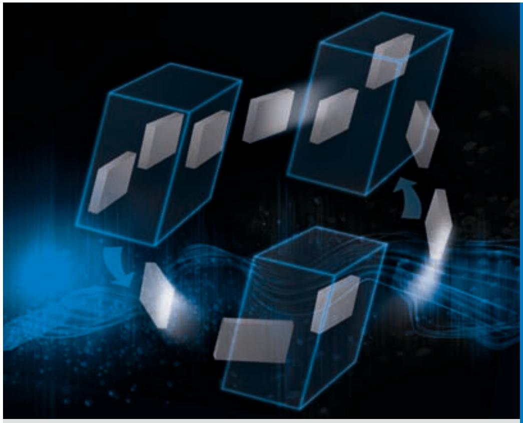
VIRTUAL DESKTOP INFRASTRUCTURE IN MICROSOFT WINDOWS SERVER 2008 R2



Virtual desktop infrastructure solutions based on the Microsoft® Windows Server® 2008 R2 Hyper-V™ platform, Microsoft System Center Virtual Machine Manager 2008 R2, and Dell™ servers and storage offer an efficient, high-performance foundation for enterprise desktop virtualization—helping to increase security, streamline client deployment and management, and reduce costs.

By David Waggoner

Ravikanth Chaganti

Gangadhar D. Bhat

Virtual desktop infrastructure (VDI) is built on a concept that has been around since the early days of computing. The term itself reflects some of the changes that have occurred in this area since the days of mainframes with remote terminals, and encompasses solutions that replace distributed desktop systems with virtual machines (VMs) that reside on centralized servers (usually with networked storage).

VDI solutions based on Microsoft Windows Server 2008 R2 Hyper-V, Microsoft System Center Virtual Machine Manager (SCVMM) 2008 R2, and Dell servers and storage provide a robust, flexible way to implement enterprise desktop virtualization (see the “Testing Microsoft VDI on Dell PowerEdge server clusters” sidebar in this article). Understanding the architecture and requirements of VDI, as well as its advantages and disadvantages, can help administrators create efficient, high-performance VDI deployments in their own environments.

VDI AND REMOTE DESKTOP SERVICES

VDI generally describes end users remotely accessing a full desktop OS environment. The end users typically connect through Remote Desktop Protocol (RDP) to an individual VM running a Microsoft Windows® client OS; they can connect from a PC running its own OS or from a thin client that includes only the software to support the RDP connection.

The ability to remotely access a virtual desktop has been available for many years through Terminal Services, now called Remote Desktop Services (RDS) in Windows Server 2008 R2. In a Terminal Services or RDS environment, multiple users access a single OS that, although usually customized on a per-user basis, lacks dedicated resources for specific users. In a VDI environment, in contrast, each user accesses a dedicated VM with an OS instance that is not shared with other users. VDI enables applications to be run as if they are on an individual PC, helping avoid the issues that can arise when running applications in an RDS environment (which can be common, because many applications are not designed for use with RDS). In a VDI environment, physical processor, memory, and disk capacity can be allocated to specific users, helping limit the effect of one user’s actions on other users.

The authors would like to thank Terry Storey of the Dell Global Infrastructure Consulting Services team for key contributions to this article.

TESTING MICROSOFT VDI ON DELL POWEREDGE SERVER CLUSTERS

During Microsoft development of Windows Server 2008 R2 Hyper-V and key management tools, Dell internal testing and early site deployments resulted in Dell filing several hundred issues and dozens of design change requests to help improve the software involved in Microsoft virtual desktop infrastructure (VDI) solutions—Windows Server 2008 R2 Hyper-V, System Center Virtual Machine Manager (SCVMM) 2008 R2, and Remote Desktop Services (RDS). Dell also ensured that updated, dedicated hardware was available to the Microsoft virtualization teams.

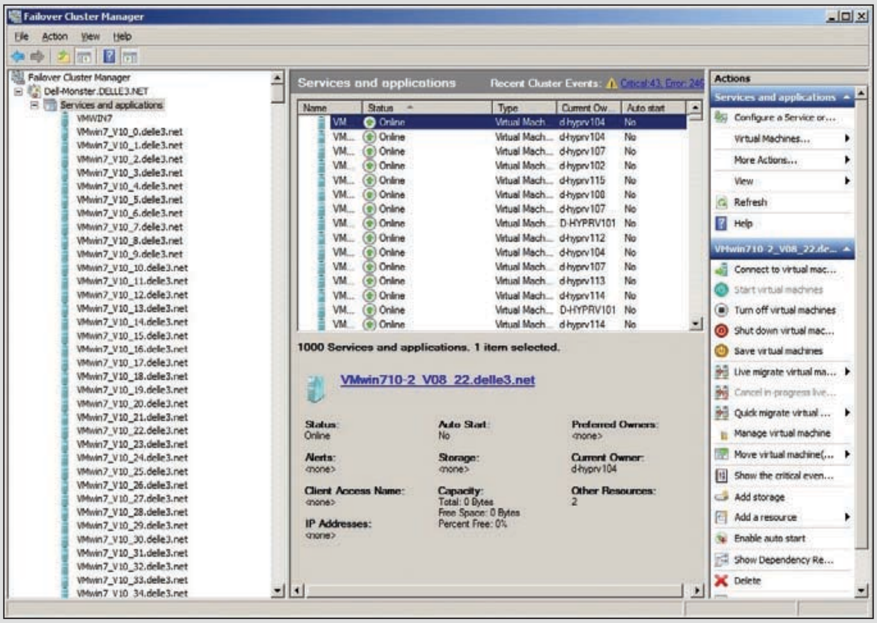
This hardware included a Dell PowerEdge™ M1000e modular blade enclosure with a mix of Intel® Xeon® processor–based PowerEdge M600 blade servers and AMD Opteron™ processor–based PowerEdge M605 blade servers. The servers incorporated a variety of mezzanine cards to support different networking and storage protocols, including Fibre Channel and Internet SCSI (iSCSI). The servers were also configured with a mix of processor types to help develop and test features such as compatibility mode, which enables virtual machine (VM) live migrations across broad families of processors. The different processors also allowed testing of the quick migration feature across the cluster.

This particular combination of blade enclosure and blade servers, deployed in a single cluster, was used during development to test the limits of Hyper-V and SCVMM performance in a VDI configuration. During this testing, the cluster was consistently able to support over 1,000 VDI VMs in a usable way (see Figure A). The VMs were successfully live migrated or quick migrated while certain operations were run inside the VMs to gauge their performance. This testing was instrumental in helping to determine the final maximum supported limit of 64 VMs per cluster host, and to help ensure that unnecessary limitations or performance hindrances were caught.

Working with a large number of VMs can be difficult and time-consuming, depending on the deployment mechanism used. Managing such a large number of VMs exposed several issues in the Hyper-V and SCVMM consoles, and led to changes enabling administrators to start, save, shut down, and edit multiple highly available VMs simultaneously.

The deployment of 1,000 VMs in a 16-node cluster was certainly challenging: the test team achieved this deployment by creating a single master Windows 7 virtual hard disk (.vhd) file with a fully configured unattend.xml file that automatically answered all prompts and performed the language selection, licensing, and naming steps. A Windows PowerShell™ script was used to export and copy the 9 GB .vhd file 1,000 times across the 16 cluster nodes.

The Windows PowerShell script processed these VMs in a batch using the resources of the cluster members, and was able to create approximately 320 VMs every 20 minutes. This process enabled the team to create all 1,000 VMs in just over an hour—although in reality, including all of the configuration and system setup time, it took approximately 24 hours to get from one functioning VM to 1,000 VMs. After the mass creation, approximately 4 percent of the VMs had some issue, with most of those (such as failed Microsoft Active Directory® joins) being relatively easy to correct.



Failover Cluster Manager
Dell-Monster.DELLE3.NET
Services and applications
VMWIN7
VMwin7_V10_0.delle3.net
VMwin7_V10_1.delle3.net
VMwin7_V10_2.delle3.net
VMwin7_V10_3.delle3.net
VMwin7_V10_4.delle3.net
VMwin7_V10_5.delle3.net
VMwin7_V10_6.delle3.net
VMwin7_V10_7.delle3.net
VMwin7_V10_8.delle3.net
VMwin7_V10_9.delle3.net
VMwin7_V10_10.delle3.net
VMwin7_V10_11.delle3.net
VMwin7_V10_12.delle3.net
VMwin7_V10_13.delle3.net
VMwin7_V10_14.delle3.net
VMwin7_V10_15.delle3.net
VMwin7_V10_16.delle3.net
VMwin7_V10_17.delle3.net
VMwin7_V10_18.delle3.net
VMwin7_V10_19.delle3.net
VMwin7_V10_20.delle3.net
VMwin7_V10_21.delle3.net
VMwin7_V10_22.delle3.net
VMwin7_V10_23.delle3.net
VMwin7_V10_24.delle3.net
VMwin7_V10_25.delle3.net
VMwin7_V10_26.delle3.net
VMwin7_V10_27.delle3.net
VMwin7_V10_28.delle3.net
VMwin7_V10_29.delle3.net
VMwin7_V10_30.delle3.net
VMwin7_V10_31.delle3.net
VMwin7_V10_32.delle3.net
VMwin7_V10_33.delle3.net
VMwin7_V10_34.delle3.net

Name	Status	Type	Current Ow...	Auto start
VM...	Online	Virtual Mach...	d-hyprv104	No
VM...	Online	Virtual Mach...	d-hyprv104	No
VM...	Online	Virtual Mach...	d-hyprv107	No
VM...	Online	Virtual Mach...	d-hyprv102	No
VM...	Online	Virtual Mach...	d-hyprv115	No
VM...	Online	Virtual Mach...	d-hyprv108	No
VM...	Online	Virtual Mach...	d-hyprv107	No
VM...	Online	Virtual Mach...	D-HYPRV101	No
VM...	Online	Virtual Mach...	d-hyprv112	No
VM...	Online	Virtual Mach...	d-hyprv104	No
VM...	Online	Virtual Mach...	d-hyprv107	No
VM...	Online	Virtual Mach...	d-hyprv113	No
VM...	Online	Virtual Mach...	d-hyprv114	No
VM...	Online	Virtual Mach...	D-HYPRV101	No
VM...	Online	Virtual Mach...	d-hyprv114	No

1000 Services and applications. 1 item selected.

VMwin710-2_V08_22.delle3.net

Status: Online
Auto Start: No
Preferred Owners: <none>
Alerts: <none>
Storage: <none>
Current Owner: d-hyprv104
Client Access Name: <none>
Capacity: Total: 0 Bytes
Free Space: 0 Bytes
Percent Free: 0%
Other Resources: 2
IP Addresses: <none>

Figure A. *Dell PowerEdge blade server cluster supporting 1,000 Microsoft VDI VMs*

VDI can offer several advantages, including the following:

- **Reduced costs:** With few (or no) moving parts, thin clients can often last longer than traditional PCs, and are typically less expensive to operate and maintain. Legacy PCs not capable of running a modern OS can be repurposed for maximum utility, if upgrade cost avoidance is a more immediate concern than the potential power savings.
- **Rapid client deployment and centralized management:** Because VDI avoids the need for custom hardware and drivers and requires fewer images than traditional deployments, it can help accelerate and simplify

client provisioning and maintenance. Administrators can quickly clone VMs and make them available for use, without requiring refresh time on the client hardware.

- **Enhanced security:** Centralizing clients helps simplify physical security, because no data is kept on the end user's system—the data is inside the VM, which could be kept in a secure facility far away from the user's location. It is also generally easier to secure network access to Hyper-V host servers than to do so for many individual client systems.
- **Simplified backup and recovery:** Having the VMs on server-class hardware inside a data center helps simplify backup and recovery processes for user data and configurations, enabling administrators to take advantage of powerful tape drives, deduplication software, and (in some cases) snapshots. If a VM becomes corrupt, administrators can quickly perform a recovery or simply create a new VM. If users' hardware fails, the VMs are waiting in the same state when they reattach. And because users can connect even on dissimilar hardware or from a different location, if a specific office is inaccessible, they can go virtually anywhere (assuming security allows) and resume work.

VDI can also come with several disadvantages, however. It requires constant connectivity to the network, so network outages or disruptions can limit or prevent users from getting work done. Ensuring redundant, reliable connections with appropriate bandwidth between Hyper-V hosts and end users is therefore a key part of VDI.

In addition, remote protocols may not be well suited for some tasks: high-end graphic design or video editing, for example, can be difficult over RDP because of delays and limited hardware audio and video acceleration. Video content with a high frame rate also generally does not work as well as on a traditional PC with its own OS. Microsoft has introduced significant enhancements in Windows 7 and Windows Server 2008 R2 to help increase multimedia performance in many cases, and content played back using Microsoft Windows Media® Player can be redirected to the user's local client in native format and played back there (assuming the client has the appropriate hardware).[1]

VDI also requires a serious commitment to virtualization, including the appropriate administrative expertise and server, storage, and networking hardware to help ensure uptime and responsiveness. This commitment can increase expenses in the short term, but the more server workloads organizations virtualize, the higher the likelihood that they can leverage the appropriate skill set and hardware.

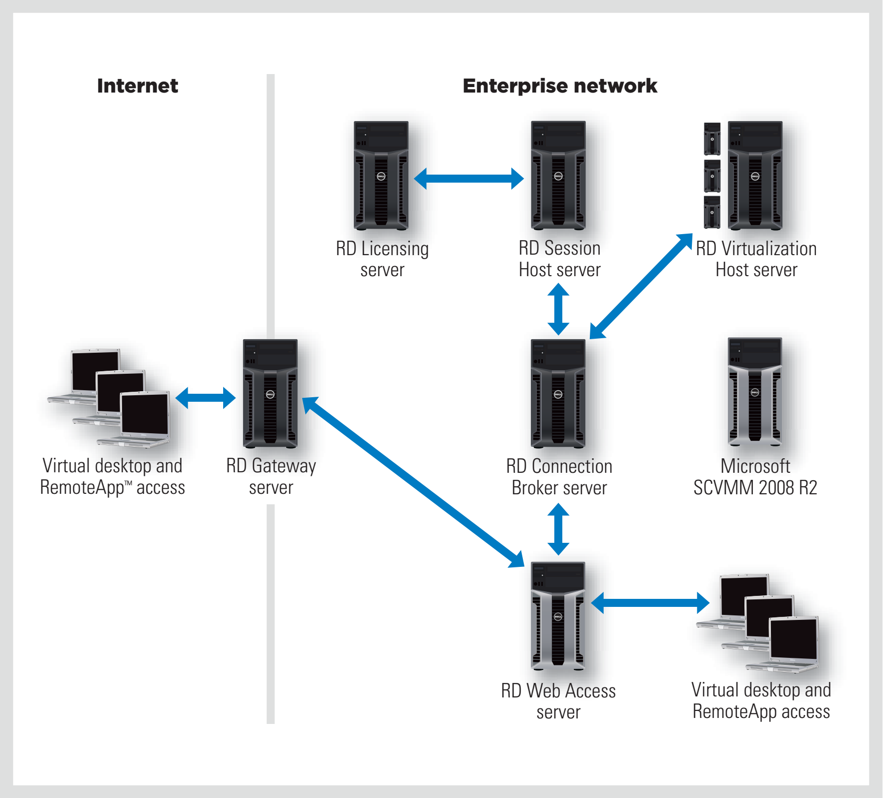

Internet
Enterprise network
RD Licensing server
RD Session Host server
RD Virtualization Host server
Virtual desktop and RemoteApp™ access
RD Gateway server
RD Connection Broker server
Microsoft SCVMM 2008 R2
RD Web Access server
Virtual desktop and RemoteApp access

Figure 1. *Typical VDI deployment based on Microsoft Windows Server 2008 R2 Hyper-V*

VDI ARCHITECTURE AND SERVER ROLES

A typical VDI deployment based on Windows Server 2008 R2 Hyper-V has each RDS role installed on a separate physical server (see Figure 1). Typically, when a client system requests access to a virtual desktop, the Remote Desktop (RD) Web Access server first sends the request to the RD Connection Broker server. This server then sends a request to the RD Virtualization Host server to start a VM in the virtual desktop pool. After the VM has started, the RD Connection Broker sends the VM name to the RD Session Host server, which redirects the virtual desktop session to the client. If a virtual desktop request originates from outside the enterprise network or from the Internet, the request is first handled by the RD Gateway server, and then forwarded to the RD Web Access server.

[1] For more information on these enhancements, visit technet.microsoft.com/en-us/library/dd560636(WS.10).aspx.

Server role	Description
RD Web Access	Enables access to Windows-based programs and virtual desktops hosted on an RD Session Host server through either Terminal Services RemoteApp and Remote Desktop Connection or a Web browser
RD Connection Broker	Enables access to load balancing in an RD Session Host server farm
RD Virtualization Host	Enables access to Hyper-V VMs as virtual desktops
RD Session Host	Hosts Windows-based programs and virtual desktops
RD Gateway	Enables access to RDS from Internet-connected devices outside the enterprise network
RD Licensing	Manages client access licenses (CALs) for each device or user accessing an RD Session Host server

Figure 2. *Remote Desktop Services server roles in Microsoft Windows Server 2008 R2*

Administrators should keep in mind that Figure 1 depicts the default behavior of the RD Connection Broker, with no IP redirection or routing token redirection. In this configuration, all client requests would be redirected by the RD Connection Broker to the RD Session Host server.

As shown in Figure 1, VDI implementations require deploying multiple server roles; Figure 2 provides a brief description of each of these roles. The servers hosting these roles, as well as the virtual desktops hosted as Hyper-V VMs, must be in the same Microsoft Active Directory domain.[2]

Using tools like SCVMM can help substantially reduce the amount of time required to provision new virtual desktops. Administrators can also take advantage of new tools introduced in Windows Server 2008 R2 such as offline domain join (djoin.exe) to create preconfigured VM templates in SCVMM and dynamically join them to an Active Directory domain. Administrators should keep in mind that the offline domain join feature is supported only in Windows 7 and Windows Server 2008 R2.

When implementing VDI, administrators should also be sure to plan for increased user loads, changing IT environments, and infrastructure availability. Implementing a load-balancing server farm model within the RDS environment can help ensure that the VDI infrastructure is both scalable and highly available.

LOAD BALANCING AND HIGH AVAILABILITY

Because Microsoft VDI uses Windows Server 2008 R2 RDS as its foundation, building high availability into the RDS environment can be critical. As the number of users accessing virtual desktops increases, the RD Session Host server can potentially become a bottleneck. To help avoid this problem, administrators can configure the RD Connection Broker server to load balance sessions between RD Session Host servers.

Implementing load balancing enables users to reconnect to an existing session in a load-balanced server farm. Administrators can deploy RD Connection Broker load balancing using the traditional Domain Name System (DNS) round-robin method, the native Windows Server 2008 R2 Network Load Balancing (NLB) feature, or a hardware load balancer. Using NLB or a hardware load balancer can help avoid limitations of the DNS round-robin method such as the caching of DNS requests on the clients, which can cause connection problems. In NLB implementations, the RD Connection Broker acts as a front-end coordinator for incoming requests and transfers the requests to the RD Session Host farm. When implementing RD Connection Broker load balancing, all servers in the farm must be running either Windows Server 2008 or Windows Server 2008 R2.

The RD Virtualization Host server integrates with Hyper-V to provide VMs as virtual desktops to end users. Administrators can help ensure high availability for these VMs by configuring the Hyper-V host servers in a failover cluster and enabling the live migration feature, which allows VMs to be migrated between Hyper-V hosts without perceived downtime by end users. Migrations can be performed for maintenance purposes, to help increase performance, or for other administrative reasons.

BEST PRACTICES FOR DEPLOYMENT AND MANAGEMENT

The number of VMs that a Hyper-V host server can support is typically directly proportional to the amount of physical memory on the host. For example, in a host with 16 GB of physical memory running VMs configured to use 1 GB of memory each, the server could only support up to 15 VMs. Administrators should be sure to evaluate their deployments based on the total number of virtual desktops they plan to host and the physical memory required to host them.

On the RD Virtualization Host server, adequate storage space for the virtual hard disk (.vhd) files is essential. Using external storage such as a Dell EqualLogic™ PS6000X Internet SCSI (iSCSI) array to host the .vhd files for each VM may help increase I/O performance and redundancy.

Each RDS component has its own management console for configuring and

[2] For more information on VDI server roles, visit technet.microsoft.com/en-us/library/dd560658(WS.10).aspx.

monitoring its functionality. SCVMM supports management of VMs that are a part of the VDI deployment, providing a centralized console for managing RD Virtualization Host servers across the enterprise. Key features of SCVMM 2008 R2 include VM templates, the ability to perform scheduled jobs, integrated live migration and quick migration capabilities, and more, helping to simplify the deployment and management of virtual desktops.[3]

FLEXIBLE, EFFICIENT INFRASTRUCTURE

VDI is designed to enable end users to access their individual desktops and data safely from virtually any network, while helping IT administrators to increase security, accelerate deployment of new capabilities without requiring the deployment and configuration of new hardware, reduce application testing requirements and compatibility issues, and simplify disaster recovery and compliance. For the enterprise, implementing VDI can help reduce hardware costs, client servicing requirements, and overall power consumption. Investing in a VDI solution based on Microsoft Windows Server 2008 R2 Hyper-V, Microsoft SCVMM 2008 R2, and highly efficient Dell servers and storage can help maximize flexibility and efficiency and help lower the costs of providing desktop services to end users.

David Waggoner is a software engineer in the Dell Server Operating Systems Group.

Ravikanth Chaganti is a lead engineer on the Dell Enterprise OS Engineering team.

Gangadhar D. Bhat is a senior Microsoft Windows debug engineer on the Dell Enterprise OS Engineering team.

MORE ONLINE
DELL.COM/PowerSolutions

QUICK LINKS

Microsoft Windows Server 2008 R2:
DELL.COM/WindowsServer2008
www.microsoft.com/windowsserver2008

Microsoft RDS team blog:
blogs.msdn.com/rds

[3] For more information on SCVMM 2008 R2, visit www.microsoft.com/systemcenter/virtualmachinemanager.