

Virtualization 3

COMP 252 - Lecture 7

Antoni Pop

antoni.pop@manchester.ac.uk

Previous Lecture: System Virtualization

▶ **Types of system virtualization**

- ▶ Native (bare-metal) hypervisor virtualization (e.g., Oracle VM Server, VMware ESX)
- ▶ Hosted virtualization (e.g., VMware player, VirtualBox, QEMU)

▶ **Implementation techniques**

- ▶ Paravirtualization (e.g., Xen) – static approach
 - ▶ OS is aware of virtualization
 - ▶ OS cooperates with VMM over resources (e.g., page tables)
 - ▶ Do not try to access resources, call VMM interface explicitly
- ▶ Detect & fix interfaces – dynamic approach
 - ▶ Guarded resources (privilege): only VMM/hypervisor has access
 - ▶ Trap when guest OS tries to access resources
 - ▶ Hardware support or use dynamic binary translation

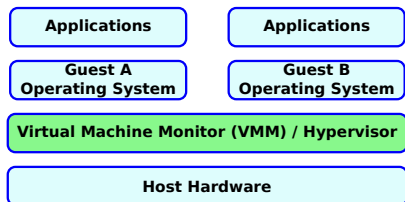
What can we do to a VM?

- ▶ To understand the VM-handling mechanisms of a hypervisor
- ▶ To understand how many different value-added services are constructed on top of VM-handling mechanisms

Starting a VM

Hypervisor

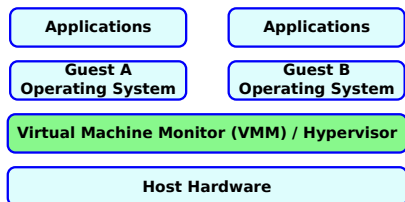
- ▶ gains control (e.g. clock tick)
- ▶ saves previous VM's CPU registers
- ▶ loads next VM's CPU registers
- ▶ jumps to next VM's next-PC (in correct privilege state)



Virtualized

Stopping a VM

- ▶ Save CPU registers into Hypervisor data area
- ▶ Hypervisor stops and starts VM all the time:
 - ▶ to share CPUs
 - ▶ to serialize access to resources
 - ▶ time multiplexing

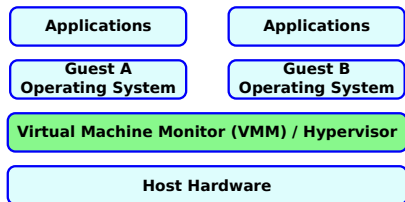


Virtualized

VM State While Stopped

VM State

- ▶ Memory (all guest physical memory)
 - ▶ Includes: Application state, OS state
- ▶ CPU state (registers)
- ▶ Small amount of I/O state
 - ▶ Let's stop VM when I/O is quiescent!



Virtualized

“Freeze” a VM

- ▶ Once suspended, the VM image is self-contained
 - ▶ VM can be (e.g.) copied to a file
 - ▶ (LARGE file!)

Applications

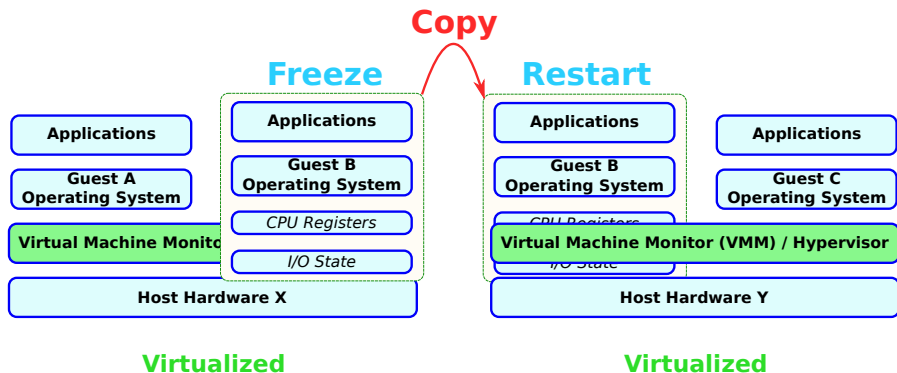
**Guest
Operating System**

CPU Registers

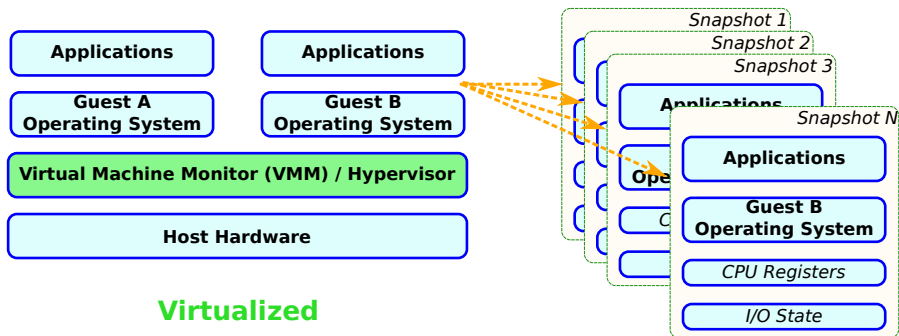
I/O State

What else can we do with this?

Move a VM



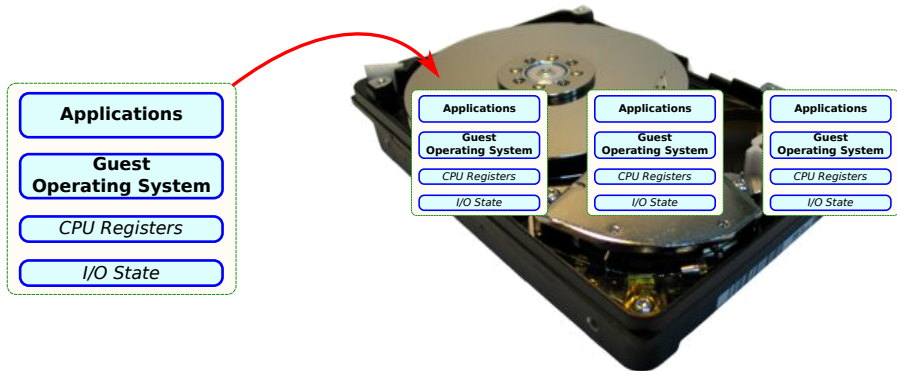
Snapshot and Rollback a VM



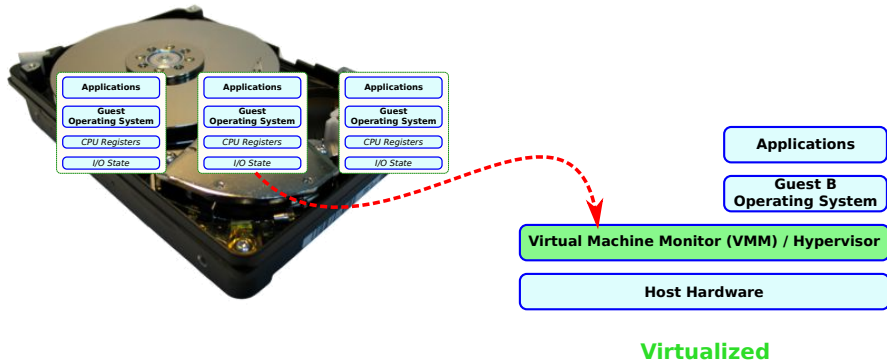
Why?

Can this process be optimized?

Archive a VM

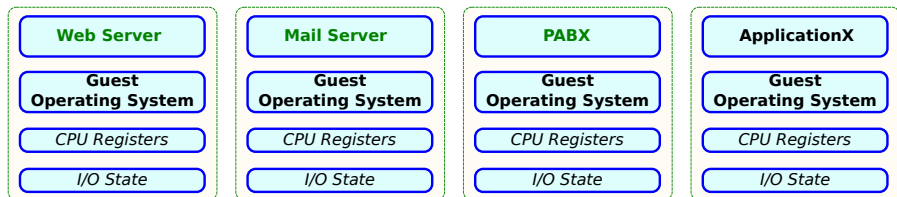


Rapid Provisioning



Virtual Appliances

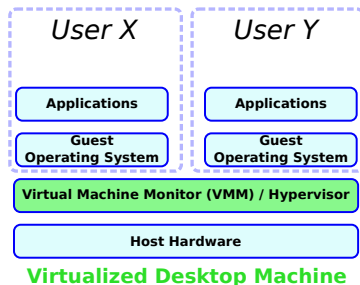
- ▶ <http://www.vmware.com/appliances/directory/>
- ▶ 1000+ downloadable appliances
- ▶ e.g., mail server, web server, hotel system, firewall, virus scanner, etc...



<https://www.aftermath-tracing.com//hipeac-2017-tutorial/Aftermath-HiPEAC17.ova>

Deploying Secure Desktops

- ▶ Increased security and flexibility
 - ▶ Better isolation between users
 - ▶ Users can have “admin” privileges within their Guest OS



Is this common? Where?

Optimizing live migration from source to destination VMM

- ▶ Copy every page from **source** to **destination** machine
 - ▶ reset *dirty* bit in VMM's page table for every page copied
- ▶ Repeat:
 - ▶ Find next *dirty* page in **source** machine
 - ▶ Copy to **destination** machine and reset *dirty* bit
- ▶ Until only minimal subset of pages left
- ▶ Suspend VM on **source**
- ▶ Copy remaining pages to **destination**
- ▶ Resume VM on **destination**

Load Balancing

- ▶ Management software monitors *load* on all physical machines
- ▶ If loads are mismatched, migrate a VM from a loaded to a less-loaded machine
- ▶ Independent of Application!
- ▶ Independent of Operating System!

High Availability

- ▶ For critical applications, keep a standby VM available on a different hardware system
- ▶ Regularly copy active VM image to standby VM (but don't activate it)
- ▶ Activate standby VM if active VM stops responding (VM crashes? VMM crashes? Hardware system fails?)

- ▶ Independent of Application!
- ▶ Independent of Operating System!

Goals of System Virtualization

- ▶ Multiple OS running on the same hardware
- ▶ Pre-configured virtual machines
- ▶ Load balancing
- ▶ High availability