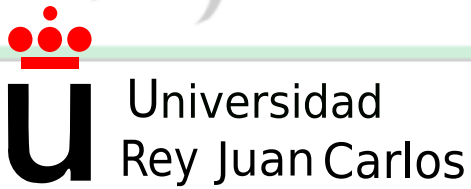
The Art of Virtualization with Free Software

Master on Free Software 2009/2010

Miguel Vidal, José Castro

{mvidal,jfcastro}@libresoft.es
GSyC/Libresoft – URJC

April 24th, 2010



Agenda

- Part 1: What is Virtualization
- Part 2: Types of Virtualization

What is Virtualization

What is Virtualization

Hardware/software combination, which allows a computer to act as several ones.

It includes making a single physical resource (such as a server, an operating system, or storage device) appears to function as multiple logical resources.

Definitions

- Virtualization is a methodology of dividing the resources of a computer into multiple execution environments.
- Virtualization applies one or more concepts or technologies such as partitioning, time-sharing, partial or complete machine simulation, emulation, quality of service, and many others.
- Colloquially, virtualization refers to the abstraction of computer resources.

Hypervisors

- In modern computing, Virtual Machine Monitors (aka hypervisors) allow many different OS, tasks and software configurations exist on the same physical machine.
- The hypervisor abstracts the physical resources of the host computer into discrete "virtual machines".
- Hypervisors ensure some level of isolation between guests.
- Hypervisors also provide a uniform interface to the hardware.

More on hypervisors

We have two hypervisor classes:

- Type 1 (or "native", "bare-metal"): the hypervisor is a layer between hardware and all OS. The "host OS" is known as Control Domain, and runs over the hypervisor.
- Type 2 (or "hosted") the hypervisor is a software layer running over the host OS.

History: Origins

- The term "virtualization" was coined in the 1960s, to refer to a virtual machine (sometimes called "pseudo machine").
- M44/44X Project (mid 1960s): the goal being to evaluate the then emerging time sharing system concepts.
- The architecture was based on virtual machines: the main machine was an IBM 7044 (M44) and each virtual machine was an experimental image of the main machine (44X).
- A number of IBM-based virtual machine systems were developed: the CP-40, the CP-67, the famous VM/370, and many more.
- A component called the virtual machine monitor (VMM) ran directly on "real" hardware.
- Multiple virtual machines could then be created via the VMM, and each instance could run its own operating system.

Virtualization Extensions for x86

- Since 2005, Intel and AMD have added additional hardware support to virtualization.
- Intel Virtualization Technology (VT) *codename* Vanderpool
- AMD Virtualization (AMD-V) *codename* Pacifica
- Added explicit funtionality to enable higher performance hypervisors for full virtualization.
- Full virtualization is easier to implement.
- Xen uses theses hardware extensions to support full virtualization.

Reasons for virtualization

- Isolate different applications and users on the same machine from interfering from each other.
- Server environment: consolidate many servers/services on a single machine (email, web, dns, etc.).
- Run operating system or other software built for one type of CPU on another kind of CPU.
- Easily and safely test software: debugging, developing, isolated –"sandboxed"– environments to study viruses, worms, etc.
- Easily deploy software using virtual software appliances. From business perspective, it provides a reduced total cost of ownership (TCO)
- Decrease power conssumption and cooling infraestructure in very dense datacenters.

Basic concepts

- The native Operating System running the virtualization software is called the Host.
  - The host has control of the real hardware.
- The virtualized OS is called a Guest.
  - There can be many Guests on a single Host.
  - Guests must not interfere with each other or the host.

Virtualization concepts

- The virtualization software is called:
  - Hypervisor.
  - Virtual Machine Manager or VMM.
- The VMM or Hypervisor is running as part of the host OS (or it's the host).
- A virtual hardware instance is called a Virtual Machine or VM.
- The Guests OS run inside a VM.

Virtualization and Cloud Computing

- Cloud Computing is not the same as virtualization management.
- But most Cloud Computing deployments depend on virtualization.
- Hardware management is highly abstracted from the buyer.
- Infrastructure capacity on CC is highly elastic (up or down).

Clouds are hardware-based resources converted into a "resource pool".

Types of Virtualization

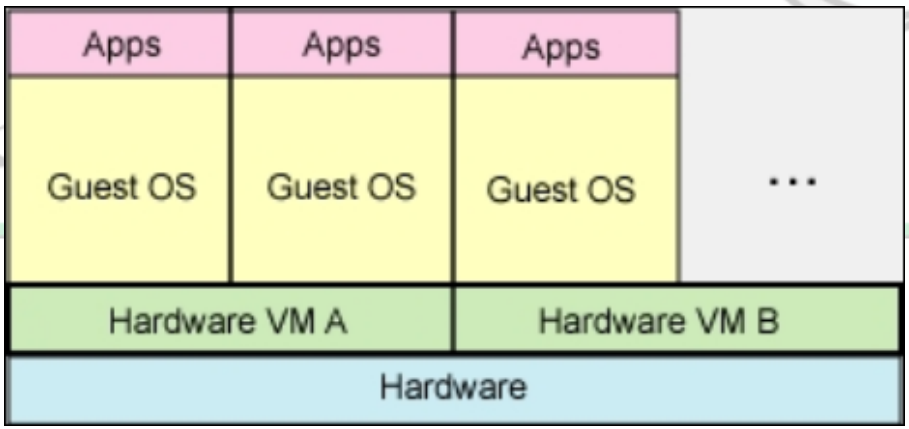
Types of Virtualization

Create the illusion of separate hardware and complete standalone systems. There are 4 major approaches in modern computing:

1. Emulation
2. Full virtualization
3. Paravirtualization
4. Operating system-level virtualization

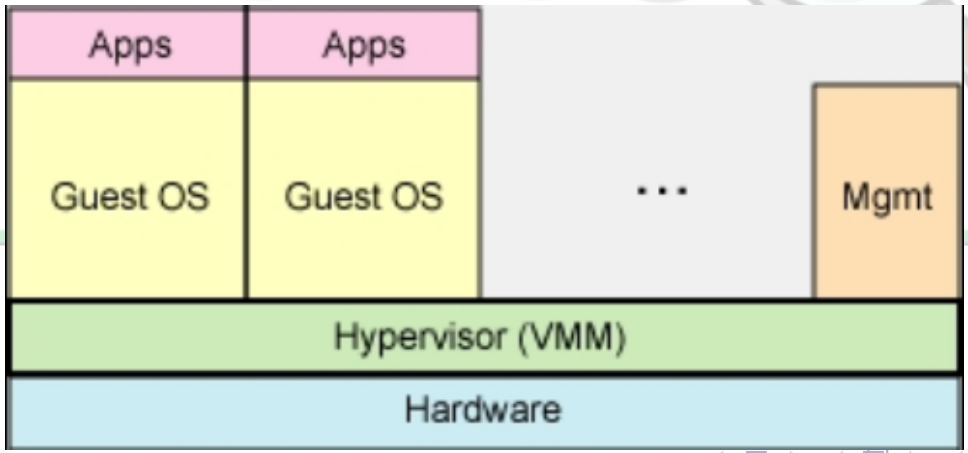
Emulation

- Virtual machine simulates the entire hardware.
- Unmodified guests in different hardware architecture runs inside a VM.
- Is used to create new OS or microcode for new hardware designs before HW is physically available.
- Advantages: Simulates hardware that is not phisically available
- Caveats: Low performance and low density (high cost)
- *Example:* basilisk II, a Mac (m68k) emulator for x86.



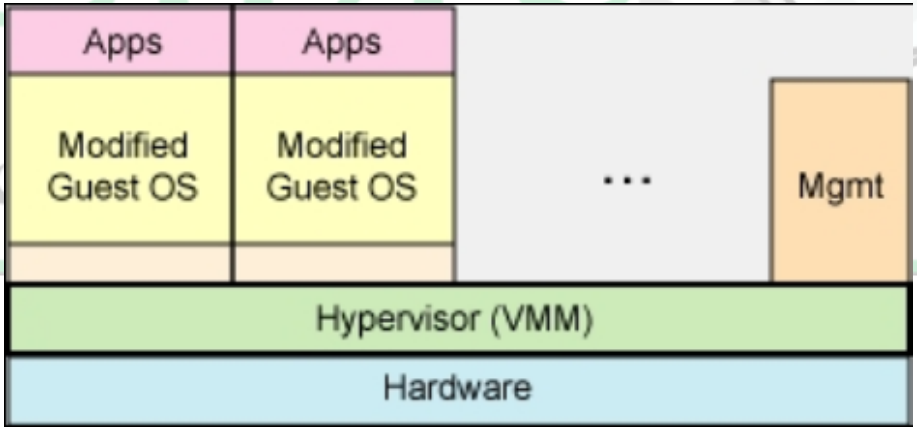
Full virtualization

- Similar to emulation, unmodified guests.
- Differs from emulation in that OSs are designed to run on the same architecture as the host.
- Hardware combined: CMT, Intel VT, AMD V, these CPUs control access to privileged instructions.
- Advantages: Flexibility, run different OSs from multiple vendors.
- Caveats: We can't emulate other archs.; Performance.
- *Examples:* VirtualBox, Xen + Intel VT.



Paravirtualization

- Hypervisor exports a modified version of the host.
- Exported VM is of the same architecture as the host.
- Advantages: Performance (lightweight and fast), scalability and manageability; strong isolation; allow virtualizing without need special CPU.
- Caveats: For same architecture. Requires porting Guest OS.
- Example: Xen with standard CPU.



Operating system-level virtualization

- Also called lightweight virtualization or *paenevirtualization* (i.e. "almost virtualization")
- Virtualization is done within a traditional single OS image (no hypervisor).
- OS is modified to allow various user space server processes in isolation from one another.
- Advantages: Fast, lightweight virtualization layer. Close to native performance. Density.
- Caveats: Strong isolation is difficult to implement.
- *Examples:* Linux VServers, FreeBSD jails, OpenSolaris Zones/Containers

Others types of Virtualization

- **Library virtualization:** Wine library (subset of the Win32 API to allow execute Windows applications)
- **Application virtualization** (managed runtime): virtual execution environment (standard API for cross-platform execution). *Example:* Java Virtual Machine.
- Advantages: Increases portability of apps.
- Caveats: Slower than native code, not capable of exucting a full operating systems

Desktop virtualization

- Decoupling of a users physical machine from the desktop and software he/she uses to work.
- It emulates the PC hardware environment of the client and run a virtual machine alongside the existing operating system located on the local machine or delivered to a thin client from a data server.
- Desktops as a service: transforming desktops into a cloud service.
- Different approaches.
- Citrix XenDesktop, Sun VDI, VMWare View.

Desktop virtualization: Advantages

- Instant provisioning of new desktops.
- Desktop computing power on demand.
- Significant reduction in the cost of new application deployment.
- Near-zero downtime in the event of hardware failures.
- Ability to access the users' enterprise desktop environment from any location.

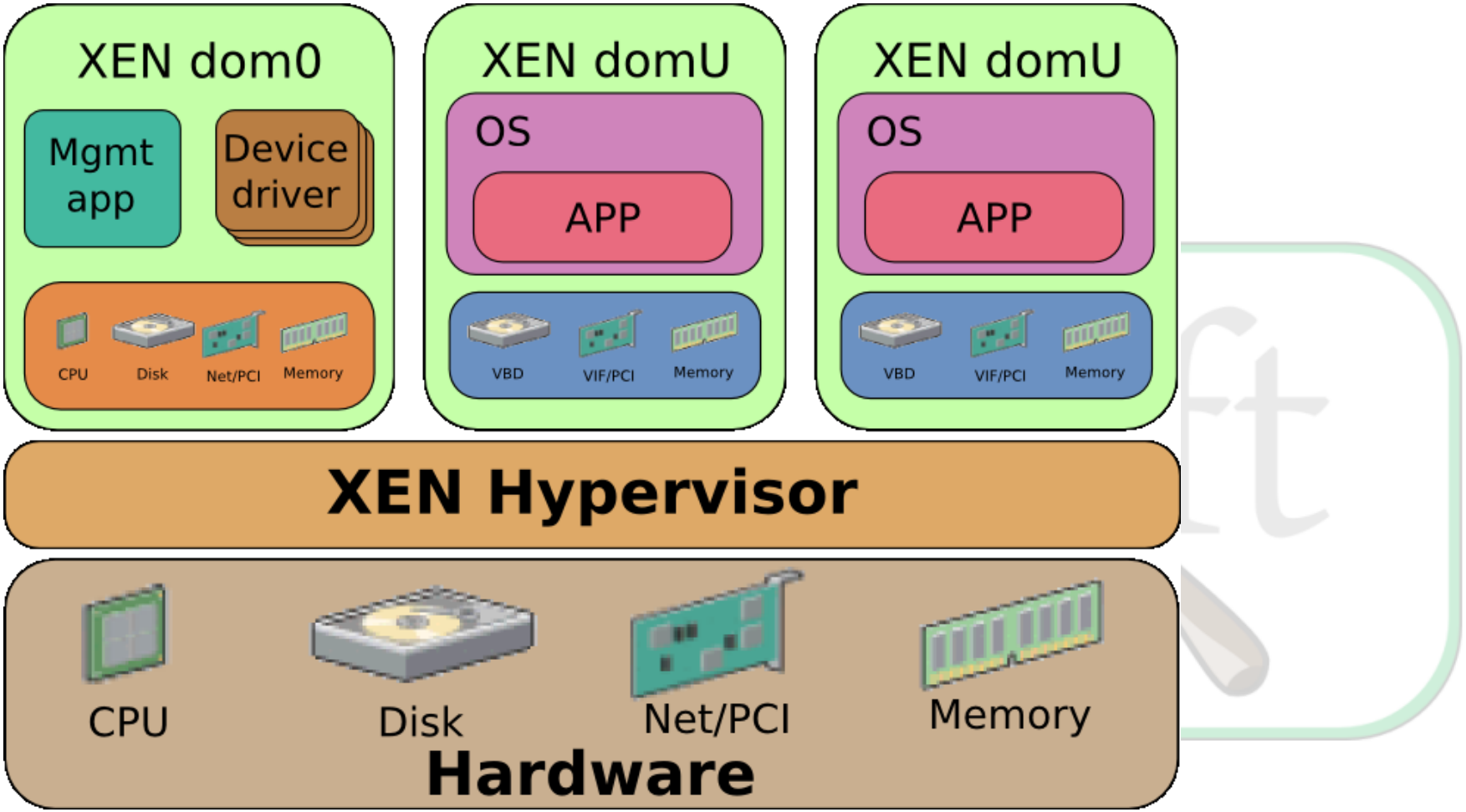
Xen: Features

- Xen uses paravirtualization, meaning that the guest operating system must be modified to use an hypervisor.
- Also allowing hardware-assisted virtualization (Intel VT or AMD-V), for unmodified guests: a significant development because it allows proprietary operating systems (such as Microsoft Windows) to be virtualized.
- Virtual machine migration.
- On x86, the Xen host kernel code runs in protection Ring 0, while the hosted domains run in Ring 1 or in Ring 3 (Rings: hierarchical protection domains)

Xen: Benefits

- Independence between virtualized systems. You can restart, and create independently.
- Better utilization of hardware of the machine: balancing resources. A virtual machine can get more resources if needed instead the other VM.
- Easy to backup, only with a copy of the virtual machine, it can be booted on a new server. Also, Xen allows hot migration, providing flexibility and little recovery time before an incident.
- You can modify parameters such as RAM, number of CPUs, disk space ... to solve the needs of the virtual machine.
- Test and Development Environments: multiple virtual machines on a single physical server for testing and development on a single machine simultaneously.

How does it work?



Kernel-based Virtual Machine (KVM)

- KVM is a Linux kernel virtualization infrastructure.
- Native virtualization using Intel VT or AMD-V.
- First version was included in Linux 2.6.20 (February 2007).
- A wide variety of guest operating systems work with KVM: Linux, BSD, Solaris, Windows, Haiku, ReactOS and a patched version of KVM is able to run Mac OS X.
- It does not perform any emulation: a user-space program (Qemu) uses the /dev/kvm interface to set up the guest VM's address space.

OpenSolaris xVM projects

- VirtualBox: VMs for all. Cross-platform.
- xVM server: Xen on Opensolaris 2009.06.
- Logical domains (LDOMs): Type 1 hypervisor, "full", for Sparc platform.
- Zones: Lightweight virtualization for OpenSolaris.
- Other: Sun xVM Ops center, Sun xVM VDI.

Containers and zones

- Virtualization which executes several instances of same operating system (and same kernel).
- Global Zone vs. local zones: we can't exit to the global filesystem (advanced chroot)
- Without hypervisor nor virtual hardware.
- Provisioning and administration very simple.
- x86: with "branding" support.
- From 2005 is part of Solaris 10 (and OpenSolaris).
- Similar to FreeBSD jails, OpenVZ (Linux containers) or Linux-VServer.

LDOMs virtualization

- "Full virtualization", Type 1 hypervisor based.
- It requires special CPUs (Chip Multithreading = CMT).
- Each domain is a full virtual machine with a reconfigurable subset of hardware resources.
- Operating systems running inside Logical Domains can be started, stopped, and rebooted independently.
- Base OS: Solaris 10 / Opensolaris 2009.06
- Virtual machine OS: Solaris 10, Opensolaris 2009.06, Sparc Linux, and other OS which support this architecture.
- But for now, only SunOS is supported.

Chip Multithreading (CMT)

- UltraSparc T1 / T2.
- Multithread. A thread is similar to a CPU.
- Example: T1 has 8 cores with 4 threads/core.
- Direct SSL support on hardware (1 MAU/core).
- LDOMs can assign cores to VMs.
- Hypervisor runs on server firmware.
- Free/Open Hardware: `http://www.opensparc.net`
- Sun Fire T / Enterprise T / Blade T servers.

References

- Amit Singh "An Introduction to Virtualization" (2004): `http://www.kernelthread.com/publications/virtualization/`
- Comparison of platform virtual machines: `http://en.wikipedia.org/wiki/Comparison_of_platform_virtual_machines`
- Jeanne Matthews *et al.* *Running Xen*. Prentice Hall, 2008 (Chapter 1: "Background and Virtualization Basics")