Distributed systems – Techs
4. Virtualization

Current interest in virtualization

- is one of the hottest topics in information technology today.
- Possible due to the increasing speed and capabilities of commodity hardware
- Provides an attractive way of making the most of the hardware
- The use of the term “virtualization” in today’s marketing literature rivals the glory days of terms such as “Internet” and “network - enabled” in the 1990s.
- Virtualization is commonly encountered in
  - networking, storage systems, and server processes,
  - at the OS level and at the machine level.
- Bad news:
  - successfully implementing, deploying, and supporting a new IT infrastructure based on virtualization

    requires the same level of planning and system design

    that any basic shift in infrastructure always has.

What is Virtualization?

- logical separation of the request for some service from the physical resources that actually provide that service
- provides the ability to run appls, OSs, or system services in a logically distinct system environment that is independent of a specific physical computer system.
- provides a level of logical abstraction that liberates applications, system services, and even the OS that supports them from being tied to a specific piece of hardware
- focus on logical operating environments rather than physical ones
- makes applications, services, and instances of an OS portable across different physical computer systems

…

- the core idea: provide logical access to physical resources!

Classical (known) example: virtual memory

- enables a computer system to appear to have more memory than is physically installed on that system.
- is a memory-management technique that enables an OS to see and use non contiguous segments of memory as a single, contiguous memory space
- is traditionally implemented in an OS by paging, which enables the Os to use a file or dedicated portion of some storage device to save pages of memory that are not actively in use.
- Known as a “paging file” or “swap space”, the system can quickly transfer pages of memory to and from this area as the OS or running appls require access to the contents of those pages.
- OSs such as Unix - like OSs (including Linux, the *BSD OS, and Mac OS X) and Microsoft Windows use some form of VM to enable the OS and appls to access more data than would fit into physical memory.

Virtualization levels

1. Application virtualization
2. Desktop virtualization
3. Network virtualization
4. Server or machine virtualization (!)
5. Storage virtualization
6. System-level or operating system virtualization

Application Virtualization

- The term describes the process of compiling applications into *machine -independent byte code* that can subsequently be executed on any system that provides the appropriate VM as an execution environment.
- Appls compiled into byte code become logical entities that can be executed on different physical systems with different characteristics, OSs, and even processor architectures.
- Examples of this approach:
  - The best know one: the byte code produced by the compilers for the Java programming language,
  - The concept was actually pioneered by the UCSD P - System in the late 1970s, for which the most popular compiler was the UCSD Pascal compiler.
  - Microsoft adopted a similar approach in the Common Language Runtime (CLR) used by .NET applications,
    - code written in languages that support the CLR are transformed, at compile time, into CIL (Common Intermediate Language), formerly known as MSIL (Microsoft Intermediate Language).
    - CIL provides a platform - independent instruction set that can be executed in any environment supporting the .NET Framework.

Desktop Virtualization

- The term describes the ability to display a graphical desktop from one computer system on another computer system or smart display device.
- Example:
  - Virtual Network Computing,
  - thin clients such as Microsoft's Remote Desktop and associated Terminal Server products,
  - Linux terminal servers such as the Linux Terminal Server project,
  - NoMachine's NX,
  - X Window System and its XDMCP display manager protocol.
    - Internal support for multiple, virtual desktops that the user can switch between
    - Also supports virtualization at the screen or display level, enabling window managers to use a display region that is larger than the physical size of your monitor.
- More a bandwagon use of the term "virtualization" than an exciting example of virtualization concept!
  - It does indeed make the graphical console of any supported system into a logical entity that can be accessed and used on different physical computer systems,
  - The remote console, the operating system it is running, and the applications you execute are actually still running on a single, specific physical machine.
  - Calling remote display software a virtualization technology are equivalent to considering a telescope to be a set of virtual eyeballs because you can look at something far away using one.

Network Virtualization

- The term describes the ability to refer to network resources logically rather than having to refer to specific physical network devices, configurations, or collections of related machines.
- Levels
  - single - machine, network-device virtualization that enables multiple virtual machines to share a single physical-network resource,
  - enterprise - level concepts such as virtual private networks and enterprise-core and edge-routing techniques for creating subnetworks and segmenting existing networks.
- Examples:
  - Xen relies on network virtualization through the Linux bridge - utils package to enable virtual machines to appear to have unique physical addresses (Media Access Control, or MAC, addresses) and unique IP addresses.
  - Server-virtualization solutions, such as UML, use the Linux virtual Point - to - Point (TUN) & Ethernet (TAP) network devices to provide user - space access to the host's network.
  - Advanced network switches and routers use techniques such as Virtual Routing and Forwarding (VRF), VRF - Lite, and Multi - VRF to segregate customer traffic into separately routed network segments and support multiple virtual-routing domains within a single piece of network hardware.

Server or Machine Virtualization

- The terms describe the ability to run an entire VM, including its own OS, on another OS.
- Each virtual machine that is running on the parent OS
    - is logically distinct,
    - has access to some or all of the hardware on the host system,
    - has its own logical assignments for the storage devices on which that OS is installed,
    - can run its own applications within its own operating environment.
- Server virtualization: virtualization tech type that most people think of when they hear “virtualization”!
- Not as common, term “machine virtualization” uniquely identify this type of virtualization,
    - clearly differentiates the level at which virtualization is taking place — the machine itself is being virtualized — regardless of the underlying technology used.
- Examples:
    - KVM,Microsoft Virtual Server&Virtual PC,Parallels Workstation,User Mode Linux,Virtual Iron,
    - VMware, Xen.
- Server virtualization is usually different from the term “virtual server”
    - Virtual server used to describe both
        1. the capability of e-mail & Web servs. to service multiple Internet domains, and
        2. system – level virt.techs used to provide ISP users with their own virtual server machine.

Usage of Server and Machine Virtualization

- Key aspect: different virtual machines do not share the same kernel and can therefore be running different OSs.
- Differs from system – level virtualization, where virtual servers share a single underlying kernel
- Provide a no. of unique infrastructure, customer & business opportunities:
  - Running legacy software, where you depend on a software product that runs only on a specific version of a specific operating system.
  - Software system-test and quality-assurance environments, where you need to be able to test a specific software product on many different OSs or versions of an OS.
  - Low - level development environments, where developers may want or need to work with specific versions of tools, an OS kernel, and a specific OS distribution.

Approaches to server or machine virtualization (1/3)

1. **Guest OS:**
   - Each server runs as a separate OS instance within a virtualization application that itself runs on an instance of a specific OS.
   - Examples: Parallels Workstation, VMWare Workstation, and VMWare GSX Server
   - The OS on which the virtualization application is running is often referred to as the “Host OS”.

2. **Parallel Virtual Machine:**
   - Physical or virtual systems are organized into a virtual machine using clustering software such as a PVM.
   - The cluster is capable of performing complex CPU and data - intensive calculations
   - This is more of a clustering concept than an alternative virtualization solution

Approaches to server or machine virtualization (2/3)

3. **Hypervisor - based:**
   - A small virtual machine monitor – hypervisor – runs on top of machine's hardware and provides:
     1. it identifies, traps, and responds to protected or privileged CPU operations made by each virtual machine.
     2. it handles queuing, dispatching, and returning the results of hardware requests from your virtual machines.
   - An administrative OS then runs on top of the hypervisor, as do the virtual machines themselves.
     - can communicate with the hypervisor and
     - is used to manage the virtual machine instances.
   - Example: paravirtualization concept
     - is the primary model used by Xen,
       - uses a customized Linux kernel to support its administrative environment, known as domain0.
       - runs unmodified versions of OSs on top of its hypervisor.
4. **Full virtualization:**
   - similar to paravirtualization,
   - uses a hypervisor, but incorporates code into the hypervisor that emulates the underlying hardware
   - Example: VMWare ESX server uses a customized version of Linux (known as the Service Console) as its administrative OS.

Approaches to server or machine virtualization (3/3)

5. **Kernel- level virtualization:**
   - does not require a hypervisor,
   - runs a separate version of Linux kernel & an associated virtual machine as a user – space process on the physical host.
   - Examples:
     - User - Mode Linux (UML),
       - requires a special build of the Linux kernel for guest operating systems,
     - Kernel Virtual Machine (KVM),
       - uses a device driver in the host's kernel for communication between the Linux kernel & VMs
       - requires processor support for virtualization (Intel VT or AMD – v Pacifica), and
       - uses a slightly modified QEMU process as the display and execution container for its VMs
6. **Hardware virtualization:**
   - very similar to both paravirtualization and full virtualization; uses a hypervisor
   - available only on systems that provide hardware support for virtualization:
     - latest generation of Intel (Intel VT, aka Vanderpool) and AMD (AMD - V, aka Pacifica) processors.
   - Examples of techs that take advantage of the hardware support for virtualization:
     - Hypervisor - based systems such as Xen and VMWare ESX Server,
     - kernel - level virtualization technologies such as KVM,
   - Virtual machines in a hardware virtualization environment can run unmodified OSs

Which one?

- Hypervisor - based virtualization is the most popular virtualization tech. in use today!
  - IBM’s VM operating system, VMWare’s ESX Server, Parallels Workstation, Virtual Iron products, and Xen.
- History – use of a hypervisor:
  - pioneered by the original commercial virtual-machine environment, IBM’s CP/CMS OS (1966),
  - popularized by IBM ’ s VM/370 operating system, introduced in 1970.

  …
  - 2006: VMware proposed a generic Virtual Machine Interface (VMI) that would enable multiple hypervisor-based virtualization technologies to use a common kernel level interface.
  - VMware and Xen agreed to work together to develop a more generic interface, known as paravirt_ops, which is being developed by IBM, VMware, Red Hat, and XenSource.
    - Inclusion of the paravirt_ops patches into to a kernel enable any compliant hypervisor-based virtualization technology to work with a vanilla Linux kernel,
  - Kernel projects such as KVM enable users to run VMs, themselves running any OS, on hardware that supports them, without requiring a hypervisor.

Storage Virtualization (1/3)

- The logical abstraction of physical storage.
- Around for many years:
  - logical volumes on systems such as Linux or AIX (LVM, LVM2, and EVMS packages),
  - RAID storage (Redundant Array of Inexpensive Disks),
    - Usual RAID + logical volumes as provided by the Linux are limited to use on the system to which the actual storage devices are physically attached.
    - Some RAID controllers are dual – ported: allowing multiple computers access to the same volumes and associated filesystems through controller
  - networked filesystems such as AFS and GFS.
- Combine available physical disk drives into pools of available storage
  - A volume is the logical equivalent of a disk partition.
- Enables greater amounts of physical storage to be available to individual systs,
- Enables existing filesysts to grow to hold information without resorting to an administrative shotgun blast of symbolic links and inter dependent mount points for networked storage.

Storage Virtualization (2/3)

- Use a logical volume manager:
  - user define the disk partitions to use for logical volumes,
  - create logical volumes on that physical storage
  - create a filesystem on the logical volumes.
  - mount and use these file systems just as you would mount and use filesystems that were created on physical disk partitions.
- NFS, the default Network File System for most UNIX - like OSs,
  - makes possible to share logical storage across multiple computer systems,
  - it does this by exporting a directory from a filesystem on the logical storage
- Distributed filesystem technologies such as AFS and GFS
  - have their own internal logical-volume creation and management mechanisms,
  - make it to share the filesystems on logical volumes between multiple computer systems
    - AFS and GFS provide locking mechanisms to synchronize simultaneous writes to shared filesystems over the network.
  - provide filesystem - level access to logical volumes => conceptually similar to Network Attached Storage devices, which provide filesystem - level access over a network to the filesystems that they contain.

Storage Virtualization (3/3)

- Storage Area Networks (SAN):
  - support block - level I/O
  - enable multiple systems to share low - level access to various types of storage devices over the network.
  - Most SANs use expensive, high-power network techns. such as Fibre Channel & InfiniBand to provide the high levels of throughput & general performance
- Newer technologies:
  - Provide less expensive mechanisms for getting block - level access to networked storage devices.
  - iSCSI (Internet Small Computer Systems Interface)
    - supports the use of the SCSI protocol over TCP/IP networks, and requires a special type of network controller
  - AoE (ATA over Ethernet)
    - provides block - level access to suitable ATA devices using only a standard Ethernet connection.
  - Both of these perform better on higher - bandwidth networks such as Gigabit Ethernet networks, although they are certainly usable on 100 - megabit networks.

System - Level or Operating System Virtualization

- The terms describe various implementations of running multiple, logically distinct system environments on a single instance of an operating system kernel.
- Based on the change root (chroot) concept that is available on all UNIX - like systs.
  - During the system boot process, the kernel can use root filesystems such as those provided by initial RAM disks or initial RAM filesystems to load drivers and perform other early - stage system initialization tasks.
- Chroot mechanism as used by system - level virt. is an extension of this concept:
  - enables the system to start virtual servers with their own sets of processes that execute relative to their own filesystem root directories.
- Operating within the confines of their own root directories and associated filesystem
  - prevents virtual servers from being able to access files in each others ' filesystems, and
  - if a chroot 'ed server is compromised, it has access only to files that are located within its own root filesyst
- The core differentiator between system - level and server virtualization is whether you can be running different OSs on different virtual systems.
  - If all virtual servers must share a single copy of an operating system kernel =>system - level virtualization.
  - If different virtual servers can run different Oss & different versions of a single OS => server virtualization

Solutions for System – Level virtualization

- Examples:
  - FreeBSD ' s chroot jails; Linux VServer, FreeVPS, and OpenVZ; Solaris Zones and Containers; Virtuozzo
- Advantages over server or machine virtualization:
  - Share a single instance of an OS kernel, system - level virtualization solutions are significantly lighter weight than the complete machines required by server virt.
  - Enables a single physical host to support many more " virtual servers " than the number of complete virtual machines that it could support.
  - FreeBSD's chroot jails, Linux - VServer, and FreeVPS used for years by businesses such as ISPs
    - To provide each user with their own virtual server, in which they can have relatively complete control without any chance of compromising the system ' s primary security configuration, system configuration files, and filesystem.
  - Used for *server consolidation* : running multiple virtual servers on a single physical hardware system
- The primary disadvantage of system - level virtualization:
  - a kernel or driver problem can take down all of the system - level virtual servers supported on that system.

Why Virtualization Today?

- Virtualization is not a new concept!
- Virtualization is more popular now than ever because it is now an option for a larger group of users & syst. adminis than ever before.
- General reasons for the increasing popularity of virtualization:
  - Power & performance of commodity x86 hardware continues to increase
    - Processors are faster than ever, support more memory than ever,
    - Latest multi - core processors literally enable single systems to perform multiple tasks simultaneously.
    - Increase the chance that the hardware may be under utilized.
    - Virtualization provides an excellent way of getting the most out of existing hardware while reducing many other IT costs.
  - Integration of direct support for hardware - level virtualization in the latest generations of Intel and AMD processors, and motherboards,
  - A wide variety of virtualization products for both desktop and server systems running on commodity x86 hardware
    - Many of these (like Xen) are open source software
- Virtualization is continuing to prove its worth in business and academic environments all over the world.

Advantages of Virtualization

1. Better Use of Existing Hardware
2. Reduction in New Hardware Costs.
3. Reduction in IT Infrastructure Costs.
4. Simplified System Administration.
5. Increased Uptime and Faster Failure Recovery.
6. Simplified Capacity Expansion.
7. Simpler Support for Legacy Systems and Applications.
8. Simplified System - Level Development.
9. Simplified System Installation and Deployment.
10. Simplified System and Application Testing.

Disadvantages of Virtualization

- Virtualization is not a panacea for all IT woes!
- It is not appropriate for all scenarios,
- It introduces real costs and concerns all its own:
  1. Single Point of Failure Problems
  2. Server Sharing and Performance Issues
  3. Per - Server Network Congestion
  4. Increase in Networking Complexity and Debugging Time
  5. Increased Administrative Complexity