

Desktop virtualization

Desktop virtualization is software technology that separates the **desktop environment** and associated **application software** from the physical client device that is used to access it.

Desktop virtualization can be used in conjunction with **application virtualization** and user profile management systems, now termed “user virtualization,” to provide a comprehensive desktop environment management system. In this mode, all the components of the desktop are virtualized, which allows for a highly flexible and much more secure desktop delivery model. In addition, this approach supports a more complete desktop disaster recovery strategy as all components are essentially saved in the data center and backed up through traditional redundant maintenance systems. If a user’s device or hardware is lost, the restore is straightforward and simple, because the components will be present at login from another device. In addition, because no data is saved to the user’s device, if that device is lost, there is much less chance that any critical data can be retrieved and compromised and also, **hosted virtual desktops** decrease IT budget while increasing productivity.

1 System architectures

Desktop virtualization implementations are classified based on whether the virtual desktop runs remotely or locally, on whether the access is required to be constant or is designed to be intermittent, and on whether or not the virtual desktop persists between sessions. Typically, software products that deliver desktop virtualization solutions can combine local and remote implementations into a single product to provide the most appropriate support specific to requirements. The degree of independent functionality of the client device is necessarily interdependent with the server location and access strategy. And virtualization is not strictly required for remote control to exist. Virtualization is employed to present independent instances to multiple users, and requires a strategic segmentation of the host server and presentation at some layer of the host’s architecture. The enabling layer—usually application software—is called a **hypervisor**. (Summarizing from.^[1])

1.1 Remote desktop virtualization

Remote desktop virtualization implementations operate as a **client/server** computing environment. Application execution takes place on a remote operating system which is linked to the local client device over a network using a remote display protocol through which the user interacts with applications. All applications and data used remain on the remote system with only display, keyboard, and mouse information communicated with the local client device, which may be a conventional PC/laptop, a **thin client device**, a **tablet**, or even a **smartphone**. A common implementation of this approach is to host multiple desktop operating system instances on a server hardware platform running a **hypervisor**. This is generally referred to as “Virtual Desktop Infrastructure” or “VDI”. It should be noted that VDI is often used incorrectly to refer to any desktop virtualization implementation.^[2]

Remote desktop virtualization is frequently used in the following scenarios:

- In distributed environments with high availability requirements and where desk-side technical support is not readily available, such as branch office and retail environments.
- In environments where high network latency degrades the performance of conventional **client/server** applications
- In environments where remote access and data security requirements create conflicting requirements that can be addressed by retaining all (application) data within the data center with only display, keyboard, and mouse information communicated with the remote client.

It is also used as a means of providing access to Windows applications on non-Windows endpoints including tablets, smart phones and non-Windows-based desktop PCs and laptops.

Remote desktop virtualization is also used as a means of resource sharing, to provide low-cost desktop computing services in environments where providing every user with a dedicated desktop PC is either too expensive or otherwise unnecessary.

For IT administrators, this means a more centralized, efficient client environment that is easier to maintain and able to respond more quickly to the changing needs of the user and business.^{[3][4]}

1.1.1 Presentation virtualization

Remote Desktop Services (formerly called Terminal Services) is a server-based computing/presentation virtualization component of **Microsoft Windows** that allows a user to access applications and data on a remote computer over a network using a remote display protocol called **RDP**. Another leading example of presentation virtualization is **Citrix XenApp**. Remote Desktop Services is functionally similar to VDI in that it hosts the users' desktop environments on remote servers; however it differs from VDI in two important ways.

1. Where a VDI service provides individual desktop operating system instances (e.g. Windows XP, Windows 7, etc.) for each user, Remote Desktop Services sessions run in a single shared server operating system (e.g. Windows Server 2008 R2, Windows Server 2012, etc.).
2. Where VDI is designed to deliver full desktop environments, Remote Desktop Services can provide connections to both individual applications or full desktops as needed.

The use of a single shared server operating system instead of individual desktop operating system instances consumes significantly fewer resources than the same number VDI sessions. At the same time VDI licensing is both more expensive and less flexible than equivalent Remote Desktop Services licenses. Together these factors can combine to make Remote Desktop Services-based remote desktop virtualization more attractive than VDI.^[5]

Remote Desktop Services is generally understood to refer to the Microsoft Windows component, however an open source remote desktop protocol server - named 'xrdp' - is under development.^[6]

However, VDI implementations allow for delivering a personalized workspace back to a user, which retains all the user's customizations. There are several methods to accomplish this. One is to deliver fully persistent VM desktops, in which a user gets a dedicated desktop that functions essentially as their own PC. Another mode is to deliver non-persistent desktops which do not include user personalization. These are VM desktops delivered from a pool which are then used for a session and returned to the pool. This approach has the advantage of keeping VM desktops "clean" as nothing is saved to the image by the user. Non-persistent desktops can be paired with user virtualization technologies to deliver user profiles and customizations (and all profiles settings, application data settings, configuration and user data), to the user upon login. The user profile is matched with a non-persistent desktop providing an experience that is very little different from a PC. All changes made to the user profile are saved to the profile and are restored to the user upon their next login. However, the VM is returned to the pool with no changes saved to it.

1.2 Application virtualization

Application virtualization improves delivery and compatibility of applications by encapsulating them from the underlying operating system on which they are executed. A fully virtualized application is not installed on hardware, in the traditional sense. Instead, a hypervisor layer intercepts the application which at runtime acts as if it is interfacing with the original operating system and all the resources managed by it, when in reality it is not. Examples of enterprise application virtualization software include **Citrix XenApp**, **Microsoft App-V**, **VMware ThinApp** and **Systancia AppliDis**. Liquidware Labs offers a slightly different approach to application virtualization with its **FlexApp** feature in its **ProfileUnity** solution. **FlexApp** does not virtualize the application, instead it virtualizes the location of the application.

1.3 User virtualization

User virtualization separates all of the software aspects that define a user's personality on a device from the operating system and applications to be managed independently and applied to a desktop as needed without the need for scripting, group policies, or use of roaming profiles. The term "user virtualization" sounds misleading; this technology is not limited to virtual desktops. User virtualization can be used regardless of platform – physical, virtual, cloud, etc. The major desktop virtualization platform vendors, Citrix, Microsoft and VMware, all offer a form of basic user virtualization in their platforms.

1.4 Layering

Desktop layering is a method of desktop virtualization that divides a disk image into logical parts to be managed individually. For example, if all members of a user group use the same OS, then the core OS only needs to be backed up once for the entire environment who share this layer. Layering can be applied to local physical disk images, client-based virtual machines, or host-based desktops. Windows operating systems are not designed for layering, therefore each vendor must engineer their own proprietary solution.

1.4.1 Desktop as a Service

For Data as a Service (DaaS), see [Data as a service](#).

Remote desktop virtualization can also be provided via a **Cloud computing** similar to that provided using a **Software as a service** model. This approach is usually referred to as **Cloud Hosted Virtual Desktops**. **Cloud Hosted Virtual Desktops** are divided into two technologies: **Managed VDI**, which is based on VDI technology

provided as an outsourced managed service and Desktop-as-a-Service (DaaS) which provides a higher level of automation and real multi-tenancy reducing the cost of the technology. The DaaS provider typically takes full responsibility for hosting and maintaining the compute, storage, and access infrastructure, as well as applications and application software licenses needed to provide the desktop service in return for a fixed monthly fee. For example, VMware's Horizon DaaS, based on VMware's acquisition of DeskTone, is a monthly fixed rate DaaS service.^[7] Amazon's WorkSpaces on Amazon EC2 is also a monthly fixed rate.^[8]

Cloud hosted virtual desktops can be implemented using both VDI and Remote Desktop Services-based systems and can be provided through public cloud, private cloud infrastructure, and hybrid cloud platforms. Private Cloud implementations are commonly referred to as "Managed VDI". Public Cloud offerings tend to be based on Desktop-as-a-Service technology.

The US Company DeskTone, which was acquired by VMware in October 2013,^[9] has the trademarks on the expressions "desktops as a service" and "DaaS" from the U.S. Patent and Trademark Office^[10]

1.5 Local desktop virtualization

Local desktop virtualization implementations run the desktop environment on the client device using hardware virtualization or emulation. For hardware virtualization, depending on the implementation both Type I and Type II hypervisors may be used.^[11]

Local desktop virtualization is well suited for environments where continuous network connectivity cannot be assumed and where application resource requirements can be better met by using local system resources. However, local desktop virtualization implementations do not always allow applications developed for one system architecture to run on another. For example, it is possible to use local desktop virtualization to run Windows 7 on top of OS X on an Intel-based Apple Mac, using a hypervisor such as VirtualBox, Parallels Desktop for Mac, or VMware Fusion, as both use the same x86 architecture, and it was possible to run Windows on a PowerPC-based Mac using Virtual PC.

2 References

- [1] Rouse, Margaret; Madden, Jack. "Desktop virtualization". TechTarget. Retrieved 1/3/2013. Check date values in: |access-date= (help)
- [2] Keith Schultz (16 March 2012). "Review: VDI without the server connection". InfoWorld. Retrieved 3 January 2013. One subset of VDI, dubbed client-hosted or offline mode

- [3] Baburajan, Rajani, "The Rising Cloud Storage Market Opportunity Strengthens Vendors," infoTECH, August 24, 2011. It.tmcnet.com. 2011-08-24.
- [4] Oestreich, Ken, "Converged Infrastructure," CTO Forum, November 15, 2010. Thectoforum.com.
- [5] "Citrix Hosted Virtual Desktops". Netrepid. Retrieved 2013-10-22.
- [6] "xrdp". xrdp. Retrieved 2013-10-22.
- [7] <http://www.vmware.com/products/daas>
- [8] <http://aws.amazon.com/workspaces>
- [9] <https://www.vmware.com/company/news/releases/vmw-euc-daas-10-15-13>
- [10] <http://tmsearch.uspto.gov/bin/showfield?f=doc&state=4810:gt9gja.2.8>
- [11] "Desktop virtualization cheat sheet". Networkworld.com. Retrieved 2010-03-17.

3 Further reading

- Paul Venezia (April 13, 2011) Virtualization shoot-out: Citrix, Microsoft, Red Hat, and VMware. The leading server virtualization contenders tackle InfoWorld's ultimate virtualization challenge, *InfoWorld*
- Keith Schultz (December 14, 2011) VDI shoot-out: Citrix XenDesktop vs. VMware View. Citrix XenDesktop 5.5 and VMware View 5 vie for the most flexible, scalable, and complete virtual desktop infrastructure, *InfoWorld*
- Keith Schultz (December 14, 2011) VDI shoot-out: HDX vs. PCoIP. The differences between the Citrix and VMware remote desktop protocols are more than skin deep, *InfoWorld*

4 External links

- xrdp open source Remote Desktop Protocol server xrdp

5 Text and image sources, contributors, and licenses

5.1 Text

- **Desktop virtualization** *Source:* https://en.wikipedia.org/wiki/Desktop_virtualization?oldid=685382435 *Contributors:* Kku, RadRafe, Palfrey, Wckdgyf, Greenrd, Chuunen Baka, Cobaltbluetony, MMCallister, Ans, AlistairMcMillan, Khalid hassani, Chowbok, J-enwiki, Discospinster, Mprove-enwiki, Xgenei, Dkushner, Enric Naval, Richi, Ivansanchez, Guy Harris, Stephan Leeds, Dziban303, Dan100, RHaworth, Rbraddy, Ruud Koot, Tabletop, Waldir, Golobm, Utuado, Bhny, Mike411, SamJohnston, Welsh, Abune, SmackBot, Elonka, Kaitai, Thumperward, Snori, Frap, Matchups, JustAnotherJoe, A5b, Zymurgy, Harryboyles, Ckatz, TastyPoutine, Shijaz, Ubaierbhat, Bwpach, Quaeler, UncleDouggy, Jcurtin, PeterAstrand, Kozuch, Hiboss1, Click23, AlandAd1, MarshBot, NapoliRoma, MER-C, Fetchcomms, Magioladitis, Theroadislong, Fallschirmjäger, Rfellows, GermanX, Edward321, Drm310, Gwern, Mbhurley, ChemNerd, Thr4wn, Tsmitty31, Kozka, Cjuans, RickManion, Mkcmmc, Kronix1986, Dave1029, Cvoisine, Brunton, Y, Jgrandy, Doctorfree, Rashadparacha, Swaq, Li-aqat76, Spazure, Echo95, ClueBot, The Thing That Should Not Be, EoGuy, Drmies, Pitvipper, Blanchardb, Lollolllolllol-enwiki, Excirial, Grafor, UnCatBot, XLinkBot, Pichpich, Skarebo, Galzigler, Lefton4ya, Addbot, Ramu50, Adegany, Rkcraig, Favonian, Comvetrix, Balabiot, Jarble, JonathanBentz, Lucas-bot, Yobot, Rickhoffman26, Lantana3, Pcap, Zippyclown, AVB, AnomieBOT, Kaseykcross, Woodyriddle, MikeBarron, Alexhobbs, Nasnema, Greenstar76, Rucadulu, Faweekee, U1024, Virtualcomputing, Wikirenee, Richsmithrr, Varslea, FrescoBot, Peppy Sue, Thinkgrid, W Nowicki, Quiksm3, Sharkoslinux, Wifone, Pairofdimes, Wisewave, BenzolBot, JayM-Goldberg, Ozbeanyman, Winterst, Takenkept99, TheVirtualizer, Xendesktop, Mikemonzon, Alec.korba, Domino27, Cjl3n, Ksg1234, Mtmoeiler, Hoverlover53, Dinamik-bot, Vrenator, Virtual desktop, Snablisky, 2nihon, Jjeff2009, Canuckian89, Vikrant270384, Jæs, Alph Bot, Garvericktechnologymarketing, Johndebay, EmausBot, Yagour, Dewritech, Sofilo38, Dre rampat, Startforce, Ccc andrew ccc, Tharaldsen, ZéroBot, Thuvictor, Les.pradels, Damtux, DavidPolitis, Amblonx, Jz3z, Moltentech, Changjeh, Donner60, UnisLumin, Tijfo098, Freeway75, RomitaNegi, Xanchester, ClueBot NG, DellTechWebGuy, Michaela.komblis, Provirtual, GreenITgirl, Widr, Helpful Pixie Bot, Calabe1992, Wbm1058, BG19bot, Omnisystems, Michael Vick PHI, Jeffs72, JMJeditor, Kieckhafer, Mchase92056, TranslucentCloud, Camberleybates, BattyBot, ChrisGualtieri, SimonBramfitt, Varinder k, Marevoula, Felixgaeta909, Numbermaniac, Cloudjpk, Vmblog, Daan Moreels, GKrokidas, I am One of Many, Ahadenfeldt, Gilwood-cs, DirectorJones, Hosteddesktopguru, Jenzb, Virtual75, Serverax, 115ash, KH-1, Haruma.Tghuen, Pablossos, Olwenmunseyzrb and Anonymous: 214

5.2 Images

5.3 Content license

- Creative Commons Attribution-Share Alike 3.0