WIKIPEDIA

Virtualization

In computing, **virtualization** refers to the act of creating a virtual (rather than actual) version of something, including virtual computer hardware platforms, storage devices, and computer network resources.

Virtualization began in the 1960s, as a method of logically dividing the system resources provided by mainframe computers between different applications. Since then, the meaning of the term has broadened.[1]

Contents

Hardware virtualization

Hardware virtualization or *platform virtualization* refers to the creation of a virtual machine that acts like a real computer with an operating system. Software executed on these virtual machines is separated from the underlying hardware resources. For example, a computer that is running Microsoft Windows may host a virtual machine that looks like a computer with the Ubuntu Linux operating system; Ubuntu-based software can be run on the virtual machine.[2][3]

In hardware virtualization, the *host machine* is the machine which is used by the virtualization and the *guest machine* is the virtual machine. The words *host* and *guest* are used to distinguish the software that runs on the physical machine from the software that runs on the virtual machine. The software or firmware that creates a virtual machine on the host hardware is called a *hypervisor* or *virtual machine monitor*.

Different types of hardware virtualization include:

- Full virtualization – almost complete simulation of the actual hardware to allow a software environments, including a guest operating system and its apps, to run unmodified.
- Paravirtualization – the guest apps are executed in their own isolated domains, as if they are running on a separate system, but a hardware environment is not simulated. Guest programs need to be specifically modified to run in this environment.

Hardware-assisted virtualization is a way of improving overall efficiency of virtualization. It involves CPUs that provide support for virtualization in hardware, and other hardware components that help improve the performance of a guest environment.

Hardware virtualization can be viewed as part of an overall trend in enterprise IT that includes autonomic computing, a scenario in which the IT environment will be able to manage itself based on perceived activity, and utility computing, in which computer processing power is seen as a utility that clients can pay for only as needed. The usual goal of virtualization is to centralize administrative tasks while improving scalability and overall hardware-resource utilization. With virtualization, several operating systems can be run in parallel on a single central processing unit (CPU). This parallelism tends to reduce overhead costs and differs from multitasking, which involves running several programs on the same OS. Using virtualization, an enterprise can better manage updates and rapid changes to the operating system and applications without disrupting the user. "Ultimately, virtualization dramatically improves the efficiency and availability of resources and applications in an organization. Instead of relying on the old model of “one server, one application” that leads to underutilized resources, virtual resources are dynamically applied to meet business needs without any excess fat" (ConsonusTech).

Hardware virtualization is not the same as hardware emulation. In hardware emulation, a piece of hardware imitates another, while in hardware virtualization, a hypervisor (a piece of software) imitates a particular piece of computer hardware or the entire computer. Furthermore, a hypervisor is not the same as an emulator; both are computer programs that imitate hardware, but their domain of use in language differs.

Snapshots

A *snapshot* is a state of a virtual machine, and generally its storage devices, at an exact point in time. A snapshot enables the virtual machine's state at the time of the snapshot to be restored later, effectively undoing any changes that occurred afterwards. This capability is useful as a backup technique, for example, prior to performing a risky operation.

Virtual machines frequently use virtual disks for their storage; in a very simple example, a 10-gigabyte hard disk drive is simulated with a 10-gigabyte flat file. Any requests by the VM for a location on its physical disk are transparently translated into an operation on the corresponding file. Once such a translation layer is present, however, it is possible to intercept the operations and send them to different files, depending on various criteria. Every time a snapshot is taken, a new file is created, and used as an overlay for its predecessors. New data is written to the topmost overlay; reading existing data, however, needs the overlay hierarchy to be scanned, resulting in accessing the most recent version. Thus, the entire stack of snapshots is virtually a single coherent disk; in that sense, creating snapshots works similarly to the incremental backup technique.

Other components of a virtual machine can also be included in a snapshot, such as the contents of its random-access memory (RAM), BIOS settings, or its configuration settings. "Save state" feature in video game console emulators is an example of such snapshots.

Restoring a snapshot consists of discarding or disregarding all overlay layers that are added after that snapshot, and directing all new changes to a new overlay.

Migration

The snapshots described above can be moved to another host machine with its own hypervisor; when the VM is temporarily stopped, snapshotted, moved, and then resumed on the new host, this is known as migration. If the older snapshots are kept in sync regularly, this operation can be quite fast, and allow the VM to provide uninterrupted service while its prior physical host is, for example, taken down for physical maintenance.

Failover

Similar to the migration mechanism described above, failover allows the VM to continue operations if the host fails. Generally it occurs if the migration has stop working. However, in this case, the VM continues operation from the *last-known* coherent state, rather than the *current* state, based on whatever materials the backup server was last provided with.

Video game console emulation

A video game console emulator is a program that allows a personal computer or video game console to emulate a different video game console's behavior. Video game console emulators and hypervisors both perform hardware virtualization; words like "virtualization", "virtual machine", "host" and "guest" are not used in conjunction with console emulators.

Nested virtualization

Nested virtualization refers to the ability of running a virtual machine within another, having this general concept extendable to an arbitrary depth. In other words, nested virtualization refers to running one or more hypervisors inside another hypervisor. Nature of a nested guest virtual machine does not need not be homogeneous with its host virtual machine; for example, application virtualization can be deployed within a virtual machine created by using hardware virtualization.[4]

Nested virtualization becomes more necessary as widespread operating systems gain built-in hypervisor functionality, which in a virtualized environment can be used only if the surrounding hypervisor supports nested virtualization; for example, Windows 7 is capable of running Windows XP applications inside a built-in virtual machine. Furthermore, moving already existing virtualized environments into a cloud, following the Infrastructure as a Service (IaaS) approach, is much more complicated if the destination IaaS platform does not support nested virtualization.[5][6]

The way nested virtualization can be implemented on a particular computer architecture depends on supported hardware-assisted virtualization capabilities. If a particular architecture does not provide hardware support required for nested virtualization, various software techniques are employed to enable it.[5] Over time, more architectures gain required hardware support; for example, since the Haswell microarchitecture (announced in 2013), Intel started to include VMCS shadowing as a technology that accelerates nested virtualization.[7]

Licensing

Virtual machines running proprietary operating systems require licensing, regardless of the host machine's operating system. For example, installing Microsoft Windows into a VM guest requires its licensing requirements to be satisfied.[8][9][10]

Desktop virtualization

Desktop virtualization is the concept of separating the logical desktop from the physical machine.

One form of desktop virtualization, virtual desktop infrastructure (VDI), can be thought of as a more advanced form of hardware virtualization. Rather than interacting with a host computer directly via a keyboard, mouse, and monitor, the user interacts with the host computer using another desktop computer or a mobile device by means of a network connection, such as a LAN, Wireless LAN or even the Internet. In addition, the host computer in this scenario becomes a server computer capable of hosting multiple virtual machines at the same time for multiple users.[11]

As organizations continue to virtualize and converge their data center environment, client architectures also continue to evolve in order to take advantage of the predictability, continuity, and quality of service delivered by their converged infrastructure. For example, companies like HP and IBM provide a hybrid VDI model with a range of virtualization software and delivery models to improve upon the limitations of distributed client computing.[12] Selected client environments move workloads from PCs and other devices to data center servers, creating well-managed virtual clients, with applications and client operating environments hosted on servers and storage in the data center. For users, this means they can access their desktop from any location, without being tied to a single client device. Since the resources are centralized, users moving between work locations can still access the same client environment with their applications and data.[12] For IT administrators, this means a more centralized, efficient client environment that is easier to maintain and able to more quickly respond to the changing needs of the user and business.[13][14]

Another form, session virtualization, allows multiple users to connect and log into a shared but powerful computer over the network and use it simultaneously. Each is given a desktop and a personal folder in which they store their files.[11] With multiseat configuration, session virtualization can be accomplished using a single PC with multiple monitors, keyboards, and mice connected.

Thin clients, which are seen in desktop virtualization, are simple and/or cheap computers that are primarily designed to connect to the network. They may lack significant hard disk storage space, RAM or even processing power, but many organizations are beginning to look at the cost benefits of eliminating “thick client” desktops that are packed with software (and require software licensing fees) and making more strategic investments.[15] Desktop virtualization simplifies software versioning and patch management, where the new image is simply updated on the server, and the desktop gets the updated version when it reboots. It also enables centralized control over what applications the user is allowed to have access to on the workstation.

Moving virtualized desktops into the cloud creates hosted virtual desktops (HVDs), in which the desktop images are centrally managed and maintained by a specialist hosting firm. Benefits include scalability and the reduction of capital expenditure, which is replaced by a monthly operational cost.[16]

Containerization

Operating-system-level virtualization, also known as containerization, refers to an operating system feature in which the kernel allows the existence of multiple isolated user-space instances. Such instances, called containers,[17] partitions, virtual environments (VEs) or jails (FreeBSD jail or chroot jail), may look like real computers from the point of view of programs running in them. A computer program running on an ordinary operating system can see all resources (connected devices, files and folders, network shares, CPU power, quantifiable hardware capabilities) of that computer. However, programs running inside a container can only see the container's contents and devices assigned to the container.

Containerization started gaining prominence in 2014, with the introduction of Docker.[18][19]

Other types

Software

- Application virtualization and workspace virtualization: isolating individual apps from the underlying OS and other apps; closely associated with the concept of portable applications

- Service virtualization: emulating the behavior of specific components in heterogeneous component-based applications such as API-driven applications, cloud-based applications and service-oriented architectures

Memory

- Memory virtualization: aggregating random-access memory (RAM) resources from networked systems into a single memory pool
- Virtual memory: giving an app the impression that it has contiguous working memory, isolating it from the underlying physical memory implementation

Storage

- Storage virtualization: the process of completely abstracting logical storage from physical storage
- Distributed file system: any file system that allows access to files from multiple hosts sharing via a computer network
- Virtual file system: an abstraction layer on top of a more concrete file system, allowing client applications to access different types of concrete file systems in a uniform way
- Storage hypervisor: the software that manages storage virtualization and combines physical storage resources into one or more flexible pools of logical storage[20]
- Virtual disk: a computer program that emulates a disk drive such as a hard disk drive or optical disk drive (see comparison of disc image software)

Data

- Data virtualization: the presentation of data as an abstract layer, independent of underlying database systems, structures and storage
- Database virtualization: the decoupling of the database layer, which lies between the storage and application layers within the application stack over all

Network

- Network virtualization: creation of a virtualized network addressing space within or across network subnets
- Virtual private network (VPN): a network protocol that replaces the actual wire or other physical media in a network with an abstract layer, allowing a network to be created over the Internet

See also

- Timeline of virtualization development
- Network Functions Virtualization
- Emulation (computing)
- Computer simulation
- Numeronym (explains that "V12N" is an abbreviation for "virtualization")
- Consolidation ratio
- I/O virtualization
- Application checkpointing

References

1. Graziano, Charles. "A performance analysis of Xen and KVM hypervisors for hosting the Xen Worlds Project" (http://lib.dr.iastate.edu/cgi/viewcontent.cgi?article=3243&context=etd). Retrieved 2013-01-29.
2. Turban, E; King, D; Lee, J; Viehland, D (2008). "Chapter 19: Building E-Commerce Applications and Infrastructure". *Electronic Commerce A Managerial Perspective*. Prentice-Hall. p. 27.

3. "Virtualization in education" (http://www-07.ibm.com/solutions/in/education/download/Virtualization%20in%20Education.pdf) (PDF). IBM. October 2007. Retrieved 6 July 2010. "A virtual computer is a logical representation of a computer in software. By decoupling the physical hardware from the operating system, virtualization provides more operational flexibility and increases the utilization rate of the underlying physical hardware."
4. Orit Wasserman, Red Hat (2013). "Nested virtualization: Shadow turtles" (http://www.linux-kvm.org/wiki/images/e/e9/Kvm-forum-2013-nested-virtualization-shadow-turtles.pdf) (PDF). KVM forum. Retrieved 2014-04-07.
5. Muli Ben-Yehuda; Michael D. Day; Zvi Dubitzky; Michael Factor; Nadav Har'El; Abel Gordon; Anthony Liguori; Orit Wasserman; Ben-Ami Yassour (2010-09-23). "The Turtles Project: Design and Implementation of Nested Virtualization" (https://www.usenix.org/legacy/event/osdi10/tech/full_papers/Ben-Yehuda.pdf) (PDF). *usenix.org*. Retrieved 2014-12-16.
6. Alex Fishman; Mike Rapoport; Evgeny Budilovsky; Izik Eidus (2013-06-25). "HVX: Virtualizing the Cloud" (http://0b4af6cdc2f0c5998459-c0245c5c937c5dedcca3f1764ecc9b2f.r43.cf2.rackcdn.com/11552-hotcloud13-fishman.pdf) (PDF). *rackcdn.com*. Retrieved 2014-12-16.
7. "4th-Gen Intel Core vPro Processors with Intel VMCS Shadowing" (https://www-ssl.intel.com/content/dam/www/public/us/en/documents/white-papers/intel-vmcs-shadowing-paper.pdf) (PDF). Intel. 2013. Retrieved 2014-12-16.
8. Foley, Mary Jo (5 July 2012). "Microsoft goes public with Windows Server 2012 versions, licensing" (https://www.zdnet.com/microsoft-goes-public-with-windows-server-2012-versions-licensing-7000000341/). *ZDNet*. CBS Interactive. "Finn explained that Standard covers 2 CPUs in a host, and goes from one VOSE (virtual operating system environment - 1 free Std install in a VM on that host) to two, and 'now has all the features and scalability of Datacenter.' He noted there will be a small price increase, but said he thought that wouldn't matter, as it 'should be virtualised anyway and the VOSE rights doubling will compensate. Windows Server Datacenter was a minimum of two 1-CPU licenses with unlimited VOSEs. 'Now it is a simpler SKU that covers two CPUs in a host with unlimited VOSEs,' Finn said."
9. "Windows Server 2012 Licensing and Pricing FAQ" (http://download.microsoft.com/download/4/D/B/4DB352D1-C610-466A-9AAF-EEF4F4CFFF27/WS2012_Licensing-Pricing_FAQ.pdf) (PDF). Microsoft. Retrieved 5 July 2012.
10. "Licensing Windows desktop operating system for use with virtual machines" (https://download.microsoft.com/download/9/8/d/98d6a56c-4d79-40f4-8462-da3ecba2dc2c/licensing_windows_desktop_os_for_virtual_machines.pdf) (PDF). *microsoft.com*. Microsoft. Retrieved 22 December 2018.
11. "Strategies for Embracing Consumerization" (http://download.microsoft.com/download/E/F/5/EF5F8B95-5E27-4CDB-860F-F982E5B714B0/Strategies%20for%20Embracing%20Consumerization.pdf) (PDF). Microsoft Corporation. April 2011. p. 9. Retrieved 22 July 2011.
12. Chernicoff, David, "HP VDI Moves to Center Stage," ZDNet, August 19, 2011. (https://www.zdnet.com/blog/datacenter/hp-vdi-moves-to-center-stage/978)
13. Baburajan, Rajani, "The Rising Cloud Storage Market Opportunity Strengthens Vendors," infoTECH, August 24, 2011. It.tmcnet.com. 2011-08-24. (http://it.tmcnet.com/channels/cloud-storage/articles/211183-rising-cloud-storage-market-opportunity-strengthens-vendors.htm)
14. "Oestreich, Ken, "Converged Infrastructure," CTO Forum, November 15, 2010. Thectoforum.com" (https://web.archive.org/web/20120113094920/http://www.thectoforum.com/content/converged-infrastructure-0). Archived from the original (https://ukessay.com/blog/converged-infrastructure) on January 13, 2012. Retrieved December 2, 2011.
15. "Desktop Virtualization Tries to Find Its Place in the Enterprise" (http://content.dell.com/us/en/enterprise/d/large-business/desktop-virtualization-place.aspx). Dell.com. Retrieved 2012-06-19.
16. "HVD: the cloud's silver lining" (https://web.archive.org/web/20121002231021/http://www.intrinsictechnology.co.uk/FileUploads/HVD_Whitepaper.pdf) (PDF). Intrinsic Technology. Archived from the original (http://www.intrinsictechnology.co.uk/FileUploads/HVD_Whitepaper.pdf) (PDF) on 2 October 2012. Retrieved 30 August 2012.
17. Hogg, Scott (2014-05-26). "Software Containers: Used More Frequently than Most Realize" (http://www.networkworld.com/article/2226996/cisco-subnet/software-containers--used-more-frequently-than-most-realize.html). *Network World*. Network World, Inc. Retrieved 2015-07-09.
18. Vaughan-Nichols, Steven J. (21 March 2018). "What is Docker and why is it so darn popular?" (https://www.zdnet.com/article/what-is-docker-and-why-is-it-so-darn-popular/). *ZDNet*. CBS Interactive.

19. Butler, Brandon (10 June 2014). "Docker 101: What it is and why it's important" (https://www.networkworld.com/article/2361465/cloud-computing/docker-101-what-it-is-and-why-it-s-important.html). *Network World*. IDG.
20. "Enterprise Systems Group White paper, Page 5" (https://web.archive.org/web/20120330235752/http://www.raido.be/frontend/files/whitepapers/20_j3ykqsmm.pdf) (PDF). Enterprise Strategy Group White Paper written and published on August 20, 2011 by Mark Peters. Archived from the original (http://www.raido.be/frontend/files/whitepapers/20_j3ykqsmm.pdf) (PDF) on March 30, 2012. Retrieved July 18, 2013.

External links

- An Introduction to Virtualization (http://www.kernelthread.com/publications/virtualization/), January 2004, by Amit Singh