**Hot Topics (CSE 422S)**

Ken Wong  
Washington University  
kenw@wustl.edu  
www.arl.wustl.edu/~kenw

---

**Topics**

- **Chip Multiprocessors**  
  » Increasing clock speed  
  » Want higher speed but at about same power as 1 CPU  
  » Multiple CPU/caches and memory attached to an interconnect

- **Software Isolated Processes (MS Singularity)**  
  » Use SW verification instead of HW protection  
  - Verify safe behavior (cannot construct or corrupt a memory ref.)  
  - Type safe and memory safe operations

- **OS Virtualization**  
  » Run multiple commodity OSes on the same hardware instance  
  - e.g., XP and Linux on the same x86 processor  
  » Want resource isolation and performance guarantees  
  » OSes sit on top of a **Virtual Machine Monitor**

---

**OS Virtualization**

- **Virtual Machine Monitor (VMM)**  
  » Provides the illusion of many virtual machines  
  » Enables server consolidation, application mobility, new distributed (Internet) services

<table>
<thead>
<tr>
<th>VMs (Guest OSes)</th>
<th>User Software</th>
<th>User Software</th>
<th>User Software</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Linux</td>
<td>BSD</td>
<td>Win XP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VMM</th>
<th>virtual x86 CPU</th>
<th>virtual phy mem</th>
<th>virtual network</th>
<th>virtual blockdev</th>
</tr>
</thead>
</table>

**Paravirtualization**

- **Def.** Provide a VM abstraction that is similar but not identical to the underlying hardware

  - **Want**  
    » No modifications to application binaries  
    » Support for full multi-application OSes  
    » High performance and strong resource isolation

  - **Examples**  
    » Xen, Denali, VMware  
    » Open Network Laboratory (ONL)  
      » Want to run different versions of TCP from different vendors
Xen x86 Interface (1)

- **Memory Management**
  - Most difficult part of paravirtualization
  - x86 doesn't have software-managed TLB
    - TLB misses serviced by processor walking the page table
  - x86 TLB doesn't have identifier tags
    - Address space switches require complete TLB flush
  - Top 64 MB is reserved for Xen and is not accessible to guest OSes
  - All page table and segment table updates are validated by Xen

Xen x86 Interface (2)

- **CPU**
  - Xen runs in privilege ring 0 (highest)
    - Guest OS runs in privilege ring 1
    - Applications run in privilege ring 3
  - Privileged instructions (e.g., install new page table) are validated and executed by Xen instead of Guest OS
  - Exception handling (e.g., memory faults, system traps)
    - Registered with Xen by each Guest OS
    - System calls handled by `fast handler` which doesn't go thru Xen
  - Interrupts
    - Replaced by lightweight event system
  - Time
    - Each Guest OS has a timer interface (real and virtual time)

- **Device I/O**
  - Data transferred using asynchronous I/O rings

References

- Whitaker, Shaw and Gribble, "Denali: Lightweight Virtual Machines for Distributed and Networked Applications"