VIRTUALIZATION TECHNIQUES

PALLAVI SHUKLA, APARNA DHOTE, PRAJAKTA WATKAR,RICHA PARATE

Pallavis667@gmail.com,aparnadhote17@gmail.com,prajaktawatkar0027@gmail.com,richa.parate@gmail.com

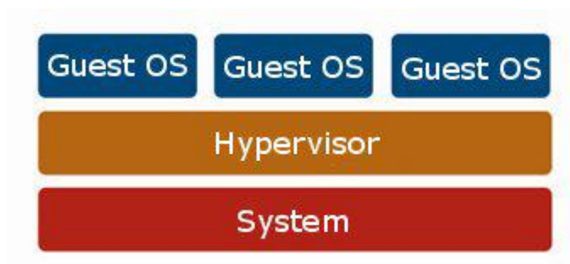
Abstract:Virtualization provides many benefits –greater efficiency in CPU utilization, greener IT with less power consumption, better management through central environment control, more availability, reduced project timelines by eliminating hardware procurement, improved disaster recovery capability, more central control of the desktop, and improved outsourcing services. With these benefits, it is no wonder that virtualization has had a meteoric rise to the 2008 Top 10 IT Projects! This white paper presents a brief look at virtualization, its benefits and weaknesses, and today's "best practices" regarding virtualization. The Association of Information Technology Professionals (AITP) recommends these "best practices" to obtain the benefits that virtualization offers

1 .Introduction:

Virtualization, in computing, refers to the act of creating a virtual (rather than actual) version of something, including but not limited to a virtual computer hardware platform, operating system (OS), storage device, or computer network resources.
Virtualization addresses IT's most pressing challenge: the infrastructure sprawl that compels IT departments to channel 70 percent of their budget into maintenance, leaving scant resources for business-building innovation.
The difficulty stems from the architecture of today's x86 servers: they're designed to run just one operating system and application at a time. As a result, even small data centers have to deploy many servers, each operating at just 5 to 15 percent of capacity—highly inefficient by any standard.
Virtualization software solves the problem by enabling several operating systems and applications to run on one physical server or "host." Each self-contained virtual machine (VM) is isolated from the others, and uses as much of the host's computing resources as it requires.

2.VIRTUALIZATION ARCHITECTURE

OS assumes complete control of the underlying hardware. Virtualization architecture provides this illusion through a hypervisor/VMM. Hypervisor/VMM is a software layer which Allows multiple Guest OS (Virtual Machines) to run simultaneously on a single physical host Provides hardware abstraction to the running Guest OSs and efficiently multiplexes underlying hardware resources.



3. WHY VIRTUAL MACHINE OVER PHYSICA L MACHINE?

A virtual machine (VM) is an operating system or application environment that is installed on software which imitates dedicated hardware. The end user has the same experience on a virtual machine as they would have on dedicated hardware. Specialized software called a Hypervisor emulates the PC client or server's CPU, memory, hard disk, network and other hardware resources completely, enabling virtual machines to share the resources. The hypervisor can emulate multiple virtual hardware platforms that are isolated from each other, allowing virtual machines to run Linux and Windows server operating systems on the same underlying physical host. Virtualization saves costs by reducing the need for physical hardware systems. Virtual machines more efficiently use hardware, which lowers the quantities of hardware and associated maintenance costs, and reduces power and cooling demand. They also ease management because virtual hardware does not fail. Administrators can take advantage of virtual environments to simplify backups, disaster recovery, new deployments and basic system administration tasks.Virtual machines do not require specialized hypervisor-specific hardware. Virtualization does however require more bandwidth, storage and processing capacity than a traditional server or desktop if the physical hardware is going to host multiple running virtual machines. VMs can easily move, be copied and reassigned between host servers to optimize hardware resource utilization. Because VMs on a physical host can consume unequal resource quantities (one may hog the available physical storage while another stores little), IT professionals must balance VMs with available resources. Physical machine has only suitable for Single OS but virtual machine is independent of os hardware. Application crashes

Affect the system but on virtual machine it isolates all apps physical machine uses the Resource under-utilization. virtual machine uses Safely multiplex resources across VMs.

4. HORIZONTAL & VERTICAL VIRTUALIZATION

鋍Another way of classifying virtualization techniques:
鋍Horizontal Virtualization:- It is virtualization across distributed back-end resources.
Software as a Service , Utility Computing and Grids are examples of this dimension.
鋍Vertical Virtualization:-It is virtualization across architectural layers.
Examples of this are Virtual Machines, Hypervisors, Virtual Appliances.

4.1 HORIZONTAL VIRTUALIZATION

a. Software as a Service (SaaS):
makes applications available in a remote data center through a service-based interfaces available to multiple external organizations.
Benefits: Reduced cost for software and infrastructure.
b. Utility Computing (Cloud):
means that resources that are managed by a single organization are made available to multiple external organizations as necessary (on demand).
Benefits: Reduced infrastructure cost.
c. Computational Grids:
Transparent sharing of computational server resources among multiple groups across or within an enterprise.
Benefits: Reduced cost of infrastructure.
d. Transactional Grids and Utilities:
Sharing distributed hardware and software platform resources to support high performance transactional applications.
Benefits : Reduced cost for transactional capabilities.
e. Data Grid and Utilities:
Transparent sharing of data servers among multiple applications across multiple groups or within an enterprise.
Benefits: Easier access to distributed data.
f. Storage Grids and Utilities:
Transparent sharing of distributed physical storage devices by multiple clients.
Benefits: Reduced infrastructure costs.

4.2 VERTICAL VIRTUALIZATION

a. Virtual Machine Monitor within Host OS:
Virtual machine capabilities built on top of a specific operating system. Can be used to partition resources or to host guest operating systems.
Benefits: Better utilization for resources.
b. Virtual Hypervisor on CPU:
Virtual machine capabilities built on top of a CPU not requiring a host operating system.
Benefits: Better utilization of resources.
c. Application Virtualization:
Platform (CPU, OS) independent and controlled environment for running applications.
Benefits: Portability.
d. Virtual Appliances:
Pre-configured bundling of application and operating system capabilities into a module that can run on a virtual machine. Provides a means of rapidly deploying applications using OVF standard.
Benefits: Ease of deployment & Managing evolving interdependencies across multiple appliances and physical enviornment.

5. ANOTHER TECHNIQUES

1.Binary Translation
2.Paravirtualization
3.Hardware Supported Virtualization

Binary Translation

鋍This approach translates kernel code to replace nonvirtualizable instructions with new sequences of instructions that have the intended effect on the virtual hardware

鋍Meanwhile, user level code is directly executed on the processor for high performance virtualization. Each virtual machine monitor provides each Virtual Machine with all the services of the physical system, including a virtual BIOS, virtual devices and virtualized memory management.

鋍This combination of binary translation and direct execution provides Full Virtualization as the guest OS is fully abstracted (completely decoupled) from the underlying hardware by the virtualization layer.

The guest OS is not aware it is being virtualized and requires no modification. Full virtualization is the only option that requires no hardware assist or operating system assist to virtualize sensitive and privileged instructions. The hypervisor translates all operating system Instructions on the fly and caches the results for future use, while user level instructions run unmodified at native speed.

<u>OS Assisted Virtualization or Paravirtualization</u>

鋍Paravirtualization refers to communication between the guest OS and the hypervisor to improve performance and efficiency.

鋍It involves modifying the OS kernel to replace nonvirtualizable instructions with hypercalls that communicate directly with the virtualization layer hypervisor.

鋍The hypervisor also provides hypercall interfaces for other critical kernel operations such as memory management, interrupt handling and time keeping.

鋍Paravirtualization is different from full virtualization , where the unmodified OS does not know it is virtualized and sensitive OS calls are trapped using binary translation.

鋍The value proposition of paravirtualization is in lower virtualization overhead

鋍While it is very difficult to build the more sophisticated binary translation support necessary for full virtualization, modifying the guest OS to enable paravirtualization is relatively easy

<u>Hardware Assisted Virtualization</u>

鋍In this section, we discuss recent architectural changes that permit classical virtualization of the x86.

鋍Hardware vendors are rapidly embracing virtualization and developing new features to simplify virtualization techniques. First generation enhancements include Intel Virtualization Technology (VTX) and AMD's AMD-V (SVM) which both target privileged instructions with a new CPU execution mode feature.

鋍Privileged and sensitive calls are set to automatically trap to the hypervisor, removing the need for either binary translation or paravirtualization. Processors with Intel VT and AMD-V became available in 2006, so only newer systems contain these hardware assist features.

鋍Extensions to x86-32 and x86-64.

鋍Allows classic trap-and-emulate.

鋍Hardware VM modes to reduce traps.

6. CONCLUSION

鋍Virtualization techniques will be present as a standard abstraction layer in the modern data center stack.

鋍Some of the concepts we have studied in the past should be updated:-

a. Computer architectures

b. Operating Systems

c. Networking

d. Security

e. Performance

鋍The final solution to virtualization seems to be a mix of the ones studied here:

鋍"Tomorrow's virtualization likely involves vendor-supported paravirtualized OSes that are installed into industry standard disk format files and able to run either natively or on a variety of compatible and interchangeable hypervisors that take advantage of hardware assisted management of the CPU, memory and I/O devices."

7. REFRENCES:

- Graziano, Charles. "A performance analysis of Xen and KVM hypervisors for hosting the Xen Worlds Project". Retrieved 2013-01-29.
- http://en.wikipedia.org/wiki/Virtualization
- . Braastad, "Management of high availability services using virtualization," Master's thesis, Oslo University College, May 2006.

- Kernel Based Virtual Machine, *http:/www.linux-kvm.org/, 2013*.