grid,” Foster et. Al
    - Local control, decentralized management
    - Open general-purpose standards
    - Non-trivial QoS
Resource sharing

- Traditional computing/data center solutions:
  - Multitask/multiuser operating systems, user accounts, file systems ...
  - Always available **but static configurations**
    - Sharing possible if apps run on similar execution environments
  - Centralized administration
    - Tight control on security, availability, users, updates, etc

- Distributed Grid/data center requirements
  - Multiple administrative domains
    - Different policies and practices at each domain
    - Many environments possible
    - Dynamic availability
  - Must run all kinds of applications
    - **Application user** will neither trust unknown users sharing the same resource nor redevelop application to run in different environments
    - **Resource owner** will neither trust arbitrary users nor change environment for others’ applications
1 user, 1 app, several environments

Slide provided by M. Zhao
Many users, 1 app, many environments
Many users, apps and environments

Challenge: VM State Transfer

Dynamic, efficient transfer of large VM state is important

Slide provided by M. Zhao
Virtualization Technology for Grids

- Resource virtualization technology
  - Enables a resource to simultaneously appear as multiple resources with possibly different functionalities
    - Polymorphism, manifolding and multiplexing
- Virtual networks, data, applications, interfaces, peripherals, instruments …
Virtual networks

Logical links:

- multiple physical links, routing via native Internet routing
- tunneling, virtual routers, switches, ...
- partial to total isolation

Slide provided by M. Tsugawa
Virtualization Data/File

**Compute Server ‘C’**

- **NFS Client**

**NFS Server ‘S’**

- **NFSD**
- **Mountd**

Export /home to all uids on compute server C

mount S:/home

**Client**  Server**

**NFS Client**

mount S:/home/user_A

**Client**  **Server**

**GVFS Proxy**

**GVFS Proxy**

Export /home/user_A to shadow1 on C

**GVFS**

**GVFS**
Grid Virtual File System (GVFS)

- Originally named PVFS, is a virtualized distributed file system
  - providing high-performance data access in grid environments and seamless integration with unmodified applications.

- It leverages existing NFS (Network File System) support in operating systems, and uses user-level proxies to authenticate and forward RPC (Remote Procedure Call) requests between the native NFS client and server, and map user identities between different domains.
A Grid-building Recipe

1. Virtualize to fit needed environments
2. Use services to generate “virtuals”
3. Aggregate and manage “virtuals”
4. Repeat 1 2 3 as needed

• Net result:
  • Users interact with virtual entities provided by services
  • Middleware interacts with physical resources
Architectural Components of VM Service

VM Creation Request from Client

1. VM Request
2. Request Estimate
3. VM Creation Cost
4. Create VM
5. VM ID
6. VM ID

VMShop (VMArchitect
VMCreator, VMCollector, VMReporter)

vmnabb
vws010
VMPlant Daemon

brady
vws001
vmnair
vws002
manning
vws003
VMPlant Daemon

favre
vws005
VMPlant Daemon

Host OS (VMPlant)

Host OS (VMPlant)

Slide provided by Arijit Ganguly
Create VM Steps

1. Clone VM
   - Instantiate a new container
   - Fast copying of a base VM image
     - Virtual disk
     - Suspended memory (if available)

2. Configure VM
   - Execute scripts/jobs inside container to tailor to a particular instance
   - Communication crossing container boundaries to provide inputs/retrieve outputs

3. Destroy VM
   - Terminate container, delete non-persistent state
User-level Extensions

- Client-side proxy disk caching
- Application-specific meta-data handling
- Encrypted file system channels and cross-domain authentication

[Zhao, Zhang, Figueiredo, HPDC’04]
VMware – Modern Virtual Machine System

- Founded 1998, Mendel Rosenblum et al.
  - Research at Stanford University

- VMware Workstation
  - Separates Host OS from virtualization layer
  - Host OS may be Windows, Linux, etc.
  - Wide variety of Guest operating systems
  - < $200

- http://www.vmware.com/
# VMware Architecture

<table>
<thead>
<tr>
<th>application</th>
<th>application</th>
<th>application</th>
<th>application</th>
</tr>
</thead>
<tbody>
<tr>
<td>guest operating system (free BSD)</td>
<td>guest operating system (Windows NT)</td>
<td>guest operating system (Windows XP)</td>
<td></td>
</tr>
<tr>
<td>virtual CPU</td>
<td>virtual CPU</td>
<td>virtual CPU</td>
<td>virtual CPU</td>
</tr>
<tr>
<td>virtual memory</td>
<td>virtual memory</td>
<td>virtual memory</td>
<td>virtual memory</td>
</tr>
<tr>
<td>virtual devices</td>
<td>virtual devices</td>
<td>virtual devices</td>
<td>virtual devices</td>
</tr>
</tbody>
</table>

**Virtualization Layer**

**Host Operating System** (Linux)

**Hardware**
- CPU
- Memory
- I/O Devices
VMware Server

- **Free version released in 2006**
  - Runs on any x86 server hardware and OS
  - Windows Server and Linux Host OS’s

- **Partition a physical server into multiple virtual server machines**
  - Target market – IT centers providing multiple services
  - Allows separate virtual servers to be separately configured for separate IT applications
  - Portability, replication, etc.
VMware Server ESX

- Total decoupling between hardware and applications
- High-end, high-performance IT applications
  - Oracle, SQL Server, Microsoft Exchange server, SAP, Siebel, Lotus Notes, BEA WebLogic, Apache
- Dynamically move running application to different hardware
  - Maintenance, hardware replacement
  - Provisioning new versions, etc.

VMware ESX is an enterprise-level computer virtualization product offered by VMware
The Java Virtual Machine

- Own idealized architecture
- Stylized machine language
  - *Byte codes*
- Readily available interpreter