



# Creating, Deploying, and Managing SQL Server Virtual Appliances

Deploying SQL Server virtual appliances for Microsoft Hyper-V is often perceived as a complex and costly process. In this expert E-Guide, brought to you by SearchSQLServer.com and Dell, you will discover the basic steps essential for effectively managing SQL Server virtual appliances. Learn why SQL Server database mirroring provides efficient protection for today's virtual databases. Explore how your businesses can effectively create and deploy SQL Server virtual appliances for Hyper-V.

*Sponsored By:*





# Creating, Deploying, and Managing SQL Server Virtual Appliances

## Table of Contents:

[Protect virtual databases through SQL Server database mirroring](#)

[How to create SQL Server virtual appliances for Hyper-V](#)

[Deploying a SQL Server virtual appliance for Microsoft Hyper-V](#)

[Resources from Dell Inc, Microsoft and AMD](#)



## Protect virtual databases through SQL Server database mirroring

Danielle Ruest and Nelson Ruest, Contributors

Virtualizing applications like SQL Server can help an organization reduce costs. Furthermore, ever since Microsoft changed its SQL Server license model to a per CPU license model, it is possible to virtualize numerous instances of SQL Server on the same physical server. These licenses are costly, and therefore it is up to you to create a powerful host server that can run multiple virtual machines (VM) and SQL Server instances.

Organizations often run several different types of SQL Server databases. For example, many management products come with run-time databases which are deemed operational, and can -- and should -- be consolidated into a central location to reduce the amount of locations SQL Server engines run. You may also have production databases -- databases which are tied to internal applications -- which are deemed informational databases. Other types of databases include financial, organizational, departmental and geographical.

While each database is important to your organization, not all of them require the same level of protection. As a result, it's a good idea to use internal SQL Server features like database mirroring, a tool that provides protection for critical databases running in VMs (see Figure 1).

**Figure 1**



Database mirroring applies fault tolerance at the database level and automatically duplicates all of the contents of a database into another SQL Server installation. It will also automatically change over to the secondary database if the primary database is not available.

The mirrored database can also be used to provide additional functionalities, like reporting services. You can even perform backups from the mirrored copy, avoiding any performance impacts on the production database.

Since your databases have different protection requirements, you can use other, non-critical database engines from different VMs to protect important data.

Furthermore, database mirroring does not require special hardware or software tools, as databases can be mirrored from one SQL Server VM to another. In the event of a host server failure -- or even a VM failure -- users are automatically redirected to the mirror database with little or no service interruption.

If you decide to run database mirroring in your VMs, *you should not make the virtual machines highly available through host server clustering*. Otherwise the host cluster will restart the VM in the case of a failure. This could cause two versions of the same database to be live on the your network -- something to avoid at all costs.

Overall, features like database mirroring allow you to mix and match the types of databases you run in VMs. This lets you reduce costs while centralizing all of your SQL Server installations and providing protection for the databases that run your most critical data.

# BETTER BUSINESS INTELLIGENCE AT A BETTER PRICE UP TO **72% OFF**

Turn your raw data into a powerful strategic advantage with Business Intelligence solutions from Dell™ and Microsoft®—and do it for up to 72% less per terabyte than the competition.\* Built on industry standards, Microsoft® SQL Server® 2005 systems from Dell are designed to speed implementations, lower risk, and reduce complexity—all while delivering the best price-for-performance in the industry.

SIMPLIFY YOUR IT AT [DELL.COM/SQLBI](http://DELL.COM/SQLBI)



\*72% claim based upon a comparison of list prices of typical Business Intelligence offerings from leading hardware manufacturers versus Dell/Microsoft combined offerings. Benchmarked systems configured with 4-5 TB of data storage, database application software, and Business Intelligence analytic software. Dell is a trademark of Dell Inc. ©2008 Dell Inc. All rights reserved.

## How to create SQL Server virtual appliances for Hyper-V

Danielle Ruest and Nelson Ruest, Contributors

The ability to work with virtual appliances (VAP) or virtual machines (VM) that are pre-configured to run specific applications is one of the most impressive aspects of virtual infrastructures.

To deploy an application -- like SQL Server -- into your network, all you need to do is generate a new VM from the source machine or appliance.

There are two types of VAPs --commercial and private. Commercial VAPs are generated by software manufacturers and can be downloaded from the manufacturer's website, while private VAPs are built in-house and are used for the organizations current applications.

Commercial VAPs based on open source code are readily available from a variety of sources including the VMware VAP Marketplace. These VAPs come in Open Virtualization Format (OVF) and as a result, can be converted to almost any hypervisor environment. For example, you can use either Citrix' Project Kensho or Sun's xVM VirtualBox to convert a commercial VAP into Hyper-V format.

On the other hand, when you need to work with operating systems that are not based on open source code, or when you want to work with a VAP for an application you already own, then you need to create private VAPs.

This is the case for appliances based on Microsoft Windows Server, like SQL Server VAP.

To deploy additional SQL Server VMs in your environment, you should create a private SQL Server VAP. This will reduce the time it takes to launch these machines.

Start with a base VM, which ideally will be running Windows Server 2008 or Windows Server 2008 R2. These operating systems are enlightened guests in a Hyper-V environment. Also, consider using the Enterprise Edition because it supports the ability to create failover clusters and can scale more easily than the Standard Edition.

The base VM should include the operating system, any required updates like Service Pack 2 for Windows Server 2008, and any required utility including management agents and/or anti-malware engines.

Configure the base operating system according to your organization's standards. You may want to copy the VM since it can be used to create appliances running a multitude of other Microsoft Server applications like Internet Information Server, SharePoint Services or Office SharePoint Server, Commerce Server and more.

Since the base virtual machine will be used for SQL Server, consider adding two additional disks to the VM. A larger disk can be used as the data disk while a smaller disk can be used as the disk storing the transaction logs.

Once the base VM is ready, install SQL Server -- ideally version 2008 or 2008 R2. First install all of the prerequisites for SQL Server, and then install SQL Server itself.

Next, configure SQL Server to store data on the data disk — usually the D: drive — and the transaction logs on the log disk — usually the E: drive. Also perform any configuration modifications you need based on your organizational standards.

Lastly, install any required updates for the SQL Server version.

Your VM is now ready to turn into a VAP.



## Deploying a SQL Server virtual appliance for Microsoft Hyper-V

Danielle Ruest and Nelson Ruest, Contributors

Virtual appliances (VAP) are fully configured virtual machines (VM) that include both an operating system and an application. As a result, they are one of the most powerful aspects of virtual infrastructures.

Deploying a VAP is simple: create a copy of the VM and then configure it for use. Compare that to the complexity of server application configurations and you will quickly warm up to using VAPs.

For commercial applications like Microsoft SQL Server, a private VAP needs to be created. Downloadable, pre-configured VMs are not available because of licensing requirements for both the operating system (Windows Server) and the application itself.

To create a private VAP, first generate a virtual machine, then install the operating system and lastly, install the application. This is a time-consuming process. Therefore, once the VAP is complete, create a copy of it to use as a backup. This will be useful when you need to install updates in the appliance.

Typically, private VAPs targeted for Hyper-V environments include the following characteristics:

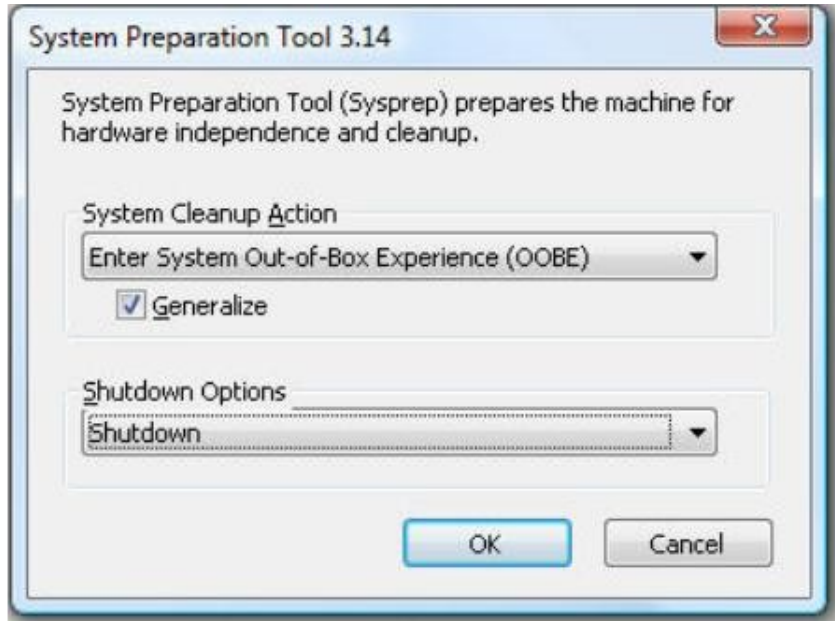
- They are based on a custom VM that runs a copy of Windows Server 2008 Enterprise Edition. Windows Server 2008 is an enlightened guest operating system for Hyper-V and as a result, it will be fully integrated with the hypervisor. Also, the Enterprise Edition allows you to scale the VM more easily than other editions and install guest failover clustering, should you need it.
- The machine includes any required patches or updates. It also includes any custom utility required for your environment.
- The machine includes at least three virtual disks: a system disk, a data disk and a logging disk.
- The machine includes SQL Server (usually SQL Server 2008) in the appropriate edition to meet your needs. It also includes a custom configuration for SQL Server -- i.e., data is stored on the data disk, logs are stored on the logging disk and other settings are modified -- as well as the appropriate updates.
- The machine has been depersonalized by the Sysprep utility.

The last step, depersonalization, is required to eliminate the security identifier (SID) and other specific components of the VM, and it turns the VM into a VAP. In Windows Server 2008, go to the %SystemRoot%\System32\Sysprep folder to launch Sysprep.exe (you'll need elevated rights to do so).

Choose the following settings (see Figure 1):

- System Cleanup Action: Enter System Out-of-Box Experience (OOBE)
- Select the Generalize checkbox
- Shutdown Options: Shutdown

**Figure 1. Preparing a virtual appliance with Sysprep**



Once the machine is shut down, you can use it to generate new virtual machines running SQL Server. Make sure you create a copy of this VM to use as a backup. The machine can then be stored in a Hyper-V source library.

In order to create a new VM based on this VAP:

1. Make sure the virtual machine is turned off.
2. Locate the source files for the VM and copy them to a new location. Create a folder using the name of the VM and store the virtual disks in this folder.
3. Generate a new virtual machine in Hyper-V and attach it to the copied disks. Assign appropriate RAM, network interface cards and other settings to the new VM.
4. Launch the new VM to generate a new security identifier, name it and join it to your production domain.
5. Load the VM with production databases and start using it as a new machine in your infrastructure.

This process is relatively simple, however, if you manage your Hyper-V environment with System Center Virtual Machine Manager (SCVMM), then you can automate the entire process (see Figure 2) and have SCVMM both prepare the VAP and generate a new VM from the VAP template in a few easy steps.

**Figure 2**

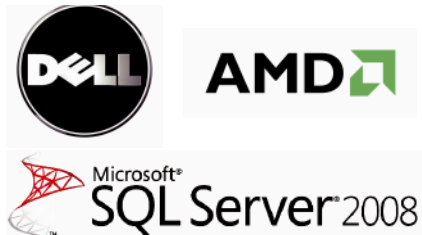
|   | Step  | Name                                                                      | Status | Start Time        | End Time          |
|---|-------|---------------------------------------------------------------------------|--------|-------------------|-------------------|
| ✓ | 1     | Create template                                                           | 100 %  | 2/9/2009 5:02:... | 2/9/2009 5:27:... |
| ✓ | 1.1   | Sysprep virtual machine                                                   | 100 %  | 2/9/2009 5:02:... | 2/9/2009 5:07:... |
| ✓ | 1.1.1 | Start virtual machine for sysprep                                         | 100 %  | 2/9/2009 5:03:... | 2/9/2009 5:03:... |
| ✓ | 1.1.2 | Stop virtual machine                                                      | 100 %  | 2/9/2009 5:07:... | 2/9/2009 5:07:... |
| ✓ | 1.2   | Store virtual machine from ServerFull01.contoso.com to servercore01.co... | 100 %  | 2/9/2009 5:07:... | 2/9/2009 5:27:... |
| ✓ | 1.2.1 | Run pre checks for transfer                                               | 100 %  | 2/9/2009 5:07:... | 2/9/2009 5:07:... |
| ✓ | 1.2.2 | Change virtual machine status                                             | 100 %  | 2/9/2009 5:07:... | 2/9/2009 5:07:... |
| ✓ | 1.2.3 | Deploy file (using LAN)                                                   | 100 %  | 2/9/2009 5:07:... | 2/9/2009 5:07:... |
| ✓ | 1.2.4 | Export Hyper-V virtual machine                                            | 100 %  | 2/9/2009 5:07:... | 2/9/2009 5:07:... |
| ✓ | 1.2.5 | Deploy file (using LAN)                                                   | 100 %  | 2/9/2009 5:07:... | 2/9/2009 5:26:... |
| ✓ | 1.2.6 | Remove virtual machine                                                    | 100 %  | 2/9/2009 5:26:... | 2/9/2009 5:27:... |
| ✓ | 1.2.7 | Fix up differencing disks                                                 | 100 %  | 2/9/2009 5:27:... | 2/9/2009 5:27:... |

Summary
Details
Change Tracking

That's it - your SQL Server VAP is now ready for deployment.

If you run virtual infrastructures and need to generate new VMs running the same applications, VAPs are the answer. Furthermore, you can -- and should -- create private VAPs of as many applications as possible since this will save you a lot of time when you need to deploy new machines.

## Resources from Dell Inc, Microsoft and AMD



[Consolidating SQL Server 2000 Databases from Dell PowerEdge 2650 Servers to a Dell PowerEdge R805 Server](#)

[Moving SAP Off Mainframe to SQL Server Nets Australia's QR Limited U.S.\\$100,000 a Month](#)

[ServiceU Achieves Outstanding Power Savings and Performance Improvements](#)

### **About Dell Inc, Microsoft and AMD:**

Together – Dell, AMD and Microsoft are working to provide our customers with solutions for line of business applications, collaboration and business intelligence. Powered by AMD Opteron™ Processors, Dell PowerEdge 2 and 4 socket servers loaded with Microsoft's SQL Server 2008 and Windows Server 2008, deliver best in industry performance. They run cooler, and deliver benchmark setting throughput that meets, and often exceeds, the demands of business users. With the added capability of Dell Professional Services to assist with SQL Server, Data Warehousing and Business Intelligence, Dell is a single source for Database and BI solutions.