Virtualization

Originally prepared by Lehigh graduate Greg Bosch; last modified April 2016 by B. Davison

I. Introduction to Virtualization

II. Virtual Appliances

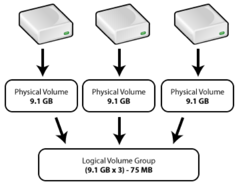
III. Benefits to Virtualization

IV. Example Virtualization Products

An Introduction to Virtualization

What is 'Virtualization'?

Technique for hiding the physical characterizes of computing resources from the way other systems, applications or end users interact with them



Two common functions:

- Making multiple physical resources appear to function as a single logical resource

We've see this before…

- Making a single physical resource appear to function as multiple logical resources

We'll talk about this today…

http://en.wikipedia.org/wiki/Virtualization

An Introduction to Virtualization

What is a 'Virtual Machine (VM)'?

Implementation of a machine that executes programs as if it were a real machine

Separated into two categories:

- **Process Virtual Machine**
  Runs as a normal application inside an operating system to abstract away the details of the underlying hardware

- **System Virtual Machine**
  Allows multiplexing (time sharing) of the underlying hardware between different operating systems

http://en.wikipedia.org/wiki/Virtual_machine

Virtual Machines

Process Virtual Machines

Designed to provide a platform-independent environment to a single process (i.e., program)

The environment is created when its associated process is started and destroyed when that process exits

Allows program to execute in the same way regardless of the physical platform it is running on

Implemented using an interpreter

The programmer's code is NOT compiled, but the interpreter requires compilation before providing the processing environment

http://en.wikipedia.org/wiki/Virtual_machine

Virtual Machines

System Virtual Machines

Designed to provide a complete platform which can support the execution of multiple, and different, operating systems

Allows for time-sharing of underlying hardware between virtual machines

Think of a scheduler that works on operating systems rather than processes

Operating Systems remain isolated from one another

The Instruction Set Architecture (ISA) provided by the virtual machine can be different from that of the real machine

http://en.wikipedia.org/wiki/Virtual_machine

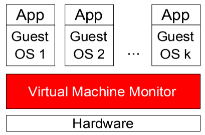
Virtual Machines

System Virtual Machines

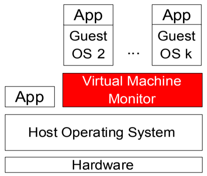
Implemented through the use of a **Virtual Machine Monitor (VMM)** also-known-as a Hypervisor

Two classifications of Hypervisors:

- **Native** (Hardware-Level): software runs directly on top of a given hardware platform as a control program for operating systems



- **Hosted** (OS-Level): software runs within an operating system environment as a control program for other operating systems



http://en.wikipedia.org/wiki/Virtualization

Virtual Machines

Native System Virtual Machines

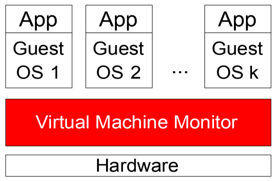
Is where virtualization began… In the time of the mainframe

IBM developed the first Native Hypervisor in the 1960s although 'hypervisor' hadn't entered the lexicon yet…

They called it CP/CMS and it consisted of two main components

- **Control Program** (CP)
  Which served to create the virtual machine environment for instances of

- **Cambridge Monitor System** (CMS)
  A lightweight single-user OS

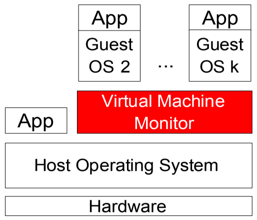


http://en.wikipedia.org/wiki/Hypervisor

Virtual Machines

Hosted System Virtual Machines

- Virtual Machine Monitor layer is moved one level higher as compared to Native VMs
- Runs within a Host operating system environment
- An operating system is installed first; as usual, on top of Hardware
- A Virtual Machine Monitor is then installed within the Host OS
- Guest operating systems can be installed on top of the VMM layer
- Host OS sees the VMM as a process
- VMM controls the allocation of time between Guest OSes
- Guest is segregated from the rest of the environment



http://en.wikipedia.org/wiki/Virtualization

Virtual Machines

Containers (OS virtualization)

Instead of virtualizing the hardware, run multiple virtual instances of same OS on single hardware

Advantage: Best Performance / Scalability, Ease of Admin.

Disadvantage: Only virtualizes copies of same OS

Single kernel means very low overhead (1..3%) compared to standalone server.

Containers provide isolation between processes, appear as separate OS.

Examples: Solaris Containers/Zones, FreeBSD Jails, OpenVZ, Linux-VServer

http://www.intel.com/technology/itj/2006/v10i3/2-io/4-virtualization-techniques.htm

Virtual Machines

Emulation or Simulation in Hosted System Virtual Machines

Virtual Machine provides a “guest” OS the (simulated) hardware environment it expects

Advantage: Guest software need not be modified

Disadvantage: Must pay performance penalty

Software is unaware that it is really talking to a virtualized device

Each interaction between Guest device driver with the emulated device hardware requires transaction with VMM

The real hardware does its job as usual, but the VMM must now translate the result for the guest

http://www.intel.com/technology/itj/2006/v10i3/2-io/4-virtualization-techniques.htm

Outline

I. Introduction to Virtualization

II. Virtual Appliances

III. Benefits to Virtualization

IV. Example Virtualization Products

Virtual Machines

Paravirtualization

Application Programming Interface (API) is provided to the Guest OS by the VMM so the guest may utilize the hardware

Advantage: Better Performance

Disadvantage: Guest OS must be modified to use API

Guest interacts with VMM at a higher level of abstraction

Instead of supplying the specifics of how to use the hardware, software provides general requests to the VMM

Decreases the number of interactions between Guest and VMM for a specific operation

http://www.intel.com/technology/itj/2006/v10i3/2-io/4-virtualization-techniques.htm

Outline

I. Introduction to Virtualization

II. Virtual Appliances

III. Benefits to Virtualization

IV. Example Virtualization Products

Benefits to Virtualization

Cost Benefit

Traditional production servers (physical box) run a single application

Utilization of hardware by that appliance estimated at 5-15%

Reduction of physical assets reduces substantial expenses
- Cost of hardware,
- Data center footprint,
- Electricity, and
- others

Virtualization allows for consolidation of appliances at ratios between 10:1 and 20:1 (perhaps closer to 2:1 for general purpose servers)

http://youtube.com/watch?v=MnNX13yBzAU&feature=related

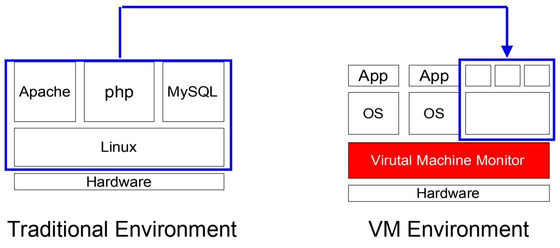
Virtual Appliance

Virtual Appliance is a virtual machine prepackaged with the necessary components to serve its intended purpose

Aimed to eliminate the installation, configuration and maintenance costs associated with complex stacks of software

For instance

LAMP Appliances (Linux + Apache + MySQL + PHP)



http://www.vmware.com/appliances/directory/465

Virtual Appliance

Example: Media Wiki

Software that runs wikipedia.org

Freely available to organizations

Packaged as a Virtual Appliance

Uses a minimal OS that installs within supported VMM

And includes all other necessary software packages

TRULY a Turn-Key system

The point of a virtual appliance: using your time to manage the application and not the system software

Lets take a quick look if we aren't behind schedule:

http://www.vmware.com/appliances/
http://bitnami.org/
http://www.turnkeylinux.org/all

http://www.vmware.com/appliances/directory/465

Benefits to Virtualization

Cost Benefit continued…

Man Hour Reduction too:

Instantiation of new virtual machine requires just minutes

As compared to the “old fashioned way” involving
- Sourcing of new hardware (purchasing and installation)
- Installation of Operating System, Patching
- Installation of relevant Applications
- Testing

Benefits to Virtualization

Operational Benefits

Automation of Installation process from OS to Applications
(i.e., Appliances)

Isolation (Sandboxing)
- Program Development and Testing
- Beneficial for running un-trusted Operating Systems or
- Un-trusted Applications

Provide legacy system support without allocating physical resources

Teaching environment for classes such as these

Check-pointing

The state of the machine can be saved, paused, restarted, even migrated to another machine

Outline

I. Introduction to Virtualization

II. Virtual Appliances

III. Benefits to Virtualization

IV. Example Virtualization Products

Xen

Originated as a University of Cambridge Project

Project led to founding of XenSource, Inc.

Acquired by Citrix in October 2007 for $500 Million

Products include:
Citrix XenServer (Free)
A free starter package for bringing virtualization to every server

Citrix XenDesktop
On-demand Windows desktop anywhere

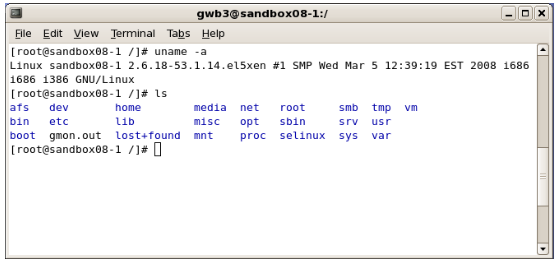
and more...

Xenproject.org open source Xen hypervisor (behind Amazon Web Services, RackSpace Public Cloud, Verizon Cloud, etc)

http://en.wikipedia.org/wiki/Xen

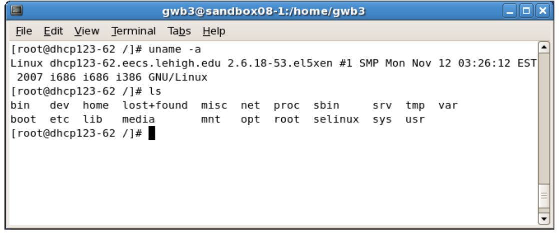
Xen

Sandbox Machine running Xen Linux Kernel



```
[root@sandbox08-1 /]# uname -a
Linux sandbox08-1 2.6.18-53.1.14.el5xen #1 SMP Wed Mar 5 12:39:19 EST 2008 i686
i686 i386 GNU/Linux
[root@sandbox08-1 /]# ls
afs   dev       home        media  net   root     smb  tmp  vm
bin   etc       lib         misc   opt   sbin     srv  usr
boot  gmon.out  lost+found  mnt    proc  selinux  sys  var
[root@sandbox08-1 /]# 
```

Virtualized CentOS environment within original Operating System



```
[root@dhcp123-62 /]# uname -a
Linux dhcp123-62.eecs.lehigh.edu 2.6.18-53.el5xen #1 SMP Mon Nov 12 03:26:12 EST
 2007 i686 i686 i386 GNU/Linux
[root@dhcp123-62 /]# ls
bin   dev  home  lost+found  misc  net  proc  sbin     srv  tmp  var
boot  etc  lib   media       mnt   opt  root  selinux  sys  usr
[root@dhcp123-62 /]# 
```

Xen hypervisor

Requires modification to Kernel of pre-installed OS

Modification installs Xen VMM just above the hardware (Native System Virtual Machine)

Xen boots from boot-loader (GRUB) and then loads the modified Host OS into the privileged domain (Dom0)

Administrator can use Host OS to install and then start guest OSs in the unprivileged domain (DomU)

Some commercial implementations (including Citrix, Oracle, etc.)

http://en.wikipedia.org/wiki/Xen

VMware

Proprietary virtualization software developer

x86-compatible architectures only

Both Desktop and Server space software packages:

Desktop:

VMware Workstation (orig. 1999) allowed users to run multiple x86 operating systems (free, but Pro version has more features)

VMware Fusion is the Mac-Intel platform product

Server:

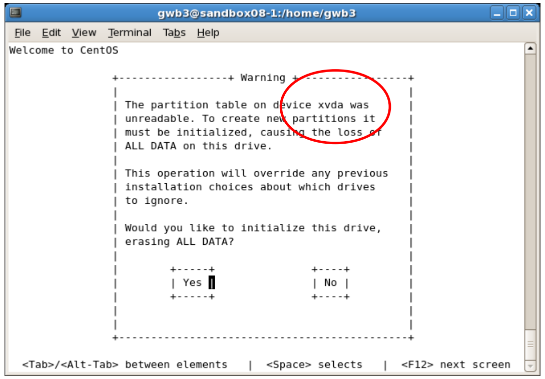
VMware vSphere/ESXi is an Enterprise marketed product

VMware Server is the less optimized, freeware version

http://en.wikipedia.org/wiki/vmware

Xen

Scary moment during installation of Guest OS



```
Welcome to CentOS

              +-----------------+ Warning +-----------------+
              |                                             |
              | The partition table on device xvda was      |
              | unreadable. To create new partitions it     |
              | must be initialized, causing the loss of    |
              | ALL DATA on this drive.                     |
              |                                             |
              | This operation will override any previous   |
              | installation choices about which drives     |
              | to ignore.                                  |
              |                                             |
              | Would you like to initialize this drive,    |
              | erasing ALL DATA?                           |
              |                                             |
              |        +-----+            +----+            |
              |        | Yes |            | No |            |
              |        +-----+            +----+            |
              |                                             |
              +---------------------------------------------+

  <Tab>/<Alt-Tab> between elements   |  <Space> selects  |  <F12> next screen
```

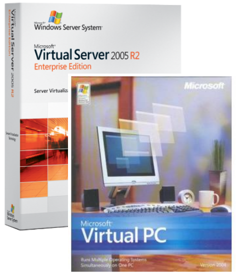
However, taking a closer look Xen is doing it’s job

Other Technologies

OpenVZ server virtualization

Parallels™

Xen™



KVM

VMware Fusion

Windows Server® 2008 Hyper-V™